Microsoft System Center 2012 R2

System Center 2012 R2 Orchestrator Documentation

Microsoft Corporation

Published: November 1, 2013

Authors

Curtis Love and Brian Wren

Applies To

System Center 2012 - Orchestrator Orchestrator in System Center 2012 SP1 System Center 2012 R2 Orchestrator

Feedback

Send suggestions and comments about this document to sc2012docs@microsoft.com.

Copyright

This document is provided "as-is". Information and views expressed in this document, including URL and other Internet website references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes. You may modify this document for your internal, reference purposes.

© 2013 Microsoft Corporation. All rights reserved.

Microsoft, Access, Active Directory, ActiveSync, ActiveX, Authenticode, Bing, BitLocker, Excel, Forefront, Hyper-V, Internet Explorer, JScript, Microsoft Press, MSDN, Outlook, SharePoint, Silverlight, SoftGrid, SQL Server, Visio, Visual Basic, Visual C++, Visual Studio, Win32, Windows, Windows Intune, Windows Mobile, Windows PowerShell, Windows Server, Windows Server System, and Windows Vista are trademarks of the Microsoft group of companies. All other trademarks are property of their respective owners.

Revision History

Release Date	Changes
October 17, 2013	Original release of this guide.
November 1, 2013	Minor updates for this guide.

Contents

Getting Started with System Center 2012 - Orchestrator	14
What's New in System Center 2012 R2 Orchestrator	
What's New in Orchestrator in System Center 2012 Service Pack 1	15
Orchestrator Capabilities	16
Orchestrator Architecture	18
Orchestrator Terminology	
Glossary for System Center 2012 - Orchestrator	
Glossary for Opalis Integration Server 6.3	
Orchestrator Resources	26
Upgrading System Center 2012 SP1 Orchestrator to System Center 2012 R2	27
Tasks to Perform Before You Begin the Upgrade	
How to Upgrade System Center 2012 SP1 Orchestrator to System Center 2012 R2	29
Troubleshoot Your Orchestrator Installation	30
How to Uninstall and Unregister an Integration Pack	32
Upgrading System Center 2012 - Orchestrator to System Center 2012 SP1	33
Tasks to Perform Before You Begin the Upgrade	
How to Upgrade System Center 2012 – Orchestrator to System Center 2012 SP1	
Troubleshoot Your Orchestrator Installation	
How to Uninstall and Unregister an Integration Pack	
Deploying System Center 2012 - Orchestrator	20
Deployment Overview	
Plan Your Orchestrator Deployment	
System Requirements	
Single-Computer Requirements for System Center 2012 R2 Orchestrator	
Single-Computer Requirements for Orchestrator in System Center 2012 SP1	
Single-Computer Requirements for System Center 2012 - Orchestrator	
Individual Feature Requirements	
Runbook Designer Requirements for System Center 2012 R2 Orchestrator	
Orchestrator Web Service Requirements for System Center 2012 R2 Orchestrator	
Runbook Server Requirements for System Center 2012 R2 Orchestrator	
Management Server Requirements for System Center 2012 R2 Orchestrator	
Runbook Designer Requirements for Orchestrator in System Center 2012 SP1	
Orchestrator Web Service Requirements for Orchestrator in System Center 2012 SP1	
Runbook Server Requirements for Orchestrator in System Center 2012 SP1	
Management Server Requirements for Orchestrator in System Center 2012 SP1	54
Management Server Requirements for System Center 2012 - Orchestrator	55
Runbook Server Requirements for System Center 2012 - Orchestrator	56

Orchestrator Web Service Requirements for System Center 2012 - Orchestrator	57
Runbook Designer Requirements for System Center 2012 - Orchestrator	58
Orchestrator Security Planning	59
Orchestrator Service Accounts	59
Orchestrator Users Group	61
Orchestration Database Security	61
Runbook Security	66
Orchestrator Web Service and Orchestration Console Security	66
Using Windows Firewall with Orchestrator	67
Orchestrator Security Scenarios	68
Orchestrator Data Encryption	72
TCP Port Requirements	74
Scale Planning	76
Feature Performance Considerations	77
Evaluate System Requirements	78
Deployment Recommendations	81
Install Orchestrator	82
How to Install Orchestrator on a Single Computer	83
Install Individual Orchestrator Features	86
How to Install a Management Server for System Center 2012 - Orchestrator	86
How to Install a Runbook Server for System Center 2012 - Orchestrator	88
How to Install the Orchestrator Web Service	90
How to Install the Runbook Designer for System Center 2012 - Orchestrator	92
Install with the Orchestrator Command Line Install Tool	93
Use Sysprep to Deploy Orchestrator	95
Perform Post-Installation Tasks	
How to Install an Integration Pack	98
How to Install GnuPG	. 101
Migrate Opalis Policies to Orchestrator	
Policy Migration Overview	. 102
Planning your Opalis Policy Migration	. 102
Supported Opalis Policy Migration Paths	. 103
System Requirements	. 103
Modify Opalis Policies that Contain Special Characters	. 104
How to Migrate Opalis Policies to Orchestrator	. 105
Modify Migrated Orchestrator Runbooks	
How to Test Migrated Runbooks	
How to Configure the Orchestrator Web Service to use HTTPS	. 110
Troubleshoot Your Orchestrator Installation	. 111
Administering System Center 2012 - Orchestrator	. 114
Accessibility for People with Disabilities	. 115

Accessibility Features of Orchestrator	116
Accessibility Features of Orchestrator Help	117
Accessibility Products and Services from Microsoft	119
How to Change the Orchestrator Database	121
How to Change the Orchestrator Users Group	123
How to Configure Orchestrator Database Connections	125
How to Configure Runbook Servers to Optimize Performance of .NET Activities	126
How to Configure Runbook Throttling	126
How to View Orchestrator Data by Using PowerPivot	127
Orchestrator Logs	130
Runbook logs	131
Activity Events	135
Audit Trail	137
Trace Logs	141
Runbook Server Properties	143
How to Back up Orchestrator	143
Migrate Orchestrator Between Environments	144
Best Practices Analyzer	148
How to Benchmark	151
How to Recover a Database	155
Database Sizing and Performance	156
How to Recover Web Components	165
Feature Performance Considerations	165
How to Use the Integration Toolkit to Extend Orchestrator Capabilities	166
Using the Orchestration Console in System Center 2012 - Orchestrator	167
Overview of Orchestration Console	168

Orchestration Console Browser Requirements	170
How to Start the Orchestration Console	171
How to Work With Runbooks in the Orchestration Console	172
Using Runbooks in System Center 2012 - Orchestrator	174
Runbook Concepts	174
Runbooks	175
Runbook Properties	176
Runbook Permissions	180
Activities	181
Standard Activities	182
Monitoring Activities	182
Customized Activities	183
Common Activity Properties	183
Workflow Control	186
Starting Point	186
Smart Links	187
Embedded Loops	191
Tools	193
Runbook Designer	194
Runbook Tester	195
Design and Build Runbooks	197
Designing a Runbook	197
Building a Runbook	198
Data Manipulation	200
Computer Groups	201
Counters	202

Functions	204
Regular Expressions	208
Schedules	210
Variables	213
Published Data	214
How to Test a Runbook	219
Deploy and Start Runbooks	220
Deploying Runbooks	220
Running Runbooks	222
Runbook Samples	222
Creating and Testing a Sample Runbook	223
Monitor a Folder within a Runbook	227
Runbook Activity Reference for System Center 2012 - Orchestrator	230
Standard Activities	231
Alphabetical List of Standard Activities	232
Ports and Protocols of Standard Activities	234
System	236
Run Program	237
Run .Net Script	241
End Process	244
Start/Stop Service	245
Restart System	247
Save Event Log	248
Query WMI	251
Run SSH Command	252
Get SNMP Variable	255

Monitor SNMP Trap	257
Send SNMP Trap	
Set SNMP Variable	262
Scheduling	264
Monitor Date/Time	
Check Schedule	267
Monitoring	268
Monitor Event Log.	269
Monitor Service	271
Get Service Status	273
Monitor Process	274
Get Process Status	276
Monitor Computer/IP	277
Get Computer/IP Status	278
Monitor Disk Space	279
Get Disk Space Status	280
Monitor Internet Application	281
Get Internet Application Status	286
Monitor WMI	292
File Management	293
Compress File	294
Copy File	296
Create Folder	299
Decompress File	300
Delete File	301
Delete Folder	303

Get File Status	304
Monitor File	306
Monitor Folder	308
Move File	311
Move Folder	313
PGP Decrypt File	314
PGP Encrypt File	316
Print File	319
Rename File	320
Email	322
Send Email	323
Notification	326
Send Event Log Message	326
Send Syslog Message	327
Send Platform Event	328
Utilities	329
Apply XSLT	330
Query XML	331
Map Published Data	332
Compare Values	336
Write Web Page	338
Read Text Log	340
Write to Database	342
Query Database	346
Monitor Counter	350
Get Counter Value	351

Modify Counter	351
Invoke Web Services	352
Format Date/Time	356
Generate Random Text	359
Map Network Path	360
Disconnect Network Path	361
Get Dial-up Status	362
Connect/Disconnect Dial-up	363
Text File Management	364
Append Line	365
Delete Line	366
Find Text	368
Get Lines	370
Insert Line	372
Read Line	373
Search and Replace Text	375
Runbook Control	377
Invoke Runbook	377
Initialize Data	379
Junction	380
Return Data	381
Service Reporting in System Center 2012 R2	382
Getting Started with Service Reporting	382
What's New in System Center 2012 R2 - Service Reporting	383
Overview of Service Reporting	384
Support for Service Reporting	385

Release Notes for System Center 2012 R2 - Service Reporting	386
System Requirements for Service Reporting	387
Troubleshooting Service Reporting	392
Planning for Service Reporting	393
Preparing Windows Azure Pack and System Center Components for Service Reporting	394
Deploying Service Reporting	400
How to Install Service Reporting	401
How to Uninstall Service Reporting	404
How to Configure Service Reporting for Windows Azure Pack and System Center	405
Upgrading Service Reporting	407
Operating Service Reporting	409
Monitoring Service Reporting by Using Operations Manager	410
Understanding Inventory and Usage Data from Windows Azure Pack and System Center	411
About Service Reporting Data Aggregation and Data Cubes	412
About VMM Monitored Data from Operations Manager	413
About Windows Azure Pack Monitored Data	420
Using Service Reporting Usage Data and Inventory Reports	422
List of Usage Data and Inventory Reports in Service Reporting	423
How to Configure the Connection Information in a Report	423
How to View and Analyze Usage and Inventory Data in Reports	424
How to Back Up and Restore Service Reporting Databases	425
Service Management Automation	427
Overview of Service Management Automation	427
Architecture of Service Management Automation	428
Deploy Service Management Automation	430
System requirements for Service Management Automation	430

How to install the Service Management Automation web service	433
How to install the Service Management Automation runbook worker	435
How to install the Service Management Automation PowerShell module	436
Install Service Management Automation from a Command Prompt window	437
Post-installation tasks for Service Management Automation	. 441
How to uninstall Service Management Automation	. 441
Administer Service Management Automation	442
Establish trust between Service Management Automation and Service Provider Foundation	443
Scaling Service Management Automation up or down	444
How to purge the Service Management Automation database	444
Extending Service Management Automation with runbooks	445
Authoring Runbooks in Service Management Automation	446
Runbook Concepts	446
Runbook and Module Operations	453
Global Resources	462
Service Management Automation system runbooks	471
Service Management Automation sample runbooks	471
Service Provider Foundation	474
Architecture Overview of Service Provider Foundation	475
Deploying Service Provider Foundation	477
System Requirements for Service Provider Foundation for System Center 2012 SP1	478
Security Planning for Service Provider Foundation	481
How to Install Service Provider Foundation for System Center 2012 SP1	482
Setup Command-Line Options for Service Provider Foundation	485
Post-Installation Tasks for Service Provider Foundation	489
How to Uninstall Service Provider Foundation	490

Release Notes for Service Provider Foundation for System Center 2012 SP1 491	
Administering Service Provider Foundation	
Manage Certificates and User Roles in Service Provider Foundation	
Walkthrough: Creating a Certificate and User Roles for Service Provider Foundation 496	
Recommended Administrator Capabilities in Service Provider Foundation	
Configuring Portals for Service Provider Foundation	
Privacy Statement for System Center 2012 - Orchestrator	
Release Notes for System Center 2012 - Orchestrator	
Release Notes for System Center 2012 - Orchestrator	
Release Notes for Orchestrator in System Center 2012 SP1	

Getting Started with System Center 2012 - Orchestrator

Orchestrator provides a workflow management solution for the data center. Orchestrator lets you automate the creation, monitoring, and deployment of resources in your environment. This guide describes the architecture of Orchestrator and includes definitions of key terms and concepts and information about where to find additional resources. After reading this guide, you should have a basic understanding of how Orchestrator works and where you can find more information.

Getting Started topics

What's New in System Center 2012 R2 Orchestrator

Provides information about new features and integration packs for System Center 2012 R2 Orchestrator.

- What's New in Orchestrator in System Center 2012 Service Pack 1
 Provides information about new features and integration packs for Orchestrator.
- Orchestrator Capabilities

Describes the features available in Orchestrator.

• Orchestrator Architecture

Describes the architecture of a basic Orchestrator deployment.

Orchestrator Resources

Provides additional resources to help you use Orchestrator.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Administering System Center 2012 Orchestrator
- Integration Packs for System Center 2012 Orchestrator Release Candidate
- Using Runbooks in System Center 2012 Orchestrator
- <u>Using the Orchestration Console in System Center 2012 Orchestrator</u>
- Runbook Activity Reference for System Center 2012 Orchestrator

What's New in System Center 2012 R2 Orchestrator

The following new features are available in System Center 2012 R2 Orchestrator.

What's New

The following new features are added in this release:

- You can install the Service Management Automation web service and up to three runbook workers from System Center 2012 R2 Orchestrator Setup program. These can be used as part of the Windows Azure Pack for Windows Server configuration or to enable you to run runbooks and perform other automation tasks using Windows PowerShell cmdlets. For evaluation purposes, you should install a single runbook worker on the same computer as the web service.
- Windows Server 2012 R2 is supported in this release.

In addition, in System Center 2012 R2, Orchestrator has the following new and updated integration Packs (IPs):

New in System Center 2012 R2

System Center Integration Pack for Microsoft SharePoint

Updated in System Center 2012 R2

Windows Azure Integration Pack for Orchestrator in System Center 2012 SP1 and System Center 2012 R2

System Center Integration Pack for System Center 2012 Virtual Machine Manager

What's New in Orchestrator in System Center 2012 Service Pack 1

The following are the new features for Orchestrator in System Center 2012 Service Pack 1 (SP1).

What's New

In System Center 2012 Service Pack 1 (SP1), Orchestrator has the following new and updated integration Packs (IPs).

New in System Center 2012 SP1

Exchange Administrator Integration Pack for Orchestrator in System Center 2012 SP1

Exchange Users Integration Pack for Orchestrator in System Center 2012 SP1

Representational State Transfer (REST)

Integration Pack Guide for Orchestrator in System Center 2012 SP1

Updated in System Center 2012 SP1

Active Directory Integration Pack for System Center 2012 - Orchestrator

HP Service Manager Integration Pack for System Center 2012 - Orchestrator

System Center Integration Pack for System Center 2012 Operations Manager

System Center Integration Pack for System Center 2012 Virtual Machine Manager

VMware vSphere Integration Pack for System Center 2012 - Orchestrator

Orchestrator Capabilities

IT administrators perform many tasks and procedures to keep the health of their computing environment up-to-date and their business running. Tasks might include the following diverse activities, for example, new employees require that accounts and resources are configured, a business acquisition requires integrating a system from another vendor, and new hardware requires provisioning. Individual tasks and subtasks are automated, but typically, not the whole process. In addition, the administrators must maintain quality standards and system efficiency. System Center 2012 - Orchestrator can tie disparate tasks and procedures together by using the graphical user-interface Runbook Designer to create reliable, flexible, and efficient end-to-end solutions in the IT environment.

By using Orchestrator, you can carry out the following tasks:

- Automate processes in your data center, regardless of hardware or platform.
- Automate your IT operations and standardize best practices to improve operational efficiency.
- Connect different systems from different vendors without having to know how to use scripting and programming languages.

Custom automation

Orchestrator provides tools to build, test, debug, deploy, and manage automation in your environment. These automated procedures, called runbooks, can function independently or start other runbooks. The standard activities defined in every installation of Orchestrator provide a variety of monitors, tasks, and runbook controls with which you can integrate a wide range of system processes. Each activity in a runbook publishes data that is available to any subsequent

activity in that runbook. You use this Published Data to provide dynamic, decision-making capabilities, which can include creating emails, alerts, log files, accounts, and more.

Your IT organization can use Orchestrator to improve efficiency and reduce operational costs to support cross-departmental objectives. Orchestrator provides an environment with shared access to common data. By using Orchestrator, you can evolve and automate key processes between groups and consolidate repetitive manual tasks. You can automate cross-functional team processes and enforce best practices for incident, change, and service management by creating runbooks that are customized for your requirements. Through automation, regularly recurring tasks reduce the number of manual and error-prone activities in your environment. Orchestrator helps you improve the reliability and predictability of your IT procedures.

Cross-platform integration

Orchestrator integrates with System Center, other Microsoft products, and non-Microsoft products to enable interoperability across the data center. Orchestrator improves efficiency across multiple tools, systems, and departments by eliminating or crossing technology and organizational process structures. You can extend the capabilities of Orchestrator with integration packs that include additional functionality for both Microsoft and non-Microsoft products and technologies. Orchestrator activities and integration packs reduce unanticipated errors and shorten service delivery time by automating the common tasks associated with enterprise tools and products.

End-to-end orchestration

Orchestration is the collective name for the automated arrangement, coordination, and management of systems, software, and practices. It enables the management of complex cross-domain processes. Orchestrator provides the tools for orchestration to combine software, hardware, and manual processes into a seamless system. These tools let you connect and automate workflows.

Just as manufacturing companies have automated common and repeatable tasks from their production processes, you can adopt this same efficiency in the IT environment by using Orchestrator to seamlessly perform and monitor your IT processes. Orchestrator can handle routine tasks, process enforcement, and reliably meet the demands of the largest enterprises. Orchestrator integrates seamlessly with other System Center products to integrate IT administrative tasks from start to finish.

Extensible structure

If you have a custom in-house solution, Orchestrator provides extensible integration to any system through the Orchestrator Integration Toolkit. You can create custom integrations that allow Orchestrator to connect to any environment.

Orchestrator uses a Representational State Transfer (REST)-based web service that can perform processes like start and stop runbook jobs and get reporting information in Open Data protocol (OData) format. The web service lets you develop applications that can use live data from Orchestrator.

See Also

Getting Started with System Center 2012 - Orchestrator

Orchestrator Architecture

This topic provides an overview of System Center 2012 - Orchestrator, including a description of the system architecture, the internals of a typical runbook workflow, and the flow of a deployed runbook.

Orchestrator deployment basics

The following table lists the features in a basic deployment of Orchestrator.

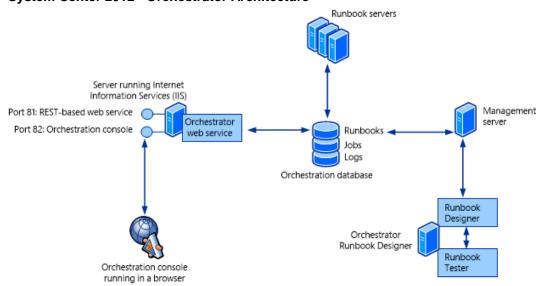
Orchestrator feature	Description
management server	The management server is the communication layer between the Runbook Designer and the orchestration database.
runbook server	A runbook server is where an instance of a runbook runs. Runbook servers communicate directly with the orchestration database. You can deploy multiple runbook servers per Orchestrator installation to increase capacity and redundancy.
orchestration database	The database is a Microsoft SQL Server database that contains all of the deployed runbooks, the status of running runbooks, log files, and configuration data for Orchestrator.
Runbook Designer	The Runbook Designer is the tool used to build, edit, and manage Orchestrator runbooks. For more information about the Runbook Designer, see <u>Using Runbooks in System Center 2012 - Orchestrator</u> .
Runbook Tester	Runbook Tester is a run-time tool used to test runbooks developed in the Runbook Designer. For more information about Runbook Tester, see How to Test a Runbook in <u>Using Runbooks in System Center 2012 - Orchestrator</u> .
Orchestration console	The Orchestration console lets you start or stop runbooks and view real-time status on a web

Orchestrator feature	Description
	browser. For more information about using the Orchestration console, see <u>Using the Orchestration Console in System Center 2012 - Orchestrator</u> .
Orchestrator web service	The Orchestrator web service is a Representational State Transfer (REST)-based service that enables custom applications to connect to Orchestrator to start and stop runbooks, and retrieve information about operations by using custom applications or scripts. The Orchestration console uses this web service to interact with Orchestrator.
Deployment Manager	Deployment Manager is a tool used to deploy integration packs (IPs), runbook servers, and Runbook Designers. For more information about this tool, see Deploying System Center 2012 - Orchestrator .

Architectural diagram

The following diagram illustrates each of the Orchestrator features and the communication between each.

System Center 2012 - Orchestrator Architecture



The orchestration database is the center of the Orchestrator installation containing all runbooks, configuration settings, and logs. The management server is required as a communication layer between the Runbook Designer and the orchestration database. One or more runbook servers communicate directly with the database to retrieve runbooks to run and store information about the jobs created from the runbooks. The web service also communicates directly with the orchestration database and provides a web browser connection for the Orchestration console.

Orchestrator Extensions

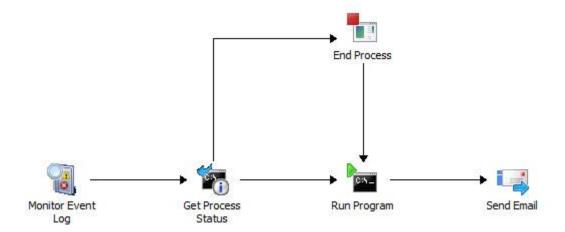
The following table shows multiple strategies available for extending the functionality provided by a standard installation of Orchestrator. For additional information, see Deploying System Center 2012 - Orchestrator.

Orchestrator feature	Description
integration pack (IP)	An integration pack is a collection of custom activities specific to a product or technology. Microsoft and other companies provide integration packs with activities to interact with their product from an Orchestrator runbook.
Orchestrator Integration Toolkit	The Orchestrator Integration Toolkit lets you extend your library of activities beyond the collection of standard activities and integration packs. The Integration Toolkit has wizard-based tools to create new activities and integration packs for Orchestrator. Developers can also use the Integration Toolkit to create integration packs from custom activities that they build by using the Orchestrator SDK.

Automation by using runbooks

To automate a task or process in Orchestrator, you use the Runbook Designer to create a runbook. You add activities to the runbook by dragging them from the **Activities** pane, and then link activities in the required order to create a workflow.

The following illustration shows a simple runbook.



This runbook monitors an event log. When it detects the specified event, the runbook checks the status of a particular process in Windows on a specific computer. If the process is found to be running, it is stopped. The runbook then starts the process and sends an email as a notification of the change of process state.

Each runbook activity finishes before proceeding to the next, and activities are available that provide complex logic such as requiring that multiple activities are completed before the runbook proceeds. By using a combination of logic on activities and smart links, you can implement whatever logic your particular automation scenario requires.

How Orchestrator processes a Runbook

After you have created a runbook, you commit it to the orchestration database by checking it in. You can then use either the Runbook Designer or the Orchestration console to start and stop the runbook.

A request to run a runbook creates a *job* that is stored in the orchestration database. Each runbook can define a primary runbook server and one or more standbys that process the runbook if the primary is unavailable. A service on each runbook server continuously monitors the orchestration database for jobs that it can process. When a runbook server detects a job, it logs that it is working on the job, copies the runbook locally, logs that it is running an instance of the runbook, and then begins processing the runbook. For any runbook not containing a monitor, you can create multiple runbook requests meaning that a single runbook can have multiple jobs.

When a runbook server processes a job, it creates an *instance* of the runbook by making a copy of it locally, and then performing the actions defined within the runbook according to the included workflow logic. Status information, activity results, and data are recorded in the orchestration database so that you can monitor the real-time and historical status of the runbook.

Permissions

Access to Orchestrator is provided by adding user accounts to a security group that is created during installation. This group can either be a domain group or a local group on the management server. Users of this group have full access to the Runbook Designer to create and modify runbooks and the Deployment Manager to deploy new Runbook Designers and runbook servers. Operators who have to start and stop runbooks but not create them can be granted this permission to individual runbooks and then use the Orchestration console.

Orchestrator Terminology

This topic provides terms and definition for System Center 2012 - Orchestrator and shows changes of Opalis Integration Server 6.3 terms to Orchestrator terminology and their definitions.

Terms and definitions

- Glossary for System Center 2012 Orchestrator
 Provides definitions for common terminology used in Orchestrator.
- Glossary for Opalis Integration Server 6.3

Provides definitions for common terminology used in Opalis Integration Server 6.3. Identifies changes between Opalis Integration Server 6.3 and Orchestrator.

See Also

Getting Started with System Center 2012 - Orchestrator

Glossary for System Center 2012 - Orchestrator

Term	Definition
activity	A single task in a runbook that performs a specific function.
check in	To save the changes in a runbook to the database.
check out	To allow edits to a runbook.
counter	A global integer variable that is used in a runbook.
data bus	A mechanism in Orchestrator that passes information from one activity in a runbook

Term	Definition
	to another activity.
instance	A unique occurrence of a runbook that is running on a runbook server.
integration pack	A collection of custom activities that is specific to a product or a technology.
IP	See Other Term: integration pack
job	A request to run a runbook.
junction	A runbook activity that synchronizes multiple branches of a runbook.
management server	The communication layer between the Runbook Designer and the deployment manager to the database.
monitor	An activity that continuously runs and that initiates a runbook when the monitor matches the criteria that you specify.
OIT	See Other Term: Orchestrator Integration Toolkit
Orchestration console	A web-based console that you can use to start, stop, and view information about runbooks.
orchestration database	The Oracle or SQL Server database where configuration information, runbooks, and logs are stored.
Orchestrator Integration Toolkit	A set of software tools that you can use to create custom integration packs.
Published Data	The data that is published to the databus from each activity in a runbook.
runbook	The sequence of activities that orchestrate actions on computers and networks.
Runbook Designer	The tool that is used by designers to create, modify, and deploy runbooks.
runbook server	The server that runs the service that manages runbooks and communicates with the orchestration database.
Runbook Tester	The tool that is used to test and validate

Term	Definition
	runbooks.
schedule	The global settings that you can use to define a set of date and time criteria for a runbook.
smart link	The connection between two activities in a runbook.
standard activity	The set of activities that is included with the standard installation of Orchestrator.
subscribe	To request data from the data bus.
variable	A global value that is used to define a frequently used setting, such as a directory path to common files or server names.

Glossary for Opalis Integration Server 6.3

The following table lists Opalis Integration Server 6.3 terms and the Orchestrator terms that replace them. A brief definition is included for each term.

Opalis Integration Server 6.3 term	System Center 2012 - Orchestrator term	Definition
Action server	runbook server	A runbook server is a computer that receives an instance of a runbook and runs the sequence of activities. Runbook servers communicate directly with the orchestration database; they do not require a management server to run runbooks.
Client	Runbook Designer	See definition for Opalis client.
custom start	initialize data	The initial runbook activity defined in a runbook to provide user-defined input parameters for the runbook.
datastore	orchestration database	The orchestration database is a SQL Server database

Opalis Integration Server 6.3 term	System Center 2012 - Orchestrator term	Definition
		containing configuration information, runbooks, and logs for Orchestrator.
foundation object	standard activity	The set of runbook activities available in a default installation. This includes monitors, tasks, and all runbook controls.
object	activity	The tasks used to create a runbook.
Object palette	Activities pane	The Activities pane is located in the tasks pane in the Runbook Designer. Collections of activities are grouped by function or integration pack.
Opalis client	Runbook Designer	An application used to create, modify, and deploy runbooks.
Operator console	Orchestration console	The interface that enables a user to see available runbooks, the real-time status of jobs and running instances, view their status, and start or stop runbooks, jobs, or instances.
Policy	runbook	A runbook is a collection of activities that orchestrates actions, events, and tasks.
Policy folder	runbook folder	A folder that contains one or more runbooks.
policy module	job process	A request to run a specific runbook that is waiting for assignment to a runbook server for processing.
Policy Testing Console	Runbook Tester	The tool used by Runbook Designers to test policies

Opalis Integration Server 6.3 term	System Center 2012 - Orchestrator term	Definition
		before deployment.
publish policy data	Published Data	Published Data is a runbook activity used to publish data from the runbook back to a calling (parent) runbook.
request	job	A job is a request to deploy and run a runbook on a runbook server. Jobs are stored in the orchestration database queue.
trigger policy	Invoke Runbook	An Invoke Runbook activity calls another runbook from within a runbook. The Invoke Runbook activity can optionally wait for the called runbook to finish before proceeding. Data is returned from the invoked runbook by using the Returned Data activity. It is equivalent to the function call found in many programming languages.
workflow control	runbook control	A collection of standard activities that manage how runbook logic behaves.

Orchestrator Resources

In addition to this online reference for System Center 2012 - Orchestrator, there are a number of resources that can provide additional information about building runbooks, by using System Center 2012 - Orchestrator SDK and applying best practices.

Resource	Location
System Center 2012 - Orchestrator	http://www.microsoft.com/systemcenter/orchestrator
Home	

Resource	Location
System Center Home on TechNet	http://technet.microsoft.com/systemcenter/
Orchestrator Team Blog on TechNet	http://blogs.technet.com/b/orchestrator/
Orchestrator Community Releases on CodePlex	http://orchestrator.codeplex.com
Orchestrator Community Forums on TechNet	http://social.technet.microsoft.com/Forums/category/systemcenterorchestrator

See Also

Getting Started with System Center 2012 - Orchestrator

Upgrading System Center 2012 SP1 Orchestrator to System Center 2012 R2

This guide will show you how to upgrade from System Center 2012 Service Pack 1 (SP1) Orchestrator to System Center 2012 R2 Orchestrator.



Warning

If you are planning to upgrade two or more System Center components, it is important to start by reading the Upgrade Sequencing for System Center 2012 R2 topic.

The order in which you perform component upgrades is important. Failure to follow the correct upgrade sequence might result in component failure for which no recovery options exist. The affected System Center components are:

- 1. Orchestrator
- 2. Service Manager
- 3. Data Protection Manager (DPM)
- 4. Operations Manager
- Configuration Manager
- 6. Virtual Machine Manager
- 7. App Controller



Because your data center must keep running while you upgrade System Center 2012 components one at a time, after you have upgraded the Orchestrator servers to System Center 2012 SP1, you can run:

- A System Center 2012 integration pack on a System Center 2012 component.
- A System Center 2012 SP1 integration pack on a System Center 2012 SP1 component.
- A System Center 2012 SP1 integration pack on a System Center 2012 R2 component (except for Virtual Machine Manager).
- A System Center 2012 R2 integration pack on a System Center 2012 R2 component (Virtual Machine Manager).

No other configurations are supported.

See Also

<u>Tasks to Perform Before You Begin the Upgrade</u>

<u>How to Upgrade System Center 2012 SP1 Orchestrator to System Center 2012 R2</u>

Troubleshoot Your Orchestrator Installation

Tasks to Perform Before You Begin the Upgrade

Before you can upgrade Orchestrator to System Center 2012 R2, you must prepare the environment by performing the following tasks:

- 1. Complete all runbooks running in the current Orchestrator installation. For information about stopping runbooks, see the <u>Running Runbooks</u> topic in the <u>Orchestrator</u> library on TechNet.
- Close any open programs and ensure that there are no pending restarts on the computer. For
 example, if you have installed a server role by using Server Manager or have applied a
 security update, you might have to restart the computer, and then log on to the computer with
 the same user account to finish the installation of the server role or the security update.
- Perform a full backup of the Orchestrator database. For information about backing up the
 Orchestrator database, see the <u>How to Back up Orchestrator</u> topic in the <u>Orchestrator</u> library
 on TechNet. You can also use tools provided by SQL Server to back up the VMM database.
 For more information, see <u>Back Up and Restore of SQL Server Databases</u>.
- 4. Upgrade the hardware, operating system, and other software if necessary to meet the requirements of Orchestrator in System Center 2012 R2.

See Also

Upgrading System Center 2012 SP1 Orchestrator to System Center 2012 R2

How to Upgrade System Center 2012 SP1 Orchestrator to System Center 2012 R2

When you upgrade a server that runs System Center 2012 R2 Orchestrator, all features that are installed on the server are upgraded. Before you begin the upgrade process, make sure that your server meets the minimum supported configurations. For more information, see the System Requirements topic in the Orchestrator library on TechNet.

To upgrade Orchestrator to System Center 2012 R2

- 1. Stop all Orchestrator runbooks.
- 2. Uninstall the Orchestrator management server, any runbook servers, the Web Service, and the Runbook Designer.
- 3. Install the Orchestrator management server in System Center 2012 R2, as described in the <u>Deployment Guide</u> (http://go.microsoft.com/fwlink/?LinkID=232709).
- 4. Install any Orchestrator runbook servers in System Center 2012 R2.
- 5. Install the Orchestrator Runbook Designer in System Center 2012 R2.
- 6. If needed, install the Orchestrator Web Service in System Center 2012 R2.
- 7. Take the Orchestrator servers out of maintenance mode.

Upgrading the integration pack for another System Center 2012 R2 component

The only System Center 2012 R2 component for which an updated integration pack is being released for System Center 2012 R2 is Virtual Machine Manager.

For more detailed instructions for upgrading System Center 2012 R2 components, see the guide "Upgrade Sequencing for System Center 2012 R2" that is included in the SC2012R2Upgrade.zip file that you downloaded from the Microsoft Connect website. For each component you will perform the following steps.

- 1. Uninstall and unregister the integration pack for the component according to the <u>How to</u> Uninstall and Unregister an Integration Pack.
 - When you install an upgrade of an integration pack, you must first uninstall any earlier version of the integration pack from all runbook servers and Runbook Designers. You then register and deploy the upgrade of the integration pack. If you do not uninstall the previous version of the integration pack prior to registering and deploying the upgrade version, the upgrade version will fail.
- 2. Upgrade the component.
- 3. Install and register the System Center 2012 R2 integration pack for the component.
- 4. Verify that Orchestrator is receiving data from the component.

Troubleshoot Your Orchestrator Installation

The latest troubleshooting information for System Center 2012 - Orchestrator is available in the release notes under the Release Notes for System Center 2012 - Orchestrator topic in the Orchestrator library on TechNet. The following information provides additional instructions and caveats that you can use during installation to resolve problems you might experience.

Orchestrator log files

If you experience problems during installation, installation log files are located in the folder C:\Users\%USERNAME%\AppData\Local\SCO\LOGS.

If you experience problems when you are running Orchestrator, the product log files are located in the folder C:\ProgramData\Microsoft System Center 2012\Orchestrator\.

Windows Firewall

When you deploy additional Runbook Designer applications to your environment, you might see a failed installation message. To correctly install the Runbook Designer, enable the following firewall rules as they apply to your operating system and deployment configuration.

Windows Firewall with Advanced Security for Windows Server 2012 R2

By default, **Windows Firewall with Advanced Security** is enabled on all Windows Server 2012 R2 computers, and blocks all incoming traffic unless it is a response to a request by the host, or it is specifically allowed. You can explicitly allow traffic by specifying a port number, application name, service name, or other criteria by configuring Windows Firewall with Advanced Security settings.

If you are running Windows Server 2012 R2, enable the following rules to allow all Monitor Event activities to function correctly:

- Windows Management Instrumentation (Async-In)
- Windows Management Instrumentation (DCOM-In)
- Windows Management Instrumentation (WMI-In)

Automated deployment

When a runbook server or Runbook Designer is installed behind a firewall, specific firewall rules are required between the remote computers that are used to deploy the runbook server and Runbook Designer. An additional rule is required for the remote connection between the Runbook Designer and the runbook server to allow the Orchestrator management service to accept remote connections. If you are using the **Monitor WMI** task, the runbook server requires a special firewall rule on the computer that uses PolicyModule.exe.

Enable the following firewall rules on your computer:

Firewall rule between the Runbook Designer and the Orchestrator management server

Operating system	Firewall rule
64-bit	%ProgramFiles (x86)%\Microsoft System Center 2012 R2\Orchestrator\Management Server\OrchestratorManagementService.exe
32-bit	%ProgramFiles%Microsoft System Center 2012 R2\Orchestrator\Management Server\OrchestratorManagementService.exe

Firewall rules between remote computers

Operating system	Firewall rules
Windows Server 2012 R2	 File and Printer Sharing Windows Management Instrumentation (WMI)
	 Program rule for OrchestratorRemotingService to accept remote connections. This rule must be enabled through the Advanced Firewall mode:
	 %SystemRoot%\SysWOW64\OrchestratorRemotingService.exe (for a 64-bit operating system)
	 %SystemRoot%\System32\OrchestratorRemotingService.exe (for a 32-bit operating system)

Firewall rules between the runbook server and the computer that uses PolicyModule.exe

Operating system	Firewall rule
64-bit	%ProgramFiles (x86)%\Microsoft System Center 2012 R2\Orchestrator\Runbook Server\PolicyModule.exe
32-bit	%ProgramFiles\Microsoft System Center 2012 R2\Orchestrator\Runbook Server\PolicyModule.exe

For more information about adding firewall rules, see Add or Edit a Firewall Rule.

RunbookService fails to start after computer reboot

When you reboot your runbook server, the RunbookService attempts to connect to the orchestration database. If the database is not available, the RunbookService fails. The event log message is **This computer was unable to communicate with the computer providing the**

server.. Typically, this can occur when the SQL server and the runbook server are installed on the same computer.

To solve this problem, you can manually start the RunbookService, or configure the RunbookService to make multiple attempts during startup to connect to database before failing.

Cannot restart runbook service if you uninstall with an account without administrator permissions

If you attempt to uninstall Orchestrator while logged in with an account that is a member of OrchestratorSystemGroup but is not an administrator, uninstall removes all accounts from OrchestratorSystemGroup. If you stop the runbook service and attempt to restart the service, the service fails because the user account does not have the correct permissions to retrieve the orchestration database connection. An account that is an administrator or a member of the OrchestratorSystemGroup is required to retrieve the orchestration database connection.

To solve this problem, an administrator can add the user back to OrchestratorSystemGroup.

Other resources for this product

- TechNet Library main page for Orchestrator
- Deploying System Center 2012 Orchestrator in the Orchestrator library on TechNet.

How to Uninstall and Unregister an Integration Pack

For instructions on how to install an integration pack, see the How to Install an Integration Pack topic in the Orchestrator library on TechNet.

Important

When you install an upgrade of an integration pack, you must first uninstall any earlier version of the integration pack from all runbook servers and Runbook Designers. You then register and deploy the upgrade of the integration pack. If you do not uninstall the previous version of the integration pack prior to registering and deploying the upgrade version, the upgrade version will fail.

To uninstall an integration pack

- 1. Open **Programs and Features** in Windows Control Panel.
- 2. Right-click the integration pack and click Uninstall.

To unregister an integration pack

- 1. Start the **Deployment Manager**.
- 2. In the navigation pane of the Deployment Manager, click Integration Packs.
- 3. Right-click the integration pack and click Unregister Integration Pack or Hotfix from

See Also

Upgrading System Center 2012 SP1 Orchestrator to System Center 2012 R2

Upgrading System Center 2012 -Orchestrator to System Center 2012 SP1

This guide will show you how to upgrade from System Center 2012 - Orchestrator to Orchestrator in System Center 2012 Service Pack 1 (SP1).



Warning

If you are planning to upgrade two or more System Center components, it is important to start by reading the guide Upgrade Sequencing for System Center 2012 SP1. The order in which you perform component upgrades is important. Failure to follow the correct upgrade sequence might result in component failure for which no recovery options exist. The affected System Center components are:

- 1. Orchestrator
- Service Manager
- 3. Data Protection Manager (DPM)
- 4. Operations Manager
- 5. Configuration Manager
- 6. Virtual Machine Manager
- 7. App Controller



Because your data center must keep running while you upgrade System Center 2012 components one at a time, after you have upgraded the Orchestrator servers to System Center 2012 SP1, you can run:

- A System Center 2012 integration pack on a System Center 2012 component.
- A System Center 2012 SP1 integration pack on a System Center 2012 SP1 component. No other configurations are supported.

See Also

Tasks to Perform Before You Begin the Upgrade

How to Upgrade System Center 2012 – Orchestrator to System Center 2012 SP1

Troubleshoot Your Orchestrator Installation

How to Uninstall and Unregister an Integration Pack

Tasks to Perform Before You Begin the Upgrade

Before you can upgrade Orchestrator to System Center 2012 Service Pack 1 (SP1), you must prepare the environment by performing the following tasks:

- 1. Complete all runbooks running in the current Orchestrator installation. For information about stopping runbooks, see the Running Runbooks topic in the Orchestrator library on TechNet.
- 2. Close any open programs and ensure that there are no pending restarts on the computer. For example, if you have installed a server role by using Service Manager or have applied a security update, you might have to restart the computer, and then log on to the computer with the same user account to finish the installation of the server role or the security update.
- Perform a full backup of the Orchestrator database. For information about backing up the App Controller database, see the <u>How to Back up Orchestrator</u> topic in the <u>Orchestrator</u> library on TechNet. You can also use tools provided by SQL Server to back up the VMM database. For more information, see <u>Back Up and Restore of SQL Server Databases</u>.
- 4. Upgrade the hardware, operating system, and other software if necessary to meet the requirements of Orchestrator in System Center 2012 SP1.

See Also

<u>Upgrading System Center 2012 - Orchestrator to System Center 2012 SP1</u>

How to Upgrade System Center 2012 – Orchestrator to System Center 2012 SP1

When you upgrade a server that runs System Center 2012 - Orchestrator, all features that are installed on the server are upgraded. Before you begin the upgrade process, make sure that your server meets the minimum supported configurations. For more information, see System Requirements.

To upgrade Orchestrator to System Center 2012 SP1

- 1. Put all of the Orchestrator servers in maintenance mode.
- 2. Uninstall the Orchestrator management server, any runbook servers, the Web Service, and the Runbook Designer.
- 3. Install the Orchestrator management server in System Center 2012 SP1, as described in the <u>Deployment Guide</u> (http://go.microsoft.com/fwlink/?LinkID=232709).
- 4. Install any Orchestrator runbook servers in System Center 2012 SP1.
- 5. Install the Orchestrator Runbook Designer in System Center 2012 SP1.
- 6. If needed, install the Orchestrator Web Service in System Center 2012 SP1.
- 7. Take the Orchestrator servers out of maintenance mode.
- 8. Return to the Upgrade Sequencing Guide.

Upgrading other System Center 2012 SP1 components

For more detailed instructions for upgrading System Center 2012 SP1 components, see the guide **Upgrade Sequencing for System Center 2012 SP1**. Those instructions for each component follow the following general outline:

- 1. Uninstall and unregister the integration pack for the component according to How to Uninstall and Unregister an Integration Pack.
 - When you install an upgrade of an integration pack, you must first uninstall any earlier version of the integration pack from all runbook servers and Runbook Designers. You then register and deploy the upgrade of the integration pack. If you do not uninstall the previous version of the integration pack prior to registering and deploying the upgrade version, the upgrade version will fail.
- 2. Upgrade the component.
- 3. Install and register the System Center 2012 SP1 integration pack for the component.
- 4. Verify that Orchestrator is receiving data from the component.

Troubleshoot Your Orchestrator Installation

The latest troubleshooting information for System Center 2012 - Orchestrator is available in the release notes at Release Notes for System Center 2012 - Orchestrator. The following information provides additional instructions and caveats that you can use during installation to resolve problems you might experience.

Orchestrator log files

If you experience problems during installation, installation log files are located in the folder C:\Users\%USERNAME%\AppData\Local\SCO\LOGS.

If you experience problems when you are running Orchestrator, the product log files are located in the folder C:\ProgramData\Microsoft System Center 2012\Orchestrator\.

Windows Firewall

When you deploy additional Runbook Designer applications to your environment, you might see a failed installation message. To correctly install the Runbook Designer, enable the following firewall rules as they apply to your operating system and deployment configuration.

Windows Firewall with Advanced Security for Windows Server 2012 R2

By default, **Windows Firewall with Advanced Security** is enabled on all Windows Server 2008 R2 computers, and blocks all incoming traffic unless it is a response to a request by the host, or it is specifically allowed. You can explicitly allow traffic by specifying a port number, application name, service name, or other criteria by configuring Windows Firewall with Advanced Security settings.

If you are running Windows Server 2012 R2, enable the following rules to allow all Monitor Event activities to function correctly:

- Windows Management Instrumentation (Async-In)
- Windows Management Instrumentation (DCOM-In)
- Windows Management Instrumentation (WMI-In)

Automated deployment

When a runbook server or Runbook Designer is installed behind a firewall, specific firewall rules are required between the remote computers that are used to deploy the runbook server and Runbook Designer. An additional rule is required for the remote connection between the Runbook Designer and the runbook server to allow the Orchestrator management service to accept remote connections. If you are using the **Monitor WMI** task, the runbook server requires a special firewall rule on the computer that uses PolicyModule.exe.

Enable the following firewall rules on your computer:

Firewall rule between the Runbook Designer and the Orchestrator management server

Operating system	Firewall rule
64-bit	%ProgramFiles (x86)%\Microsoft System Center 2012 R2\Orchestrator\Management Server\OrchestratorManagementService.exe
32-bit	%ProgramFiles%Microsoft System Center 2012 R2\Orchestrator\Management Server\OrchestratorManagementService.exe

Firewall rules between remote computers

Operating system	Firewall rules
Windows Server 2008 R2	 File and Printer Sharing Windows Management Instrumentation (WMI)
	 Program rule for OrchestratorRemotingService to accept remote connections. This rule must be enabled through the Advanced Firewall mode:
	 %SystemRoot%\SysWOW64\OrchestratorRemotingService.exe (for a 64-bit operating system)
	 %SystemRoot%\System32\OrchestratorRemotingService.exe (for a 32-bit operating system)

Firewall rules between the runbook server and the computer that uses PolicyModule.exe

Operating system	Firewall rule
64-bit	%ProgramFiles (x86)%\Microsoft System Center 2012 R2\Orchestrator\Runbook Server\PolicyModule.exe
32-bit	%ProgramFiles\Microsoft System Center 2012 R2\Orchestrator\Runbook Server\PolicyModule.exe

For more information about adding firewall rules, see Add or Edit a Firewall Rule.

RunbookService fails to start after computer reboot

When you reboot your runbook server, the RunbookService attempts to connect to the orchestration database. If the database is not available, the RunbookService fails. The event log message is **This computer was unable to communicate with the computer providing the server.** Typically, this can occur when the SQL server and the runbook server are installed on the same computer.

To solve this problem, you can manually start the RunbookService, or configure the RunbookService to make multiple attempts during startup to connect to database before failing.

Cannot restart runbook service if you uninstall with an account without administrator permissions

If you attempt to uninstall Orchestrator while logged in with an account that is a member of OrchestratorSystemGroup but is not an administrator, uninstall removes all accounts from OrchestratorSystemGroup. If you stop the runbook service and attempt to restart the service, the services fails because the user account does not have the correct permissions to retrieve the orchestration database connection. An account that is an administrator or a member of the OrchestratorSystemGroup is required to retrieve the orchestration database connection.

To solve this problem, an administrator can add the user back to OrchestratorSystemGroup.

HTTP errors when starting the Orchestration console

If you get HTTP errors when starting the Orchestration console, try the following mitigations:

- HTTP error 404.3 Not Found: Ensure that both .NET Framework 4.5 and Windows
 Communication Foundation (WCF) HTTP Activation are installed. You can find the
 instructions in <u>Single-Computer Requirements for System Center 2012 R2 Orchestrator</u>.
- HTTP error 404.17 Not Found: Ensure that Windows Communication Foundation (WCF)
 HTTP Activation is installed. You can find the instructions in <u>Single-Computer Requirements</u>
 for System Center 2012 R2 Orchestrator.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Deployment Overview
- Plan Your Orchestrator Deployment
- Install Orchestrator
- Perform Post-Installation Tasks

How to Uninstall and Unregister an Integration Pack

For instructions on how to install an integration pack, see the <u>How to Install an Integration Pack</u> topic in the <u>Orchestrator</u> library on TechNet.

Important

When you install an upgrade of an integration pack, you must first uninstall any earlier version of the integration pack from all runbook servers and Runbook Designers. You then register and deploy the upgrade of the integration pack. If you do not uninstall the previous version of the integration pack prior to registering and deploying the upgrade version, the upgrade version will fail.

To uninstall an integration pack

- 1. Open Programs and Features in Windows Control Panel.
- 2. Right-click the integration pack and click Uninstall.

To unregister an integration pack

- 1. Start the **Deployment Manager**.
- 2. In the navigation pane of the Deployment Manager, click Integration Packs.
- 3. Right-click the integration pack and click **Unregister Integration Pack or Hotfix from the Orchestrator Management Server**.

See Also

<u>Tasks to Perform Before You Begin the Upgrade</u>

<u>Upgrading System Center 2012 - Orchestrator to System Center 2012 SP1</u>

<u>Troubleshoot Your Orchestrator Installation</u>

Deploying System Center 2012 - Orchestrator

System Center 2012 - Orchestrator is a workflow management solution for the data center. It enables you to automate the creation, monitoring, and deployment of resources in your environment. This document describes System Center 2012 - Orchestrator planning and deployment.

Deployment topics

• Deployment Overview

Provides a brief overview of the steps to deploy Orchestrator.

Plan Your Orchestrator Deployment

Provides planning guidelines and best practices for your Orchestrator deployment.

• Install Orchestrator

Provides step-by-step instructions to install Orchestrator.

Perform Post-Installation Tasks

Describes required and optional post-installation tasks.

Troubleshoot Your Orchestrator Installation

Provides guidance on common installation issues.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Getting Started with System Center 2012 Orchestrator
- Administering System Center 2012 Orchestrator
- Integration Packs for System Center 2012 Orchestrator Release Candidate
- Using Runbooks in System Center 2012 Orchestrator
- Using the Orchestration Console in System Center 2012 Orchestrator
- Runbook Activity Reference for System Center 2012 Orchestrator

Deployment Overview

The procedures in the following sections describe how to plan your deployment and install System Center 2012 - Orchestrator.

Use the following steps to install Orchestrator.

Task	Information
Step 1: Plan your deployment.	Plan Your Orchestrator Deployment
Step 2: Review the system prerequisites.	System Requirements

Task	Information
Step 3: Install Orchestrator.	Install Orchestrator
Step 4: Perform post-installation tasks.	Perform Post-Installation Tasks



This release supports only databases that are compatible with System Center 2012 -Orchestrator. You cannot use the databases from Opalis 6.3 or the System Center 2012 - Orchestrator beta version with this product.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator

Plan Your Orchestrator Deployment

This section describes the planning required before you install System Center 2012 -Orchestrator.

Planning Your Deployment

System Requirements

Describes the hardware, operating system, and software requirements for Orchestrator.

Orchestrator Security Planning

Describes the service accounts and security groups for Orchestrator.

TCP Port Requirements

Describes the TCP port and web service requirements for Orchestrator.

Scale Planning

Provides scale planning for Orchestrator.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- **Deployment Overview**
- Install Orchestrator
- Perform Post-Installation Tasks
- **Troubleshoot Your Orchestrator Installation**

System Requirements

The following sections describe the system requirements for System Center 2012 - Orchestrator, Orchestrator in System Center 2012 Service Pack 1 (SP1), and System Center 2012 R2 Orchestrator, depending on your System Center 2012 version, your particular configuration, and choice of features to install.

System requirements topics

- Single-Computer Requirements for System Center 2012 R2 Orchestrator
 - Describes the minimum hardware and software requirements of a single computer running all Orchestrator features in System Center 2012 R2.
- Single-Computer Requirements for Orchestrator in System Center 2012 SP1
 - Describes the minimum hardware and software requirements of a single computer running all Orchestrator features in System Center 2012.
- Single-Computer Requirements for System Center 2012 Orchestrator
 - Describes the minimum hardware and software requirements of a single computer running all Orchestrator features in System Center 2012 - Orchestrator.
- **Individual Feature Requirements**
 - Describes the minimum hardware and software requirements for each Orchestrator feature.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Plan Your Orchestrator Deployment
- Orchestrator Security Planning
- TCP Port Requirements
- Scale Planning

Single-Computer Requirements for System Center 2012 R2 Orchestrator

This section describes the minimum hardware and software configuration required for a full installation of System Center 2012 R2 Orchestrator on a single computer.



Note

Orchestrator is not supported when installed on the same computer as a domain controller.

Hardware

The following minimum hardware configuration is required for a full installation of Orchestrator:

- Minimum 1 gigabyte (GB) of RAM, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigahertz (GHz) or better

Operating system

The following table lists the supported operating systems for a full installation of Orchestrator on a single computer.

Feature	Operating system
Management server	Windows Server 2008 R2, Windows Server
Orchestrator web service	2012, Windows Server 2012 R2
Runbook Designer	
Runbook server	

Software

The following software is required for a full installation of Orchestrator on a single computer:

Microsoft SQL Server 2008 R2 or Microsoft SQL Server 2012 - Orchestrator requires only the basic SQL Server features found in the Database Engine Service. No additional features are required. Orchestrator supports SQL_Latin1_General_CP1_CI_AS for collation. The installation wizard uses SQL_Latin1_General_CP1_CI_AS as the default collation to create the orchestration database.



Note

Management servers and runbook servers installed on the same computer must use the same database. The management server must run as a 32-bit application.

- Microsoft Internet Information Services (IIS) Orchestrator Setup enables IIS if it is not enabled.
- Microsoft .NET Framework 3.5 Service Pack 1 Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.
- Microsoft .NET Framework 4.5 (which further requires WCF HTTP Activation)

To install .NET Framework 4.5 and HTTP Activation on Windows Server 2012

- 1. On the Windows Start screen, click the Server Manager tile.
- 2. On the Manage menu in the Server Manager console, click Add Roles and Features.
- 3. Go through the wizard until you reach the **Features** page.
- 4. Expand .NET Framework 4.5 Features.
- 5. Select .NET Framework 4.5 if it isn't already selected.
- 6. Expand WCF Services.
- 7. Select HTTP Activation if it isn't already selected.
- 8. Click Next and follow the prompts to finish the installation. If you have problems, check the issues covered in Troubleshoot Your Orchestrator Installation.

We recommend the following software for a full installation of Orchestrator on a single computer:

Join the computer to an Active Directory domain.

Note

On first use of the Orchestration console, you are prompted to install Microsoft Silverlight 4 on the computer if it is not already installed.

Running Orchestrator on Windows Azure virtual machines

System Center 2012 R2 Orchestrator runs on Windows Azure just as it does on physical computer systems.

Orchestrator was tested by Microsoft by installing and using it in a Windows Azure virtual machine. The testing concluded that Orchestrator was fully functional and operated exactly the same as it does on physical hardware. Stability and performance benchmarks inside a Windows Azure virtual machine were at a level where no special considerations were needed. Orchestrator does not require a domain controller to be deployed with it, and the virtual machine requirements do not differ from what is recommended for the product that is deployed in an on-premises virtual machine.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Plan Your Orchestrator Deployment
- System Requirements
- **Individual Feature Requirements**

Single-Computer Requirements for Orchestrator in System Center 2012 SP1

This section describes the minimum hardware and software configuration required for a full installation of System Center 2012 - Orchestrator on a single computer.



Orchestrator is not supported when installed on the same computer as a domain controller.

Hardware

The following minimum hardware configuration is required for a full installation of Orchestrator:

- Minimum 1 gigabyte (GB) of RAM, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigahertz (GHz) or better

Operating system

The following table lists the supported operating systems for a full installation of Orchestrator on a single computer.

Feature	Operating system
management server	Windows Server 2008 R2, Windows Server
Orchestrator web service	2012
Runbook Designer	
runbook server	

The following software is required for a full installation of Orchestrator on a single computer:

Microsoft SQL Server 2008 R2 or Microsoft SQL Server 2012 - Orchestrator requires only the basic SQL Server features found in the Database Engine Service. No additional features are required. Orchestrator supports SQL_Latin1_General_CP1_CI_AS for collation. The installation wizard uses SQL_Latin1_General_CP1_CI_AS as the default collation to create the orchestration database.



Note

Management servers and runbook servers installed on the same computer must use the same database. The management server must run as a 32-bit application.

- Microsoft Internet Information Services (IIS) Orchestrator Setup enables IIS if it is not enabled.
- Microsoft .NET Framework 3.5 Service Pack 1 Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.
- Microsoft .NET Framework 4.5 (which further requires HTTP Activation)
- WCF HTTP Activation)

To install .NET Framework 4.5 and HTTP Activation on Windows Server 2012

- 1. On the Windows Start screen, click the Server Manager tile.
- 2. On the Manage menu in the Server Manager console, click Add Roles and Features.
- 3. Go through the wizard until you reach the **Features** page.
- 4. Expand .NET Framework 4.5 Features.
- 5. Select .NET Framewok 4.5 if it isn't already selected.
- 6. Expand WCF Services.
- 7. Select **HTTP Activation** if it isn't already selected.
- 8. Click **Next** and follow the prompts to finish the installation. If you have problems, check the issues covered in Troubleshoot Your Orchestrator Installation.

We recommend the following software for a full installation of Orchestrator on a single computer:

Join the computer to an Active Directory domain.



Note

On first use of the Orchestration console, you are prompted to install Microsoft Silverlight 4 on the computer if it is not already installed.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Plan Your Orchestrator Deployment
- **System Requirements**
- **Individual Feature Requirements**

Single-Computer Requirements for System Center 2012 - Orchestrator

This section describes the minimum hardware and software configuration required for a full installation of System Center 2012 - Orchestrator on a single computer.



Orchestrator is not supported when installed on the same computer as a domain controller.

Hardware

The following minimum hardware configuration is required for a full installation of Orchestrator:

- Minimum 1 gigabyte (GB) of RAM, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigahertz (GHz) or better

Operating system

The following table lists the supported operating systems for a full installation of Orchestrator on a single computer.

Feature	Operating system
Management server	Windows Server 2008 R2
Orchestrator web service	
Runbook Designer	
runbook server	

Software

The following software is required for a full installation of Orchestrator on a single computer:

Microsoft SQL Server 2008 R2 – Orchestrator requires only the basic SQL Server features found in the Database Engine Service. No additional features are required. Orchestrator supports SQL Latin1 General CP1 CI AS for collation. The installation wizard uses

SQL_Latin1_General_CP1_CI_AS as the default collation to create the orchestration database.



Note

Management servers and runbook servers installed on the same computer must use the same database. The management server must run as a 32-bit application.

- Microsoft Internet Information Services (IIS) Orchestrator Setup enables IIS if it is not enabled.
- Microsoft .NET Framework 3.5 Service Pack 1 Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.
- Microsoft .NET Framework 4

We recommend the following software for a full installation of Orchestrator on a single computer:

Join the computer to an Active Directory domain.



Note

On first use of the Orchestration console, you are prompted to install Microsoft Silverlight 4 on the computer if it is not already installed.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Plan Your Orchestrator Deployment
- **System Requirements**
- **Individual Feature Requirements**

Individual Feature Requirements

This section describes the minimum hardware and software configuration required to install individual features in System Center 2012 - Orchestrator, Orchestrator in System Center 2012 Service Pack 1 (SP1), or System Center 2012 R2 Orchestrator depending on your version of System Center 2012.

Individual Feature Requirements for System Center 2012 R2 Orchestrator

- Management Server Requirements for System Center 2012 R2 Orchestrator
- Runbook Server Requirements for System Center 2012 R2 Orchestrator
- Orchestrator Web Service Requirements for System Center 2012 R2 Orchestrator
- Runbook Designer Requirements for System Center 2012 R2 Orchestrator

Individual Feature Requirements for Orchestrator in System Center 2012 SP1

- Management Server Requirements for Orchestrator in System Center 2012 SP1
- Runbook Server Requirements for Orchestrator in System Center 2012 SP1
- Orchestrator Web Service Requirements for Orchestrator in System Center 2012 SP1
- Runbook Designer Requirements for Orchestrator in System Center 2012 SP1

Individual Feature Requirements for System Center 2012 - Orchestrator

- Management Server Requirements for System Center 2012 Orchestrator
- Runbook Server Requirements for System Center 2012 Orchestrator
- Orchestrator Web Service Requirements for System Center 2012 Orchestrator
- Runbook Designer Requirements for System Center 2012 Orchestrator

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Plan Your Orchestrator Deployment
- Single-Computer Requirements for Orchestrator in System Center 2012 SP1

Runbook Designer Requirements for System Center 2012 R2 Orchestrator



These system requirements are for System Center 2012 R2 Orchestrator. To see the system requirements for Orchestrator in System Center 2012 Service Pack 1 (SP1), see Management Server Requirements for Orchestrator in System Center 2012 SP1. To see the system requirements for System Center 2012 - Orchestrator, see Management Server Requirements for System Center 2012 - Orchestrator.

This topic describes the hardware and software requirements for an installation of the Orchestrator Runbook Designer.

Hardware

The following minimum hardware configuration is required for the Orchestrator Runbook Designer:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigabyte (GHz) or faster

Operating system

The following table lists the supported operating systems for the Orchestrator Runbook Designer.

Feature	Operating system
Runbook Designer	Windows Server 2008 R2
	Windows Server 2012
	Windows Server 2012 R2
	Windows 7, 32-bit or 64-bit

Software

The following software must be deployed and available to successfully install the Orchestrator Runbook Designer:

A functional Orchestrator management server and database.

The targeted computer requires the following software to install the Orchestrator Runbook Designer:

Microsoft .NET Framework 3.5 Service Pack 1 - Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

See Also

Install Individual Orchestrator Features

Orchestrator Web Service Requirements for System Center 2012 R2 Orchestrator



Important

These system requirements are for System Center 2012 R2 Orchestrator. To see the system requirements for Orchestrator in System Center 2012 Service Pack 1 (SP1), see Management Server Requirements for Orchestrator in System Center 2012 SP1. To see the system requirements for System Center 2012 - Orchestrator, see Management Server Requirements for System Center 2012 - Orchestrator.

This topic describes the hardware and software requirements for an installation of the Orchestrator web service.

Hardware

The following minimum hardware configuration is required for the Orchestrator web service:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigahertz (GHz) or faster

Operating system

The following table lists the supported operating systems for the Orchestrator web service.

Feature	Operating system
Orchestrator web service	Windows Server 2008 R2
	Windows Server 2012
	Windows Server 2012 R2

Software

The following must be deployed and available to successfully install the Orchestrator web service:

A functional Orchestrator management server and database.

The targeted computer requires the following software to install the Orchestrator web service:

- Internet Information Services (IIS) 7.0 and enabled IIS role Orchestrator Setup enables the IIS role if it is not already enabled.
- Microsoft .NET Framework 3.5 Service Pack 1 Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

- Microsoft .NET Framework 4.5 (which further requires HTTP Activation)
- WCF HTTP Activation)

To install .NET Framework 4.5 and HTTP Activation on Windows Server 2012

- 1. On the Windows Start screen, click the Server Manager tile.
- 2. On the Manage menu in the Server Manager console, click Add Roles and Features.
- 3. Go through the wizard until you reach the **Features** page.
- 4. Expand .NET Framework 4.5 Features.
- 5. Select .NET Framewok 4.5 if it isn't already selected.
- 6. Expand WCF Services.
- 7. Select HTTP Activation if it isn't already selected.
- 8. Click **Next** and follow the prompts to finish the installation. If you have problems, check the issues covered in Troubleshoot Your Orchestrator Installation.



Microsoft Silverlight 4 is not required for the Orchestrator web service installation. It is required for any computer that runs the Orchestration console.

See Also

Install Individual Orchestrator Features

Runbook Server Requirements for System Center 2012 R2 Orchestrator



These system requirements are for System Center 2012 R2 Orchestrator. To see the system requirements for Orchestrator in System Center 2012 Service Pack 1 (SP1), see Management Server Requirements for Orchestrator in System Center 2012 SP1. To see the system requirements for System Center 2012 - Orchestrator, see Management Server Requirements for System Center 2012 - Orchestrator.

This topic describes the hardware and software requirements for installation of the Orchestrator runbook server.

Hardware

The following minimum hardware configuration is required for an Orchestrator runbook server:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigabyte (GHz) or faster

Operating system

The following table lists the supported operating systems for an Orchestrator runbook server.

Feature	Operating system
runbook server	Windows Server 2008 R2
	Windows Server 2012
	Windows Server 2012 R2

The following software must be deployed and available to install the Orchestrator runbook server:

A functional Orchestrator management server and database.

The targeted computer requires the following software to install the Orchestrator runbook server:

Microsoft .NET Framework 3.5 Service Pack 1 - Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

See Also

Install Individual Orchestrator Features

Management Server Requirements for System Center 2012 R2 Orchestrator



Important

These system requirements are for System Center 2012 R2 Orchestrator. To see the system requirements for Orchestrator in System Center 2012 Service Pack 1 (SP1), see Management Server Requirements for Orchestrator in System Center 2012 SP1. To see the system requirements for System Center 2012 - Orchestrator, see Management Server Requirements for System Center 2012 - Orchestrator.

This topic describes the hardware and software requirements for installation of the System Center 2012 - Orchestrator management server.

Hardware

The following minimum hardware configuration is required for the Orchestrator management server:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigahertz (GHz) or better

Operating system

The following table lists the supported operating systems for the Orchestrator management server.

Feature	Operating system
Management server	Windows Server 2008 R2
	Windows Server 2012
	Windows Server 2012 R2

The following software must be deployed and available to install the Orchestrator management server:

Microsoft SQL Server 2008 R2 or Microsoft SQL Server 2012- Orchestrator requires only the basic SQL Server features found in the Database Engine Service. No additional features are required. The instance of SQL Server can either be installed locally on the management server or on a separate dedicated database server. Orchestrator supports SQL Latin1 General CP1 CI AS for collation. The installation wizard uses SQL_Latin1_General_CP1_CI_AS as the default collation to create the orchestration database.



Note

Management servers and runbook servers installed on the same computer must use the same database. The management server must run as a 32-bit application.

The targeted computer requires the following software to install the Orchestrator management server:

Microsoft .NET Framework 3.5 Service Pack 1 - Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

See Also

Individual Feature Requirements

Runbook Designer Requirements for Orchestrator in System Center 2012 SP1



Important

These system requirements are for Orchestrator in System Center 2012 Service Pack 1 (SP1). To see the system requirements for System Center 2012 R2 Orchestrator, see Runbook Designer Requirements for System Center 2012 R2 Orchestrator. To see the system requirements for System Center 2012 - Orchestrator, see Runbook Designer Requirements for System Center 2012 - Orchestrator.

This topic describes the hardware and software requirements for an installation of the System Center 2012 - Orchestrator Runbook Designer.

Hardware

The following minimum hardware configuration is required for the Orchestrator Runbook Designer:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigabyte (GHz) or better

Operating system

The following table lists the supported operating systems for the Orchestrator Runbook Designer.

Feature	Operating system
Runbook Designer	Windows Server 2008 R2
	Windows Server 2012
	Windows 7, 32-bit or 64-bit

The following software must be deployed and available to successfully install the Orchestrator Runbook Designer:

A functional Orchestrator management server and database.

The targeted computer requires the following software to install the Orchestrator Runbook Designer:

Microsoft .NET Framework 3.5 Service Pack 1 - Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

See Also

Install Individual Orchestrator Features

Orchestrator Web Service Requirements for Orchestrator in System Center 2012 SP1



Important

These system requirements are for Orchestrator in System Center 2012 Service Pack 1 (SP1). To see the system requirements for System Center 2012 R2 Orchestrator, see Orchestrator Web Service Requirements for System Center 2012 R2 Orchestrator. To see the system requirements for System Center 2012 - Orchestrator, see Orchestrator Web Service Requirements for System Center 2012 - Orchestrator.

This topic describes the hardware and software requirements for an installation of the Orchestrator web service.

Hardware

The following minimum hardware configuration is required for the Orchestrator web service:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigahertz (GHz) or better

Operating system

The following table lists the supported operating systems for the Orchestrator web service.

Feature	Operating system
Orchestrator web service	Windows Server 2008 R2
	Windows Server 2012

Software

The following must be deployed and available to successfully install the Orchestrator web service:

A functional Orchestrator management server and database.

The targeted computer requires the following software to install the Orchestrator web service:

- Internet Information Services (IIS) 7.0 and enabled IIS role Orchestrator Setup enables the IIS role if it is not already enabled.
- Microsoft .NET Framework 3.5 Service Pack 1 Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.
- Microsoft .NET Framework 4.5 (which further requires HTTP Activation)
- WCF HTTP Activation)

To install .NET Framework 4.5 and HTTP Activation on Windows Server 2012

- 1. On the Windows Start screen, click the Server Manager tile.
- 2. On the Manage menu in the Server Manager console, click Add Roles and Features.
- 3. Go through the wizard until you reach the **Features** page.
- 4. Expand .NET Framework 4.5 Features.
- 5. Select .NET Framewok 4.5 if it isn't already selected.
- 6. Expand WCF Services.
- 7. Select HTTP Activation if it isn't already selected.
- 8. Click **Next** and follow the prompts to finish the installation. If you have problems, check the issues covered in Troubleshoot Your Orchestrator Installation.



Microsoft Silverlight 4 is not required for the Orchestrator web service installation. It is required for any computer that runs the Orchestration console.

See Also

Install Individual Orchestrator Features

Runbook Server Requirements for Orchestrator in System Center 2012 SP1



Important

These system requirements are for Orchestrator in System Center 2012 Service Pack 1 (SP1). To see the system requirements for System Center 2012 R2 Orchestrator, see Runbook Server Requirements for System Center 2012 R2 Orchestrator. To see the system requirements for System Center 2012 - Orchestrator, see Runbook Server Requirements for System Center 2012 - Orchestrator.

This topic describes the hardware and software requirements for installation of the System Center 2012 - Orchestrator runbook server.

Hardware

The following minimum hardware configuration is required for an Orchestrator runbook server:

• 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended

- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigabyte (GHz) or better

Operating system

The following table lists the supported operating systems for an Orchestrator runbook server.

Feature	Operating system
runbook server	Windows Server 2008 R2
	Windows Server 2012

Software

The following software must be deployed and available to install the Orchestrator runbook server:

A functional Orchestrator management server and database.

The targeted computer requires the following software to install the Orchestrator runbook server:

Microsoft .NET Framework 3.5 Service Pack 1 - Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

See Also

Install Individual Orchestrator Features

Management Server Requirements for Orchestrator in System Center 2012 SP1



Important

These system requirements are for Orchestrator in System Center 2012 Service Pack 1 (SP1). To see the system requirements for System Center 2012 R2 Orchestrator, see Management Server Requirements for System Center 2012 R2 Orchestrator. To see the system requirements for System Center 2012 - Orchestrator, see Management Server Requirements for System Center 2012 - Orchestrator.

This topic describes the hardware and software requirements for installation of the System Center 2012 - Orchestrator management server.

Hardware

The following minimum hardware configuration is required for the Orchestrator management server:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigahertz (GHz) or better

Operating system

The following table lists the supported operating systems for the Orchestrator management server.

Feature	Operating system
Management server	Windows Server 2008 R2
	Windows Server 2012

The following software must be deployed and available to install the Orchestrator management server:

Microsoft SQL Server 2008 R2 or Microsoft SQL Server 2012- Orchestrator requires only the basic SQL Server features found in the Database Engine Service. No additional features are required. The instance of SQL Server can either be installed locally on the management server or on a separate dedicated database server. Orchestrator supports SQL_Latin1_General_CP1_CI_AS for collation. The installation wizard uses SQL Latin1 General CP1 CI AS as the default collation to create the orchestration database.



Note

Management servers and runbook servers installed on the same computer must use the same database. The management server must run as a 32-bit application.

The targeted computer requires the following software to install the Orchestrator management server:

Microsoft .NET Framework 3.5 Service Pack 1 - Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

See Also

Individual Feature Requirements

Management Server Requirements for System Center 2012 - Orchestrator



Important

These system requirements are for System Center 2012 - Orchestrator. To see the system requirements for System Center 2012 R2 Orchestrator, see Management Server Requirements for System Center 2012 R2 Orchestrator. To see the system requirements for Orchestrator in System Center 2012 Service Pack 1 (SP1), see Management Server Requirements for Orchestrator in System Center 2012 SP1.

This topic describes the hardware and software requirements for installation of the System Center 2012 - Orchestrator management server.

Hardware

The following minimum hardware configuration is required for the Orchestrator management

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigahertz (GHz) or better

Operating system

The following table lists the supported operating systems for the Orchestrator management server.

Feature	Operating system
Management server	Windows Server 2008 R2

Software

The following software must be deployed and available to install the Orchestrator management

Microsoft SQL Server 2008 R2 - Orchestrator requires only the basic SQL Server features found in the Database Engine Service. No additional features are required. The instance of SQL Server can either be installed locally on the management server or on a separate dedicated database server. Orchestrator supports SQL Latin1 General CP1 CI AS for collation. The installation wizard uses SQL_Latin1_General_CP1_CI_AS as the default collation to create the orchestration database.



Note

Management servers and runbook servers installed on the same computer must use the same database. The management server must run as a 32-bit application.

The targeted computer requires the following software to install the Orchestrator management server:

Microsoft .NET Framework 3.5 Service Pack 1 - Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

See Also

Individual Feature Requirements

Runbook Server Requirements for System Center 2012 - Orchestrator



Important

These system requirements are for System Center 2012 - Orchestrator. To see the system requirements for System Center 2012 R2 Orchestrator, see Runbook Server Requirements for System Center 2012 R2 Orchestrator. To see the system requirements for Orchestrator in System Center 2012 Service Pack 1 (SP1), see Runbook Server Requirements for Orchestrator in System Center 2012 SP1.

This topic describes the hardware and software requirements for installation of the System Center 2012 - Orchestrator runbook server.

Hardware

The following minimum hardware configuration is required for an Orchestrator runbook server:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space

Dual-core Intel microprocessor, 2.1 gigabyte (GHz) or better

Operating system

The following table lists the supported operating systems for an Orchestrator runbook server.

Feature	Operating system
runbook server	Windows Server 2008 R2

Software

The following software must be deployed and available to install the Orchestrator runbook server:

A functional Orchestrator management server and database.

The targeted computer requires the following software to install the Orchestrator runbook server:

 Microsoft .NET Framework 3.5 Service Pack 1 - Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

See Also

Install Individual Orchestrator Features

Orchestrator Web Service Requirements for System Center 2012 - Orchestrator



Important

These system requirements are for System Center 2012 - Orchestrator. To see the system requirements for System Center 2012 R2 Orchestrator, see <u>Orchestrator Web Service Requirements for System Center 2012 R2 Orchestrator</u>. To see the system requirements for Orchestrator in System Center 2012 Service Pack 1 (SP1), see <u>Orchestrator Web Service Requirements for Orchestrator in System Center 2012 SP1</u>.

This topic describes the hardware and software requirements for an installation of the Orchestrator web service.

Hardware

The following minimum hardware configuration is required for the Orchestrator web service:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigahertz (GHz) or better

Operating system

The following table lists the supported operating systems for the Orchestrator web service.

Feature	Operating system
Orchestrator web service	Windows Server 2008 R2

Software

The following must be deployed and available to successfully install the Orchestrator web service:

A functional Orchestrator management server and database.

The targeted computer requires the following software to install the Orchestrator web service:

- Internet Information Services (IIS) 7.0 and enabled IIS role Orchestrator Setup enables the IIS role if it is not already enabled.
- Microsoft .NET Framework 3.5 Service Pack 1 Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.
- Microsoft .NET Framework 4



Note

Microsoft Silverlight 4 is not required for the Orchestrator web service installation. It is required for any computer that runs the Orchestration console.

See Also

Install Individual Orchestrator Features

Runbook Designer Requirements for System Center 2012 - Orchestrator



Important

These system requirements are for System Center 2012 - Orchestrator. To see the system requirements for System Center 2012 R2 Orchestrator, see Runbook Designer Requirements for System Center 2012 R2 Orchestrator. To see the system requirements for Orchestrator in System Center 2012 Service Pack 1 (SP1), see Runbook Designer Requirements for Orchestrator in System Center 2012 SP1.

This topic describes the hardware and software requirements for an installation of the System Center 2012 - Orchestrator Runbook Designer.

Hardware

The following minimum hardware configuration is required for the Orchestrator Runbook Designer:

- 1 gigabyte (GB) of RAM minimum, 2 GB or more recommended
- 200 megabyte (MB) of available hard disk space
- Dual-core Intel microprocessor, 2.1 gigabyte (GHz) or better

Operating system

The following table lists the supported operating systems for the Orchestrator Runbook Designer.

Feature	Operating system
Runbook Designer	Windows Server 2008 R2
	Windows 7, 32-bit or 64-bit

Software

The following software must be deployed and available to successfully install the Orchestrator Runbook Designer:

A functional Orchestrator management server and database.

The targeted computer requires the following software to install the Orchestrator Runbook Designer:

 Microsoft .NET Framework 3.5 Service Pack 1 - Orchestrator Setup installs and enables .NET Framework 3.5 SP1 if it is not installed and enabled.

See Also

Install Individual Orchestrator Features

Orchestrator Security Planning

This topic describes the service account and user account requirements, as well as security considerations for your System Center 2012 - Orchestrator deployment. You should review this topic, create the required accounts and groups, and determine if you have any additional security requirements before starting the Orchestrator installation.

The following topics are discussed:

- Orchestrator Service Accounts
- Orchestrator Users Group
- Orchestration Database Security
- Runbook Security
- Orchestrator Web Service and Orchestration Console Security
- Using Windows Firewall with Orchestrator
- Orchestrator Security Scenarios
- Orchestrator Data Encryption

Orchestrator Service Accounts

Service accounts are required for the services listed in the following table. You must create these accounts before installing the features that use them. Details for about each account are provided below.

Server	Service
Management server	Orchestrator Management Service
	Orchestrator Runbook Server Monitor service
Runbook server	Orchestrator Runbook Service

Orchestrator Management Service account

The Orchestrator Management Service is installed on the management server. Its service account is specified during the installation of Orchestrator. If you installed the management server and the runbook server on the same computer at the same time, this is the same account used by the Management Server Service and Runbook Server Service on each computer to access

system resources. If you installed the runbook server after you already installed the management server, or if you installed the runbook server on a different computer, you can use different accounts.

The Orchestrator Management Service is responsible for maintaining the orchestration database, communicating with the Runbook Designers, and communicating with the Deployment Manager.

The account used for the Orchestrator Management Service can be a local account on the management server if the database is installed locally or if you are using SQL Server authentication to communicate with the database (although this is not recommended). However, this configuration might not allow access to other network resources. If the database is located on another server, either the account must be joined to the Active Directory domain so it can access the database server, or you must use SQL Server authentication. Use the latter option if your database server is in a different domain than the management server.

This service account does not have to be an Administrator or a domain Administrator account. Note, however, that the Deployment Manager requires administrator privileges.

The service account for the Management Server Service must have the following permissions:

- Permission to log on to the management server as a service. This permission is automatically granted during the installation process.
- Member of the Microsoft.SystemCenter.Orchestrator.Admins role in the orchestration database. The account is automatically added to this role during the installation process.

Orchestrator Runbook Server Monitor service account

The Runbook Server Monitor is installed on the management server and is responsible for monitoring the health of runbook servers. It uses the same account as the Orchestrator Management Service and requires the same permissions.

Orchestrator Runbook Service account

The Runbook Server Service is installed on each runbook server. If you installed the management server and the runbook server on the same computer at the same time, this is the same account used by the Management Server Service and Runbook Server Service on each computer to access system resources. If you installed the runbook server after you already installed the management server, or if you installed the runbook server on a different computer, you can use different accounts. The service is responsible for running runbooks and for communicating with the orchestration database.

By default, all activities in a runbook run under the service account of the runbook server on which they are running. Some activities can specify different credentials to be used for individual actions as required. Because runbook activities often access resources on other computers, it is recommended that the account used for the Orchestrator Runbook Service be an Active Directory domain account so that it can be granted access to these external resources.

The account for the Orchestrator Runbook Service must have the following permissions:

- Permission to log on to the runbook server as a service.
- Depending on the resources that the activities in your runbooks access, the service account might require additional credentials on remote computers. Specific activities can also be

configured with alternate credentials if the service account does not have access to particular resources.

Orchestrator Users Group

Users gain access to Orchestrator through membership in the Orchestrator Users group. Any user account added to this group is granted permission to use the Runbook Designer and Deployment Manager tools. By default, users in this group have the authority to perform the following actions:

- Create new runbooks. View, change, and run existing runbooks.
- Deploy new runbook servers
- Deploy new Runbook Designers
- Register and deploy integration packs
- View and change global settings for a management server

The Orchestrator Users group has the following permissions in the management server DCOM component:

- Local & Remote Launch
- Local & Remote Activation
- Local & Remote Access

If you enable remote access for the user group (by selecting Remote Permissions during installation), the user group is added to the machine limits – Local and Remote launch, activation and access.

You specify the Orchestrator Users group during the Orchestrator installation process. Because the Orchestrator web service uses the same group for authorization, you must use a domain group in Active Directory if the Orchestration console is not installed on the management server. If the Orchestration console is installed on the management server, the group can be a local group on the management server.

The decision of which to use depends on where you want to manage the group's users. Typically using an Active Directory group provides better centralized access to the group as opposed to managing it locally on the management server.



Note

A member of the Orchestrator Users group can grant access to other users to view and run runbooks from the Orchestration console without having to add those users to the group. Those who only use the Orchestration console are referred to as operators. They typically require the ability to run runbooks, but not to create them. For information about setting permissions for individual runbooks, see Runbook Permissions in Using Runbooks in System Center 2012 - Orchestrator.

Orchestration Database Security

The following sections provide information about securing the orchestration database in Orchestrator:

- <u>Database roles</u>
- Securing SQL server connections
- Encryption keys

Database roles

Security to the orchestration database is implemented through database roles in the supported versions of Microsoft SQL Server. The table below lists the roles that are created in the orchestration database and the permissions granted to each. These roles are configured and populated with the required members during the installation process, so there is typically no requirement to work directly with them. The information provided here is to help the administrator better understand the security behind the configuration and prepare for possible custom scenarios.

Account	Database role
Management Service Account	Microsoft.SystemCenter.Orchestrator.Admins
Member of Orchestrator Admins Group	Microsoft.SystemCenter.Orchestrator.Admins
Orchestrator Runbook Service Account	Microsoft.SystemCenter. Orchestrator.Runtime
Orchestrator Runbook Server Monitor Service Account	Microsoft.SystemCenter. Orchestrator.Runtime
Orchestrator Web Service User Account	Microsoft.SystemCenter. Orchestrator.Operators

Role	Permission	Object
Microsoft.SystemC	SELECT	[Microsoft.SystemCenter.Orchestrator.Runtime].[Jobs],
enter. Orchestrator.Oper		[Microsoft.SystemCenter.Orchestrator.Runtime].[RunbookInst ances],
ators		[Microsoft.SystemCenter.Orchestrator.Runtime].[RunbookInst anceParameters],
		[Microsoft.SystemCenter.Orchestrator.Runtime].[RunbookServers],
		[Microsoft.SystemCenter.Orchestrator.Runtime].[ActivityInstan ces],
		[Microsoft.SystemCenter.Orchestrator.Runtime].[ActivityInstan ceData],
		[Microsoft.SystemCenter.Orchestrator.Runtime].[Events],
		[Microsoft.SystemCenter.Orchestrator.Statistics].[Statistics]
Microsoft.SystemC	EXECUTE	[Microsoft.SystemCenter.Orchestrator].[GetSecurityToken],

Role	Permission	Object
enter.		[Microsoft.SystemCenter.Orchestrator].[AccessCheck],
Orchestrator.Oper ators		[Microsoft.SystemCenter.Orchestrator].[ComputeAuthorization Cache],
		[Microsoft.SystemCenter.Orchestrator.Statistics.Internal].[Get StatisticsSummary],
		[Microsoft.SystemCenter.Orchestrator.Runtime].[CreateJob],
		[Microsoft.SystemCenter.Orchestrator.Runtime].[CancelJob]
Microsoft.SystemC	SELECT	All tables,
enter.		dbo.[POLICIES_VIEW],
Orchestrator.Runti me		dbo.[POLICY_REQUEST_HISTORY]
Microsoft.SystemC enter.	INSERT	dbo.[OBJECT_AUDIT]
Orchestrator.Runti me		
Microsoft.SystemC	INSERT,	dbo.[OBJECTS],
enter.	UPDATE	dbo.[ACTIONSERVERS],
Orchestrator.Runti		dbo.[POLICYINSTANCES],
me		dbo.[OBJECTINSTANCES],
		dbo.[OBJECTINSTANCEDATA]
Microsoft.SystemC	INSERT,	dbo.[COUNTERINSTANCES],
enter. Orchestrator.Runti me	DELETE	dbo.[POLICYRETURNDATA]
Microsoft.SystemC enter. Orchestrator.Runtime	UPDATE	dbo.[POLICY_PUBLISH_QUEUE]
Microsoft.SystemC	CONTROL	[ORCHESTRATOR_ASYM_KEY],
enter. Orchestrator.Runti me		[ORCHESTRATOR_SYM_KEY]
Microsoft.SystemC	EXECUTE	dbo.sp_insertevent,
enter.		dbo.sp_PublishPolicy,
Orchestrator.Runti me		dbo.sp_UnpublishPolicy,
		dbo.sp_UnpublishPolicyRequest,

Role	Permission	Object
		dbo.fn_GetPolicyInstanceStatus,
		dbo.fn_NumFailedInstancesPerServer,
		dbo.fn_NumInstancesPerServer,
		dbo.fn_NumRunningInstancesPerServer,
		[Microsoft.SystemCenter.Orchestrator.Cryptography].[Encrypt]
		[Microsoft.SystemCenter.Orchestrator.Cryptography].[Decrypt]
		[Microsoft.SystemCenter.Orchestrator.Internal].[RethrowError]
Microsoft.SystemC enter. Orchestrator.Admi ns	SELECT, INSERT, UPDATE, DELETE, ALTER, CREATE TABLE	SCHEMA::dbo
Microsoft.SystemC enter. Orchestrator.Admins	REFEREN CES	dbo.[OBJECTS]
Microsoft.SystemC enter. Orchestrator.Admi	SELECT	dbo.[POLICIES_VIEW], GRANT SELECT ON dbo.[POLICY_REQUEST_HISTORY]
Microsoft.SystemC enter. Orchestrator.Admins	CONTROL	[ORCHESTRATOR_ASYM_KEY], [ORCHESTRATOR_SYM_KEY]
Microsoft.SystemC enter. Orchestrator.Admi ns	EXECUTE	[Microsoft.SystemCenter.Orchestrator.Cryptography].[CreateOrchestratorKeys], [Microsoft.SystemCenter.Orchestrator.Cryptography].[DropOrchestratorKeys], [Microsoft.SystemCenter.Orchestrator.Cryptography].[Encrypt],
		[Microsoft.SystemCenter.Orchestrator.Cryptography].[Decrypt] , [Microsoft.SystemCenter.Orchestrator.Internal].[RethrowError],

Role	Permission	Object
		dbo.sp_CustomLogCleanup,
		dbo.sp_GetLogEntriesForDelete_FilterByDays,
		dbo.sp_GetLogEntriesForDelete_FilterByEntries,
		dbo.sp_GetLogEntriesForDelete_FilterByEntriesAndDays,
		dbo.sp_insertevent,
		dbo.sp_PublishPolicy,
		dbo.sp_UnpublishPolicy,
		dbo.sp_UnpublishPolicyRequest,
		dbo.fn_GetPolicyInstanceStatus,
		dbo.fn_NumFailedInstancesPerServer,
		dbo.fn_NumInstancesPerServer,
		dbo.fn_NumRunningInstancesPerServer,
		[Microsoft.SystemCenter.Orchestrator.Internal].AddUserToRol
		e,
		[Microsoft.SystemCenter.Orchestrator].[SetPermissions],
		[Microsoft.SystemCenter.Orchestrator.Internal].[SetProductInf o]

The Database Configuration Utility (**DBSetup.exe**) requires permissions as a user on the computer where the management server is installed and is a member of either the Administrators or Orchestrator Users Group to access the settings.dat file. Custom tools that connect to the database directly through DBDataStore.dll require the same permissions.

Security

When installing Orchestrator, ensure that the account used to connect to SQL server has minimum privileges on the SQL server to avoid a potential elevation of privileges.

Securing SQL server connections

The SQL server connections in a default deployment of Orchestrator are not secure. The exception to this is when Orchestrator stores or retrieves sensitive data. In this case, Orchestrator creates a secure connection to SQL server with a self-signed certificate. This certificate does not provide strong security and is susceptible to man-in-the-middle attacks.

For information about encrypting connections to SQL Server, go to <u>Encrypting Connections to SQL Server (configuring SSL)</u>. For information on how to enable connections to the database engine, go to <u>How to: Enable Encrypted Connections to the Database Engine (SQL Server Configuration Manager)</u>.

Encryption keys

As part of your security planning, you should plan for rotating your encryption keys at a regular interval. The National Institute of Standards and Technology(NSIT) recommends that keys be rotated at least once every two years. For more information about NSIT security standards, go to NSIT Computer Security Division Computer Security Resource Center.

To rotate encryption keys

- 1. From the Runbook Designer, export all of your runbooks, global settings, variables, schedules, and so on.
 - You should provide a password for the export.
 - During export, all encrypted data is decrypted and re-encrypted with a new key created by the password.
- 2. If you want, change the SQL Server Master Database key.
 - Orchestrator encrypts data using both the SQL Server Master Database key and the master database key for the orchestration database.
 - For information on how to change the SQL Server Master Database key, go to <u>SQL</u> <u>Server and Database Encryption Keys (Database Engine)</u>.
- 3. Re-install the management server and create a new database.
 - For information on how to install the management server, see the topic <u>How to Install a Management Server for System Center 2012 Orchestrator</u>.
 - Do not connect to the existing database. A new cryptographic key is generated when a new database is created.
- 4. From the Runbook Designer, re-import the runbooks and any other data you exported. Provide the password used for the export. The data in the export file is decrypted using the password, and encrypted as it is imported to the database using the new Orchestrator master database key.

Runbook Security

All elements of a runbook are accessible to all Runbook Designers, as well as to any runbook servers in your environment. You can modify the permissions for runbook elements (such as a folder), but any permissions you set are not enforced.

Orchestrator Web Service and Orchestration Console Security

If you plan to install the Orchestrator web service and orchestration console, you should choose a secure protocol such as HTTPS to secure communication and prevent malformed requests from a man-in-the-middle attack. For more information on securing your Orchestrator web service and the Orchestration console, go to How to Configure the Orchestrator Web Service to use HTTPS.

In the default configuration of an Orchestrator deployment, web service calls are not logged. This applies to requests made with the Orchestration console as well as the Orchestration Integration Toolkit (OIT). The result is that a user can start a job and pass parameters into a runbook with no record of who started the job.

To record all requests to your Orchestrator web service, you should enable audit trail logging with **atlc.exe**. For more information about logging using **atlc.exe**, go to Audit Trail.

Using Windows Firewall with Orchestrator

Windows Firewall with Advanced Security is enabled by default on all Windows 2008 R2 computers, and blocks all incoming traffic unless it is a response to a request by the host or it is specifically allowed by a firewall rule to allow the traffic. You can explicitly allow traffic by specifying a port number, application name, service name, or other criteria by configuring Windows Firewall with Advanced Security settings.

When you configure a Runbook Designer or a runbook server outside of a firewall, certain rules must be enabled on the management server computer to allow the Runbook Designer and the runbook server to communicate with the management. Additionally, for some activities such as the Monitoring Activities, if the target computer is outside the firewall, you must enable certain firewall rules to allow WMI communication.

Configuration of Orchestrator computers

When a Runbook Designer or a runbook server is installed behind a firewall, specific firewall rules are required between the management server and the remote computers.

Enable the following rules as they apply to your configuration.

To enable access to your SQL server

1. On the remote computer where a Runbook Designer or a runbook server is installed, open a port to connect to your SQL server. The default SQL port is TCP:1433.

To enable access between the Runbook Designer and the management server

1. On the computer running the Management Server Service, add a firewall rule to allow Runbook Designer or runbook server to access **ManagementService.exe**.

Location of Orchestrator Management Service

Operating system	Firewall rule	
64-bit	%Program Files (x86)%\Microsoft System Center 2012\Orchestrator\Management Server\ManagementService.exe	

To grant privilege to the Runbook Server Service account

1. On the remote runbook server computer, confirm that the Runbook Server Service

account has the Logon as service privilege.

To allow remote deployments with the Deployment Manager

 On the remote computer where you deployed the runbook server or the Runbook Designer, add a rule to allow the Deployment Manager to access the Orchestrator Remoting Service.

Location of Orchestrator Remoting Service

Operating system	File location
64-bit	%SystemRoot%\SysWOW64\OrchestratorRemotingService.exe
32-bit	%SystemRoot%\System32\OrchestratorRemotingService.exe

For more information about adding firewall rules see Add or Edit a Firewall Rule.

Firewall rules for activities

Any activities that use WMI communication, such as any of the Monitoring Activities, require certain Windows Firewall rules to function correctly.

For Windows Server 2008 R2, enable the following rules to allow any activity that uses WMI to function correctly:

- Windows Management Instrumentation (Async-In)
- Windows Management Instrumentation (DCOM-In)
- Windows Management Instrumentation (WMI-In)

Orchestrator Security Scenarios

The following information provides best practices for using Orchestrator securely. This information is provided in the format of scenarios. The following scenarios are available:

- Scenario: Securely transitioning from development to test to production environments
- Scenario: Effectively managing Orchestrator Users group membership

Scenario: Securely transitioning from development to test to production environments

The Orchestrator password data contained in runbooks can be securely shared between different instances of Orchestrator. For example, one may wish to export runbooks built in a development environment and import them into a test environment or export tested runbooks into a production environment. This export and import process would need to secure the encrypted data in each phase of the export in such a way that the exported data could be imported into a different Orchestrator environment.

This is accomplished using the Import/Export functionality available in the Runbook Designer. The export and import features are available from the **Actions** item on the Runbook Designer menu bar or by right-clicking a runbook folder. The export feature is also available by right-clicking a runbook tab, a feature commonly referred to as a "single runbook export."

Regardless of how a runbook is exported, the encrypted data contained in runbooks will be stored securely in the resulting XML export file. This is accomplished by providing a password upon export. When Orchestrator exports the runbooks and their related configuration, any encrypted data contained in Runbooks is decrypted and encrypted again upon export using the provided password.

Note

- 1. The encryption key used for the export is different from that used to store the data in the Orchestrator database. Essentially, the "export" feature decrypts the encrypted data and reencrypts it in the export file. The export file contains the encrypted password.
- 2. The export process does not protect the runbook itself nor the non-encrypted data contained in Runbooks. The export only protects encrypted data contained in Runbooks.

When an export file is re-imported the import requires a password be provided. If the password matches then the encrypted data contained in export will be imported and re-encrypted for storage in the Orchestrator database by using the encryption key.

Note

- The Export/Import password feature does not support password complexity rules that may be required by your organization. A blank value for the password is permitted, although not recommended for exports that contain sensitive data that has been encrypted.
- If the password for your export is lost one can still perform an import of the runbooks and their related configuration. On the Import screen simply clear the Import Orchestrator encrypted data option. Any Orchestrator platform-encrypted data will not be imported and created with blank values in the Orchestrator database.

Scenario: Effectively managing Orchestrator Users group membership

Orchestrator has two core user roles: Runbook Authors and Operators. These user roles have different rights in Orchestrator. Runbook Authors are individuals that have rich administrative access to Orchestrator including its database and configuration. Runbook Authors grant access to Runbook Operators. Runbook Operators have access to the Orchestration Console and Web Service based on rights granted to them by Runbook Authors.

User Role	Identified by	Rights	
Runbook Author	Membership in the Orchestrator Users Group	Administrators of Orchestrator	
	(see below)	Read, write, update Orchestrator configuration	
		Full control of the Orchestrator database	
		Full encrypt/decrypt rights	
		Access to Runbook Activities that can interact with external systems via	

User Role	Identified by	Rights	
		Integration Packs	
-	Runbook Folder permissions granted by Runbook Authors	Non-administrative rights to Orchestrator	
	in the Runbook Designer	Access to the Orchestration Console and Web Service	
		View and invoke runbooks based on rights granted by Runbook Authors	
	No access to the Orchestrator database		
		No encrypt/decrypt rights	



Placing a user account in the Orchestrator Users group identifies this user account as being an administrator of Orchestrator. All Orchestrator users are essentially equallyprivileged administrators with full access to Orchestrator and the data contained in the database. This would include access to encrypt and decrypt data contained in the Orchestrator database.

Orchestrator manages security through membership in two security groups created at installation time. These are the Orchestrator Users group and the Orchestrator System group. Membership in either or both of these groups identifies accounts that are considered administrators of Orchestrator ("trusted personas"). Administrative rights include the ability to update runbooks and their related configuration data, update the configuration of runbook servers, interact with external systems via integration packs, install and deploy integration packs, interact programmatically with the Orchestrator database, update the database configuration and encrypt/decrypt encrypted data stored in the Orchestrator database.



Note

Membership in either or both of these groups grants full administrative access to Orchestrator including access to all data contained in the Orchestrator database and full encrypt/decrypt rights.

Security group	Associated persona	Security group purpose
Orchestrator Users Group	Runbook authors and anyone who deploys integration packs	This security group defines user accounts that will be able to launch the Runbook Designer, Deployment Manager and Data Store

Security group	Associated persona	Security group purpose
		Configuration utility. Membership in this group grants privileged access to the Orchestrator database. This would include the ability to read and update the database configuration as well as access and decrypt encrypted data.
Orchestrator System Group	None (used for service accounts)	This security group defines the service accounts that require privileged access to the Orchestrator database. This would include the ability to read and update the database configuration as well as access and decrypt encrypted data.

The following user roles are considered trusted/untrusted personas in Orchestrator.

Security domain	Context	Cryptography rights	Identified by	Trusted persona
Run Time	Orchestrator Services "Invoke Runbook" Alternate Credentials	Full encrypt & decrypt	Orchestrator Systems Group in Active Directory / Credentials on "Invoke Runbook" Runbook Activity	Yes
Design Time	Runbook Designer Deployment Manager Data Store Configuration	Full encrypt & decrypt	Orchestrator Users Group in Active Directory	Yes
Operator	Orchestration Console Web Service	No explicit access to encrypted or decrypted data.	User rights defined in the Runbook Designer by the Runbook Author role	No

Security domain	Context	Cryptography rights	Identified by	Trusted persona
Database Administrator	MS SQL Server 20008 R2	Full Encrypt & decrypt	Rights to SQL Server as a DBA with rights to the Orchestrator database	Yes
Windows Administrator	Windows Server 2008 R2	No explicit rights are granted, however Windows administrators are considered trusted personas.	Rights to Windows	Yes

Orchestrator Data Encryption

The following sections provide information about data encryption in Orchestrator:

- What data is encrypted and decrypted in Orchestrator?
- How is encrypted data managed in Orchestrator?
- How can encrypted data be moved between Orchestrator instances?

Best practices for encrypted variables

Introduced in System Center 2012, encrypted variables in Orchestrator allow you to more securely use variables to provide sensitive data to runbook activities. Encrypted variables are used exactly like standard global variables; that is, by means of a subscription. If you subscribe to these variables in activity fields that get republished, the variable contents can be exposed on the data bus. Because of this, encrypted variables should be subscribed to only in fields that are not republished. This best practice is not enforced by Orchestrator, but it should be a part of your planning process.

However, if encrypted data must be published on the data bus in order to be sent to another system (for example, a product that runs on a different server), you should ensure that the channel to that product is secure. For example, BMC Remedy supports a secure mode for connection, and products with web interfaces typically allow using the Secure Sockets Layer connection (using the HTTPS protocol).

What data is encrypted and decrypted in Orchestrator?

Orchestrator provides a code set of encryption and decryption services that are used to generate Orchestrator platform-encrypted data. These services are used to secure data flagged for encryption in the Orchestrator database as well as decrypt the data to plain-text so it can be used as part of a runbook. These core encryption services are managed by the Orchestrator database

and management server. Rights to these services are granted through membership in the Orchestrator Users group or the Orchestrator System group.



Orchestrator runbooks could contain data encrypted by an external encryption service and used as runbook Published data. Orchestrator would not handle data from such an external system any differently than any other piece of data.

Orchestrator uses encryption in the following product feature areas:

Feature area	Description
Runbook activities	Any property masked out when one types in the field is an encrypted property. This would include passwords on the Security Credentials tab but can include other properties as well.
Options menu	The Options menu is used to store credentials and other information used to configure integration packs. Properties of connection settings can contain encrypted properties.
Variables	Variables that have the Encrypted Variable checkbox selected will be encrypted.



Note

Encrypted variables are intended to be used via subscription in properties that require an encrypted value such as a password used in a runbook activity. If an encrypted variable is subscribed to in a non-encrypted field the encrypted value will be provided. The plaintext value is only available when used in an encrypted property.

How is encrypted data managed in Orchestrator?

Orchestrator has a core cryptographic service whose design is based on AES using SQL Server cell-level encryption. As such, all encryption and decryption is performed centrally by SQL Server. Encryption keys are centrally managed by SQL Server. Both the SQL Server Service Master Key and the Orchestrator Database Master Key are required to encrypt and decrypt data.

Orchestrator uses cryptography in both the Run Time and Design Time experiences. Runbook authors interact with runbook activities in the Runbook Designer and often these activities will interact with external systems to "discover" property grids, list values, and other properties. Likewise, when a runbook is tested in the Runbook Tester the encrypted data provided in protected fields needs to be decrypted so it can be passed to the target system. Finally, the Runbook Servers need to be able to decrypt encrypted data to allow runbooks to interact with external systems. As such, the database cryptographic services need to be accessed from the Runbook Servers, Runbook Designer and Runbook Tester.

Since the core cryptographic services reside in the Orchestrator database, access to the database essentially defines access to the unencrypted data.

- Runbook servers access the database directly. As such they directly access the crypto services provided by SQL Server. Run Time access to the crypto services provided by SQL Server are limited to members of the Orchestrator System Group.
- Runbook Designers and the Runbook Tester access the database indirectly through the
 management server. The management server offers a new service that services requests for
 encryption/decryption from the Runbook Designer and Runbook Tester. The management
 server passes through the security context of the runbook author and these credentials are
 used to access the crypto services. Design Time access to the crypto services provided by
 SQL Server are limited to members of the Orchestrator Users group.

Access to encrypted data from Orchestrator is managed by the Orchestrator Users group and the Orchestrator Systems group. Members of these two security groups essentially have rich administrative access to Orchestrator including rights to access the core cryptographic services as well as decrypt data stored encrypted in the database.

How can encrypted data be moved between Orchestrator instances?

When the Orchestrator database is installed a database master encryption key is created. This database master key is used in conjunction with the SQL Server master key to encrypt and decrypt data stored in the Orchestrator database. This means encrypted data is essentially "keyed" to the instance of SQL Server 2008 R2 where the data was encrypted. For example, one can't "copy" an encrypted string from a column of one instance of SQL Server 2008 R2 and "paste" the value into another instance of an Orchestrator database and decrypt the data unless both the database master key and server master key matched that of the system where the data was encrypted.

Hence moving encrypted data between Orchestrator instances requires one of two scenarios:

- 1. Both the SQL Server service master key and the Orchestrator database master key are the same as the keys on the system where the data was originally encrypted.
- 2. Export the runbooks and related encrypted data and Import into the new system.

Essentially, the Export functionality creates an export file whose encrypted data has been encrypted a password provided by the user during export. This export file contains encrypted data that can be decrypted by providing the same password during import. The data will be encrypted and stored into the database by using the encryption keys for the new database.

TCP Port Requirements

Communication between Orchestrator features on different computers occurs over TCP/IP. If you have firewalls in your environment between these features, you must enable the ports indicated in the following table.

Source	Targeted computer	Default port	Configurable	Notes
Runbook Designer	Management	135, 1024-	Yes	The Runbook

Source	Targeted computer	Default port	Configurable	Notes
	server	65535		Designer communicates with the management server over DCOM. By default, DCOM communicates over port 135 and dynamically allocates a port between 1024 and 65535. For information about configuring DCOM for a specific port range, see Configuring Microsoft Distributed Transaction Coordinator (DTC) to work through a firewall.
Management server runbook server Web service	orchestration database	1433	Yes	Specified during Microsoft SQL Server installation
Client browser	Orchestrator REST-based web service	81	Yes	Specified during Orchestrator installation. Both
	Orchestration console	82		ports must be accessible for the Orchestration console.
Activities	Various targeted computers depending on activity		Packs for System	integration packs, see Center 2012 -

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Plan Your Orchestrator Deployment
- System Requirements
- Orchestrator Security Planning
- Scale Planning

See Also

Plan Your Orchestrator Deployment

Integration Packs for System Center 2012 - Orchestrator

Scale Planning

This section describes planning considerations for designing a large-scale deployment of System Center 2012 - Orchestrator. When planning your system requirements, you have to consider how to use Orchestrator runbooks, the types and number of runbooks you plan to run, the amount of required data logging, the systems you are integrating with, and the level of fault tolerance you require.

Scale planning

• Feature Performance Considerations

Describes the Orchestrator features and how their behaviors affect system performance.

• Evaluate System Requirements

Provides guidance for evaluating your deployment tasks.

Deployment Recommendations

Provides recommendations for the number of systems to install in your Orchestrator deployment.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Plan Your Orchestrator Deployment
- System Requirements
- Orchestrator Security Planning
- TCP Port Requirements

Feature Performance Considerations

This section describes the Orchestrator processes that influence performance in a production environment. The focus of this procedure is lies in identifying processes that occur during runtime, in the web service, and during authoring. While many authoring activities occur outside a production environment, considerations for setting up a production environment to test performance should also include variances, for example, whether special requests by an Orchestrator administrator are supported.

Runbooks

Despite the variance in their design and complexity, runbooks have a simple structure. They perform three operations: they run activities, manage published data, and perform branch logic. The following sections provide more details about these operations.

Activity internals

Runbook activities contain two types of code: platform code and domain code. Platform code is built on a framework that is shared between all runbooks. Platform code manages Orchestrator processes. Domain code refers to the code in a runbook activity that manages processes outside Orchestrator. For example, the Invoke Web Service activity contains platform code to handle processing in Orchestrator, such as publishing data, and domain code specific to invoking a web service.

There is little processing variability between runbooks when you compare activities that run similar platform code. Domain code depends on latency issues external to Orchestrator. Potentially, domain code varies greatly between activities. To understand the domain code dependencies and their impact on runbook performance, you must test the performance of individual activities before you determine the requirements for the production environment.

Published Data

Runbooks in Orchestrator share data between activities. Every activity consumes Published Data that earlier runbook activities created. How an activity uses the published data depends on the domain code. All runbook activities publish a minimum set of run-time parameters called Common Published Data. Domain code can, but is not required to, publish data. The Published Data that the domain code creates is called Activity-Specific Published Data. The data that an activity produces can contain data elements that are single or multi-valued. For example, every activity produces a single record of single-value Common Published Data. Domain code can produce multiple records of single and multi-value data.

Publishing data to the orchestration database is a resource-intensive activity. Runbook performance depends on the amount of data that each activity publishes and the performance and resiliency of the computer that hosts the orchestration database. As part of planning your performance requirements, consider the amount of published data your runbooks require and the performance of the computer that hosts your orchestration database.

Branching

Runbook activities create a branch if an activity requires data to pass at the same time to two or more activities. When a runbook starts, processing consists of a single thread. When this thread

encounters a branch, a thread is created for each branch. Each thread references the published data from all previous activities along the thread. The total number of threads in a runbook depends on the number of branches used in a runbook. Multi-threaded runbooks require more processing power than single threaded runbooks.

As part of assessing your runbook performance requirements, consider the number of branches you plan to include in a runbook. Runbooks with lots of branches require more processing power on the runbook servers than runbooks that contain no branches.

Operator experience

The Operator experience occurs on the Orchestration console and the Orchestrator web service. The Orchestration console is a Microsoft Silverlight-based web application that requires Orchestrator web service to connect to the orchestration database. The Orchestration console and the Orchestrator web service depend on the performance of the orchestration database and the Internet Information Services (IIS) server that hosts the Orchestrator web service.

Service Manager connector

The Orchestrator web service supports the Service Manager connector. Service Manager targets IT customers who serve approximately 50,000 users. Service Manager request-management scenarios assume that each user submits one request per month. This produces a request volume of 2,500 requests per day (200 requests/hour or approximately three requests every minutes). Service Manager uses the Orchestrator web service to update the status of activities, requiring support for a like number of status requests. Also, note that the Service Manager connector discovers published runbooks. The response time to discover any given runbook folder depends on the number of runbooks in the folder.

See Also

Scale Planning

Evaluate System Requirements

This section summarizes the ITIL best practices to determine your deployment requirements as it applies to Orchestrator. The following table shows the sequence of evaluation criteria.

Task	Information
1: Define the scope of the project.	Define scope of work
2: Identify the tasks you plan to automate.	Identify tasks
3: Identify the system workloads for Orchestrator and the tasks you plan to automate.	Define individual workloads
4: Estimate the number of running jobs per hour.	Determine total jobs running

Task	Information
5: Identify the integration packs required for your environment.	Identify required integration packs
6: Determine security requirements.	Determine the security model
7: Determine the number and placement of runbook servers.	Design runbook server requirements
8: Determine the requirements for fault tolerance.	Fault tolerance
9: Identify additional resources required for your deployment.	Resource requirements
10: Identify network traffic and potential bottlenecks.	<u>Network</u>
11: Identify your service and operations requirements.	Service and operations requirements
12: Determine the level of integration with other System Center products.	Integration with System Center
13: Determine authoring requirements.	Authoring
14: Design your Orchestrator test environment.	Test environment
15: Design your Orchestrator pre-productions environment.	Pre-production environment

Define scope of work

As part of planning the size of your deployment, begin by identifying your business requirements. This process should define the processes you want to automate by using Orchestrator, the reporting requirements for your organization, and departments impacted by this installation. Identify all applications, services, servers, and manual processes associated with the tasks you want to perform. Prioritize these requirements based on their business impact to prioritize the deployment tasks effectively.

Identify tasks

What processes do you plan to automate? Map the processes you intend to automate to the individual steps involved. This level of detail simplifies the task of authoring runbooks. You should identify business-critical processes as requiring more validation effort before relying on the runbook in a production environment.

Define individual workloads

For the processes you automate, determine how frequently you intend them to run. A runbook that is started one time per day uses significantly fewer resources than a continuously running runbook that is monitoring a system process. Consider both the workload on the Orchestrator system and the automated process. A server that previously responded to manually input requests can behave much differently when the request input occurs by automation.

Consider how much logging of Published Data is required in each of your runbooks. As logging increases, network traffic and load on the server that is hosting the Orchestrator database increases.

Determine total jobs running

When you have individual workloads defined, calculate the total number of jobs that could be running at any point in time. Your system design should take a maximum workload into account. The number and placement of your runbook servers in addition to the resources of the processes you are automating have to be sized to accommodate the largest number of running runbooks.

Identify required integration packs

Devices and applications that are not produced by Microsoft are automated through integration packs. Determine the integration packs required for your automated processes. Each software and hardware product typically requires its own integration pack. If there is no commercially available integration pack, can you create script level automation? Do you have to create custom integration packs for full automation?

Determine the security model

Security model planning should include determining if you require your Runbooks servers and resources to be located in more than one Active Directory forest. Is there a cross-domain trust? Are there Operations Manager gateways that require certificates? Review the current security requirements for your environment to identify permission and certificate requirements.

Design runbook server requirements

Do you plan to locate runbook servers across wide area network (WAN) links and trust boundaries? If so, you must determine gateway server placement in relationship to the Orchestrator database and runbook servers. While a running management server is not required to start runbooks or save runbook data, an Orchestrator database is required for all active runbook servers.

Fault tolerance

Determine the level of fault tolerance for your Orchestrator deployment. Depending on your requirements, you can design your Orchestrator environment to be highly available in the case of a single failure.

Resource requirements

Determine the requirements for your Orchestrator deployment, and any additional load that increased requirements on processes impacted by automation create. Do you have adequate

runbook servers for the number of runbooks that can be running at a given time? Is the Orchestrator database the appropriate size to handle all requests and log Published Data?

Service and operations requirements

Identify all requirements for your environment. Include any data consolidation strategies and requirements for cross-management group, data-retention, data-warehouse size, or fault-tolerance.

Network

Determine if additional bandwidth is required to support the increased traffic the runbook servers and the Orchestrator database generate. Do you have to change any network port settings to accommodate the Orchestrator web service?

Integration with System Center

Orchestrator fully supports all System Center products such as Service Manager or Operations Manager. Identify existing System Center products in your environment to determine if additional management servers or gateways are required.

Authoring

Determine where and how authoring of runbooks is carried out. Authoring of runbooks typically occurs on computers isolated from production. However, your business requirements might include the requirement to author runbooks when they were not planned.

Test environment

If you are authoring in isolation from your production environment, identify the necessary resources to build and test new runbooks.

Pre-production environment

It is prudent to deploy high impact runbooks in a pre-production environment before introducing the runbook into a production environment. Pre-production environments should closely approximate the full-scale production environment.

See Also

Scale Planning

Deployment Recommendations

The following guidelines provide options in an Orchestrator deployment to improve high availability and performance.

Management server

An Orchestrator deployment is limited to one management server. A management server does not have to be available for runbook servers or runbooks to function. If the management server is not available, you cannot connect the Runbook Designer to publish runbooks or start, monitor, or stop runbooks. You can still start, monitor, and stop runbooks with the Orchestration console.

Orchestrator database

For high availability, you can deploy the Orchestrator database on a Microsoft SQL Server cluster with a minimum of two nodes.

Orchestrator web service

The Orchestrator web service must be installed on a server that is running Internet Information Services (IIS). The Orchestrator web service does not have to be available for runbook servers or runbooks to function. If the Orchestrator web service is not available, you cannot run the Orchestration console to start, monitor, or stop runbooks. You can install the web service on multiple IIS servers configured for load balancing to provide high availability and additional capacity.

Runbook servers

For high availability, you should have at least two runbook servers. If the primary runbook server for a runbook is unavailable, the runbook can run on another server. runbook servers are not designed to run on a computer configured as a cluster node.

For more information about specifying the runbook servers for a runbook, see the Using Runbooks in System Center 2012 - Orchestrator.

Runbooks

By default, runbook servers can run 50 runbooks simultaneously. The physical computer resources and the complexity of the runbook limit the actual number of runbooks that a runbook server can manage.

For the process to modify the number of runbooks that can run simultaneously, see How to Configure Runbook Throttling.

See Also

Using Runbooks in System Center 2012 - Orchestrator

Install Orchestrator

This section provides details about how to install System Center 2012 - Orchestrator on a single server, as an individual feature, and as a highly available deployment.

For information about how to install Orchestrator at the command prompt and details about the available command line arguments, see the section Install with the Orchestrator Command Line Install Tool.



Important

If you do not want to use the System Center 2012 - Orchestrator Setup Wizard to install Orchestrator features because Server Message Block (SMB) is not permitted through your firewall or you have concerns about SMB security, you can manually install runbook servers and integration packs with the Orchestrator command line install tool.

For more information about the tool, see the topic Install with the Orchestrator Command Line Install Tool.

Install Orchestrator

- How to Install Orchestrator on a Single Computer
 - Provides detailed information about how to install Orchestrator on a single computer.
- Install Individual Orchestrator Features
 - Provides detailed information about how to install each Orchestrator feature.
- Install with the Orchestrator Command Line Install Tool
 - Provides options for installing Orchestrator from the command line.
- Use Sysprep to Deploy Orchestrator
 - Provides details for installing Orchestrator with a Sysprep image.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- **Deployment Overview**
- Plan Your Orchestrator Deployment
- Perform Post-Installation Tasks
- **Troubleshoot Your Orchestrator Installation**

How to Install Orchestrator on a Single Computer

Use the following steps to install all System Center 2012 - Orchestrator features on a single computer.

To install Orchestrator on a single computer

1. To start the System Center 2012 - Orchestrator Setup Wizard on the server where you want to install Orchestrator, double-click SetupOrchestrator.exe.



Important

Before you begin setup, close any open programs and ensure that there are no pending restarts on the computer. For example, if you have installed a server role by using Service Manager or have applied a security update, you might have to restart the computer, and then log on to the computer with the same user account to finish the installation of the server role or the security update.



Note

If User Account Control is enabled, then you will be prompted to verify that you want to allow the setup program to run. This is because it requires administrative access to make changes to the system.

2. On the main setup page, click Install.



Warning

If Microsoft .NET Framework 3.5 Service Pack 1 is not installed on your computer, a dialog box appears asking if you want to install .NET Framework 3.5 SP1. Click **Yes** to proceed with the installation.

3. On the **Product registration** page, provide the name and company for the product registration, and then click Next.



Note

For this evaluation release, a product key is not required.

- 4. On the Please read this license agreement page, review and accept the Microsoft Software License Terms, and then click Next.
- 5. On the **Select features to install** page, verify that all features are selected, and then click Next.



Note

You can choose to remove individual features. The management server is mandatory and is selected by default. The check boxes for the other features can be cleared as required.



If you want to install only an individual feature after installing a management server, use the information in Install Individual Orchestrator Features.

- 6. Your computer is checked for required hardware and software. If your computer meets all of the requirements, the All prerequisites are installed page appears. Click Next and proceed to the next step.
 - If a prerequisite is not met, a page displays information about the prerequisite that has not been met and how to resolve the issue. Follow these steps to resolve the failed prerequisite check:
 - a. Review the items that did not pass the prerequisite check. For some requirements, such as Microsoft .NET Framework 4, you can use the link provided in the Setup Wizard to install the missing requirement. The Setup Wizard can install or configure other prerequisites, such as the Internet Information Services (IIS) role.



Warning

If you enable prerequisites during setup, such as Microsoft .NET Framework 4, your computer can require a restart. If you restart your computer, you must run setup again from the beginning.

- b. After you resolve the missing prerequisites, click Verify prerequisites again.
- 7. On the Configure the service account page, enter the user name and password for the Orchestrator Management Service account. Click Test to verify the account credentials. If the credentials are accepted, click Next.

Important

The Orchestrator Management Service account must be created before this step. For more information about the Orchestrator Management Service account, see Orchestrator Management Service account in Orchestrator Security Planning.

- 8. On the Configure the database server page, enter the name of the server and the name of the instance and port number of the Microsoft SQL Server that you want to use for Orchestrator. You can also specify whether to use Windows Authentication or SQL Server Authentication, and whether to create a new database or use an existing database.
- 9. Click Test Database Connection to verify the account credentials. If the credentials are accepted, click Next.
- 10. On the Configure the database page, select an existing database or specify the name of a new database, and then click Next.
- 11. On the Configure Orchestrator management group page, accept the default configuration or enter the name of the user group to manage Orchestrator permissions, and then click Next.



Note

For more information about the Orchestrator users group, see Orchestrator Users Group.

12. On the Configure the port for the web service page, verify the port numbers for the Orchestrator web service and the Orchestration console, and then click Next.



Note

For more information about the TCP ports, see TCP Port Requirements.

- 13. On the **Select the installation location** page, verify the installation location for Orchestrator, and then click Next.
- 14. On the Microsoft Update page, optionally indicate whether you want to use the Microsoft Update services to check for updates, and then click **Next**.



Note

If you have previously accepted Microsoft Update on this computer, this page is skipped.

- 15. On the Help improve Microsoft System Center Orchestrator page, optionally indicate whether you want to participate in the Customer Experience Improvement Program or Error Reporting, and then click Next.
- 16. Review the **Installation summary** page, and then click **Install**.

The **Installing features** page appears and displays the installation progress.

17. On the **Setup completed successfully** page, optionally indicate whether you want to start the Runbook Designer, and then click Close to complete the installation.

To install the Service Management Automation web service

1. Follow the steps that are detailed in Web Service Installation

To enable network discovery for the Runbook Designer

- On the desktop of the computer that is running Windows Server, click Start, click Control Panel, click Network and Internet, click Network and Sharing Center, click Choose Home group and Sharing Options, and then click Change advanced sharing settings.
- 2. For the **Domain** profile, if needed, click the **Arrow** icon to expand the section options.
- Select Turn on network discovery, and then click Save changes.
 If you are prompted for an administrator password or confirmation, type the password or provide confirmation.

See Also

Install Individual Orchestrator Features
Orchestrator Security Planning
TCP Port Requirements

Install Individual Orchestrator Features

This section provides information about how to install each of the Orchestrator features. You can use this information to install features on individual computers or to add additional servers or features to your Orchestrator deployment.

How to install individual Orchestrator features

- How to Install a Management Server for System Center 2012 Orchestrator
- How to Install a Runbook Server for System Center 2012 Orchestrator
- How to Install the Orchestrator Web Service
- How to Install the Runbook Designer for System Center 2012 Orchestrator

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Install Orchestrator
- How to Install Orchestrator on a Single Computer

How to Install a Management Server for System Center 2012 - Orchestrator

Use the following steps to install an System Center 2012 - Orchestrator management server.

To install an Orchestrator management server

 On the server where you want to install Orchestrator, start the System Center 2012 -**Orchestrator Setup Wizard.**

To start the System Center 2012 - Orchestrator Setup Wizard, on your product media or network share, double-click SetupOrchestrator.exe.



Important

Before you begin setup, close any open programs and ensure that there are no pending restarts on the computer. For example, if you have installed a server role by using Service Manager or have applied a security update, you might have to restart the computer, and then log on to the computer with the same user account to finish the installation of the server role or the security update.



Note

If User Account Control is enabled, then you will be prompted to verify that you want to allow the setup program to run. This is because it requires administrative access to make changes to the system.

2. On the main page of the System Center 2012 - Orchestrator Setup Wizard, click Install.



Warning

If Microsoft .NET Framework 3.5 Service Pack 1 is not installed on your computer, a dialog box appears asking if you want to install .NET Framework 3.5 SP1. Click **Yes** to proceed with the installation.

3. On the **Product registration** page, provide the name and company for the product registration, and then click Next.



Note

For this evaluation release, a product key is not required.

- 4. On the Please read this license agreement page, review and accept the Microsoft Software License Terms, and then click Next.
- 5. On the Select features to install page, ensure that Management Server is the only feature selected, and then click Next.
- 6. Your computer is checked for required hardware and software. If your computer meets all of the requirements, the All prerequisites are installed page appears. Click Next and proceed to the next step.

If a prerequisite is not met, a page displays information about the prerequisite that has not been met and how to resolve the issue. Use the following steps to resolve the failed prerequisite check:

a. Review the items that did not pass the prerequisite check. For some requirements, such as Microsoft .NET Framework 4, you can use the link provided in the Setup Wizard to install the missing requirement. The Setup Wizard can install or configure other prerequisites, such as the Internet Information Services (IIS) role.



Warning

If you enable prerequisites during setup, such as Microsoft .NET Framework 4, your computer can require a restart. If you restart your computer, you must run setup again from the beginning.

- b. After you resolve the missing prerequisites, click Verify prerequisites again.
- c. Click Next to continue.
- On the Configure the service account page, enter the user name and password for the Orchestrator service account. Click Test to verify the account credentials. If the credentials are accepted, then click Next.
- 8. On the Configure the database server page, enter the name of the server and the name of the instance of Microsoft SQL Server that you want to use for Orchestrator. You can also specify whether to use Windows Authentication or SQL Server Authentication, and whether to create a new database or use an existing database. Click Test Database Connection to verify the account credentials. If the credentials are accepted, click Next.
- On the Configure the database page, select a database or create a new database, and then click Next.
- On the Configure Orchestrator management group page, accept the default configuration or enter the name of the Active Directory user group to manage Orchestrator, and then click Next.
- 11. On the **Select the installation location** page, verify the installation location for Orchestrator, and then click **Next**.
- 12. On the **Microsoft Update** page, optionally indicate whether you want to use the Microsoft Update services to check for updates, and then click **Next**.
- 13. On the **Help improve Microsoft System Center Orchestrator** page, optionally indicate whether you want to participate in the **Customer Experience Improvement Program** or **Error Reporting**, and then click **Next**.
- 14. Review the **Installation summary** page, and then click **Install**.
 - The **Installing features** page appears and displays the installation progress.
- 15. On the **Setup completed successfully** page, optionally indicate whether you want to start Runbook Designer, and then click **Close** to complete the installation.

See Also

Install Individual Orchestrator Features

How to Install a Runbook Server for System Center 2012 - Orchestrator

Use the following steps to install an System Center 2012 - Orchestrator runbook server.

To install an Orchestrator runbook server

1. On the server where you want to install an Orchestrator runbook server, start the System Center 2012 - Orchestrator Setup Wizard.

To start the **System Center 2012 - Orchestrator Setup Wizard**, on your product media or network share, double-click **SetupOrchestrator.exe**.

Note

Before you begin setup, close any open programs and ensure that there are no pending restarts on the computer. For example, if you have installed a server role by using Service Manager or have applied a security update, you might have to restart the computer, and then log on to the computer with the same user account to finish the installation of the server role or the security update.

2. On the main setup page, under **Standalone installations**, click **Runbook server**.



Warning

If Microsoft .NET Framework 3.5 Service Pack 1 is not installed on your computer, a dialog box appears asking whether you want to install .NET Framework 3.5 SP1. Click **Yes** to proceed with the installation.

3. On the **Product registration** page, provide the name and company for the product registration, and then click Next.



Note

For this evaluation release, a product key is not required.

- 4. On the Please read this license agreement page, review and accept the Microsoft Software License Terms, and then click Next.
- 5. Your computer is checked for required hardware and software. If your computer meets all of the requirements, the All prerequisites are installed page appears. Click Next and proceed to the next step.

If a prerequisite is not met, a page displays information about the prerequisite that has not been met and how to resolve the issue. Use the following steps to resolve the failed prerequisite check:

Review the items that did not pass the prerequisite check. For some requirements, such as Microsoft .NET Framework 4, you can use the link provided in the Setup Wizard to install the missing requirement. The Setup Wizard can install or configure other prerequisites, such as the Internet Information Services (IIS) role.



Warning

If you enable prerequisites during setup, such as Microsoft .NET Framework 4, your computer can require a restart. If you restart your computer, you must run setup again from the beginning.

- b. After you resolve the missing prerequisites, click Verify prerequisites again.
- c. Click Next to continue.
- 6. On the Configure the service account page, enter the user name and password for the Orchestrator service account. Click Test to verify the account credentials. If the credentials are accepted, click Next.
- 7. On the **Configure the database server** page, enter the name of the database server associated with your Orchestrator management server. You can also specify whether to use Windows Authentication or SQL Server Authentication, and whether to create a new database or use an existing database. Click Test Database Connection to verify the

- account credentials. If the credentials are accepted, click **Next**.
- 8. On the **Configure the database** page, select the Orchestrator database for your deployment, and then click Next.
- 9. On the **Select the installation location** page, verify the installation location for Orchestrator, and then click Next.
- 10. On the Microsoft Update page, optionally indicate whether you want to use the Microsoft Update services to check for updates, and then click **Next**.
- 11. On the Help improve Microsoft System Center Orchestrator page, optionally indicate whether you want to participate in the Customer Experience Improvement Program or Error Reporting, and then click Next.
- 12. Review the **Installation summary** page, and then click **Install**.
 - The **Installing features** page appears and displays the installation progress.
- 13. On the Setup completed successfully page, optionally indicate whether you want to start the Runbook Designer, and then click **Close** to complete the installation.

Install Individual Orchestrator Features

How to Install the Orchestrator Web Service

Use the following steps to install the System Center 2012 - Orchestrator web service.

To install the Orchestrator web service

1. On the server where you want to install the Orchestrator web service, start the System Center 2012 - Orchestrator Setup Wizard.

To start the System Center 2012 - Orchestrator Setup Wizard, on your product media or network share, double-click **SetupOrchestrator.exe**.



Note

Before you begin the installation of the Orchestrator web service, close any open programs and ensure that there are no pending restarts on the computer. For example, if you have installed a server role by using Service Manager or have applied a security update, you might have to restart the computer, and then log on to the computer with the same user account to finish the installation of the server role or the security update.

2. On the main System Center 2012 - Orchestrator Setup Wizard page, click Install.



Warning

If Microsoft .NET Framework 3.5 Service Pack 1 is not installed on your computer, a dialog box appears asking if you want to install .NET Framework 3.5 SP1. Click **Yes** to proceed with the installation.

3. On the **Product registration** page, provide the name and company for the product registration, and then click **Next**.



For this evaluation release, a product key is not required.

- 4. On the Please read this license agreement page, review and accept the Microsoft Software License Terms, and then click Next.
- 5. Your computer is checked for required hardware and software. If your computer meets all of the requirements, the All prerequisites are installed page appears. Click Next and proceed to the next step.
 - If a prerequisite is not met, a page displays information about the prerequisite that has not been met and how to resolve the issue. Use the following steps to resolve the failed prerequisite check:
 - a. Review the items that did not pass the prerequisite check. For some requirements, such as Microsoft .NET Framework 4, you can use the link provided in the Setup Wizard to install the missing requirement. The Setup Wizard can install or configure other prerequisites, such as the Internet Information Services (IIS) role.



Warning

If you enable prerequisites during setup, such as Microsoft .NET Framework 4, your computer can require a restart. If you restart your computer, you must run setup again from the beginning.

- b. After you resolve the missing prerequisites, click **Verify prerequisites again**.
- c. Click Next to continue.
- 6. On the Configure the service account page, enter the user name and password for the Orchestrator service account. Click Test to verify the account credentials. If the credentials are accepted, click Next.
- 7. On the Configure the database server page, enter the name of the database server associated with your Orchestrator management server. You can also specify whether to use Windows Authentication or SQL Server Authentication, and whether to create a new database or use an existing database. Click Test Database Connection to verify the account credentials. If the credentials are accepted, click **Next**.
- 8. On the Configure the database page, select the Orchestrator database for your deployment, and then click Next.
- 9. On the Configure the port for the web service page, verify the port numbers for the Orchestrator web service and the Orchestration console, and then click Next.
- 10. On the **Select the installation location** page, verify the installation location for Orchestrator, and then click Next.
- 11. On the Microsoft Update page, optionally indicate whether you want to use the Microsoft Update services to check for updates, and then click Next.
- 12. On the Help improve Microsoft System Center Orchestrator page, optionally indicate whether you want to participate in the Customer Experience Improvement Program or Error Reporting, and then click Next.
- 13. Review the **Installation summary** page, and then click **Install**.

The **Installing features** page appears and displays the installation progress.

14. On the **Setup completed successfully** page, optionally indicate whether you want to start the Runbook Designer, and then click **Close** to complete the installation.

See Also

Install Individual Orchestrator Features

How to Install the Runbook Designer for System Center 2012 - Orchestrator

Use the following steps to install the System Center 2012 - Orchestrator Runbook Designer on a single computer.

To install the Orchestrator Runbook Designer on a single computer

1. On the server where you want to install the Orchestrator Runbook Designer, start the System Center 2012 - Orchestrator Setup Wizard.

To start the System Center 2012 - Orchestrator Setup Wizard, on your product media or network share, double-click **SetupOrchestrator.exe**.



Note

Before you begin the install of the Runbook Designer, close any open programs and ensure that there are no pending restarts on the computer. For example, if you have installed a server role by using Service Manager or have applied a security update, you might have to restart the computer, and then log on to the computer with the same user account to finish the installation of the server role or the security update.

2. On the main System Center 2012 - Orchestrator Setup Wizard page, click Runbook Designer.



Warning

If Microsoft .NET Framework 3.5 Service Pack 1 is not installed on your computer, a dialog box appears asking if you want to install .NET Framework 3.5 SP1. Click **Yes** to proceed with the installation.

- 3. Your computer is checked for required hardware and software. If your computer meets all of the requirements, proceed to the next step.
 - If a prerequisite is not met, a page displays information about the prerequisite that has not been met and how to resolve the issue. Use the following steps to resolve the failed prerequisite check:
 - a. Review the items that did not pass the prerequisite check. For some requirements, such as Microsoft .NET Framework 4, you can use the link provided in the Setup Wizard to install the missing requirement. The Setup Wizard can install or configure other prerequisites, such as the Internet Information Services (IIS) role.
 - b. After you resolve the missing prerequisites, click Verify prerequisites again.
 - c. Click Next to continue.
- 4. On the **Select the installation location** page, verify the installation location for

Orchestrator, and then click **Next**.

- 5. Review the **Installation summary** page, and then click **Install**.
 - The **Installing features** page appears and displays the installation progress.
- 6. On the Setup completed successfully page, optionally indicate whether you want to start the Runbook Designer, and then click Close to complete the installation.

To connect a Runbook Designer to a management server

1. In the Runbook Designer, select the Connect to a server icon in the navigation pane under the **Connections** pane.



Note

If the Runbook Designer is connected to another management server, the Connect to a server icon is disabled. Click the Disconnect icon before you connect to a different management server.

2. In the System Center Orchestrator 2012 Connection dialog box, enter the name of the server that hosts your Orchestrator management server, and then click **OK**.

To enable network discovery

- 1. On the desktop of your computer running Windows server, click Start, click Control Panel, click Network and Internet, click Network and Sharing Center, click Choose Home group and Sharing Options, and then click Change advanced sharing settings.
- 2. To change the **Domain** profile, if needed, click the **Arrow** icon to expand the section options and make any necessary changes.
- 3. Select Turn on network discovery, and then click Save changes. If you are prompted for an administrator password or confirmation, type the password or provide confirmation.

See Also

Install Individual Orchestrator Features

Install with the Orchestrator Command Line Install Tool

To install Orchestrator at a command prompt, use the SetupOrchestrator.exe executable program with the command-line options in the following table.

Option	Description
/Silent	Installation is performed without displaying a dialog box.
/Uninstall	Product is uninstalled. This option is performed silently.

Option	Description
/Key:[Product Key]	Specifies the product key. If no product key is specified, Orchestrator is installed as an evaluation edition.
/ServiceUserName:[User Name]	Specifies the user account for the Orchestrator Management Service. This value is required if you are installing Management Server, Runbook Server, or web services.
/ServicePassword:[Password]	Specifies the password for the user account for the Orchestrator Management Service. This value is required if you are installing Management Server, Runbook Server, or web services.
/Components:[Feature 1, Feature 2,]	Specifies the features to install. Possible values are ManagementServer, RunbookServer, RunbookDesigner, WebComponents, and All.
/InstallDir:[Path]	Specifies the path to install Orchestrator. If no path is specified, C:\Program Files (x86)\Microsoft System Center 2012\Orchestrator is used.
/DbServer:[Computer[\Instance]]	Specifies the computer name and instance of the database server. This value is required if you are installing Management Server, Runbook Server, or web services.
/DbUser:[User Name]	Specifies the user account to access the database server. This value is only required for SQL Authentication. If Windows Authentication is used, no value should be specified.
/DbPassword:[Password]	Specifies the password for the user account to access the database server. This value is only required for SQL Authentication. If Windows Authentication is used, then no value should be specified.
/DbNameNew:[Database Name]	Specifies the database name if a new database is being created. Cannot be used with DbNameExisting.
/DbNameExisting:[Database Name]	Specifies the database name if an existing database is being used. Cannot be used with

Option	Description
	DbNameNew.
/WebServicePort:[Port]	Specifies the port to use for the web service. Required if web services are installed.
/WebConsolePort:[Port]	Specifies the port to use for the Orchestrator console. Required if web services are installed.
/OrchestratorUsersGroup:[Group SID]	Specifies the SID of the domain or local group that will be granted access to Management server. If no value is specified, the default local group is used.
/OrchestratorRemote	Specifies that remote access should be granted to the Runbook Designer.
/UseMicrosoftUpdate:[0 1]	Specifies whether to opt in for Microsoft Update. A value of 1 will opt in. A value of 0 does not change the current opt in status of the computer.
/SendCEIPReports:[0 1]	Specifies that Orchestrator should send CEIP (Customer Experience Improvement Program) reports to Microsoft. A value of 1 opts in. A value of 0 does not change the current opt-in status of the computer.
/EnableErrorReporting:[value]	Specifies that Orchestrator should send program error reports to Microsoft. Possible values are always, queued, and never.

Install Orchestrator

Use Sysprep to Deploy Orchestrator

You can deploy Orchestrator using Sysprep. This enables you to deploy any component in Orchestrator in a distributed environment in an automatic process.

For Orchestrator components you can create a Sysprep image by performing the following steps:

- 1. Prepare the Windows 2008 R2 image
- 2. Create the Orchestrator answer file for sysprep
- 3. Install Orchestrator using sysprep.

Prepare the Windows 2008 R2 image

Use the following steps to prepare the Windows 2008 R2 image.

To prepare the Windows 2008 R2 image

- 1. Install Windows Server 2008 R2.
- 2. Install .NET Framework 4 from http://go.microsoft.com/fwlink/?LinkId=246814. (This is only required for the web feature components of Orchestrator.)

Create the Orchestrator answer file for sysprep

Before you can use the Sysprep tool to install Orchestrator on Windows Server 2008 R2, install Orchestrator as part of the Sysprep process using an answer file.

See <u>Sample Orchestrator.xml file</u> for sample unattend.xml file. You can customize this sample file and import it into the Windows System Image Manager.

To create the answer file

- 1. Create the Orchestrator.xml unattend file using the sample provided.
- 2. Copy the file to %systemdrive%\windows\system32\sysprep.
- Create the Orchestrator batch file that will install the Orchestrator components on this
 computer. An example of this file is available in <u>Sample Orchestrator.xml file</u>. This is
 referred to in the Orchestrator.xml file. See <u>Install with the Orchestrator Command Line</u>
 <u>Install Tool</u> for the available command line options that can be used to install
 Orchestrator.
- 4. Run the following command:

```
sysprep /generalize /oobe /shutdown
/unattend:%systemdrive%\windows\system32\sysprep\Orchestrator
.xml
```

Install Orchestrator using sysprep

You now have a Windows 2008 R2 Sysprep image that you can use to automatically deploy Orchestrator in the environment.

For information on creating a SQL Sysprep image for use with Orchestrator, refer to http://go.microsoft.com/fwlink/?LinkId=246815.

Sample Orchestrator.xml file

This is a sample Orchestrator.xml to be used for deploying Orchestrator with sysprep. Customize this using the Windows System Image Manager (available in the Windows Automated Installation Kit, at http://go.microsoft.com/fwlink/?LinkId=246813).

```
<?xml version="1.0" encoding="utf-8"?>
<unattend xmlns="urn:schemas-microsoft-com:unattend">
```

```
<settings pass="oobeSystem">
        <component name="Microsoft-Windows-Shell-Setup" processorArchitecture="amd64"</pre>
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <AutoLogon>
                <Password>
                    <Value>password</Value>
                    <PlainText>true</PlainText>
                </Password>
                <LogonCount>1</LogonCount>
                <Enabled>true</Enabled>
                <Username>Administrator
            </AutoLogon>
            <FirstLogonCommands>
                <SynchronousCommand wcm:action="add">
                    <CommandLine>cmd /c %systemdrive%\sco\install.bat</CommandLine>
                    <Order>1</Order>
                    <RequiresUserInput>false</RequiresUserInput>
                </SynchronousCommand>
            </FirstLogonCommands>
        </component>
   </settings>
   <cpi:offlineImage</pre>
cpi:source="wim:c:/windowsenterprise/extracted/sources/install.wim#Windows Server 2008 R2
SERVERENTERPRISE" xmlns:cpi="urn:schemas-microsoft-com:cpi" />
</unattend>
```

This is a sample install.bat file that is referenced in the Orchestrator.xml unattend file for the FirstLogonCommand. Create this batch file in the %systemdrive%\sco directory along with the Orchestrator setup files. This file can be customized by using the command line install tool. For more information, see Install with the Orchestrator Command Line Install Tool.

```
%systemdrive%\sco\setup\setup.exe /Silent /ServiceUserName:%computername%\administrator /ServicePassword:password /Components:All /DbServer:%computername% /DbPort:1433 /DbNameNew:OrchestratorSysPrep /WebConsolePort:82 /WebServicePort:81 /OrchestratorRemote /UseMicrosoftUpdate:1 /SendCEIPReports:1 /EnableErrorReporting:always
```

Perform Post-Installation Tasks

This section describes the tasks you can perform after a successful installation of System Center 2012 - Orchestrator.

Post-installation tasks

How to Install an Integration Pack

Describes how to register and deploy Orchestrator integration packs.

How to Install GnuPG

Describes how to install GnuPG.

Migrate Opalis Policies to Orchestrator

Describes how to migrate Opalis Policies to Orchestrator runbooks.

How to Configure the Orchestrator Web Service to use HTTPS

Describes how to secure the Orchestrator web service by using HTTPS.

How to Uninstall and Unregister an Integration Pack

Describes how to remove integration packs from Orchestrator.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Deployment Overview
- Plan Your Orchestrator Deployment
- Install Orchestrator
- Troubleshoot Your Orchestrator Installation

How to Install an Integration Pack

System Center 2012 - Orchestrator includes a set of standard activities that are automatically installed with Orchestrator. You can expand functionality and ability of Orchestrator to integrate platforms and products by Microsoft and other companies by installing integration packs. Each integration pack contains activities that provide unique functions. Microsoft provides integration packs for all of the System Center products, a number of other Microsoft products, and technologies and products from other companies.

Integration packs are available from the <u>Microsoft Download Center</u>. Each integration pack has a guide that provides installation instructions, known issues, and reference information for the activities in that integration pack. To review the current integration pack guides, see <u>Integration Packs for System Center 2012 – Orchestrator</u> in the TechNet Library.

The following procedures contain general instructions that apply to most integration packs. See the relevant integration pack guide for system requirements and any special installation instructions for that integration pack.

Important

System Center 2012 - Orchestrator supports integration packs designed for System Center 2012 - Orchestrator. Integrations packs for Opalis or pre-release versions of System Center 2012 - Orchestrator are not supported.

Important

Orchestrator does not support a downgrade of integration packs. If you have an integration pack that is currently registered or previously registered in Orchestrator, installation fails if you attempt to install an earlier version of the same integration pack. You should test integration packs and upgraded integration packs in a test environment before you implement them in a production environment. If you require a downgrade of an integration pack in a production environment, contact Microsoft Customer Support for assistance.

Registering and deploying an integration pack

After you download the integration pack, you register the integration pack file with the Orchestrator management server, and then deploy it to runbook servers and computers that have the Runbook Designer installed. For more information about how to install a specific integration pack, see the guide for that integration pack.

When you install an upgrade of an integration pack, you must first uninstall any earlier version of the integration pack from all runbook servers and Runbook Designers. You then register and deploy the upgrade of the integration pack. If you do not uninstall the previous version of the integration pack prior to registering and deploying the upgrade version, the upgrade version will fail.

To register an integration pack

1. On the management server, copy the .OIP file for the integration pack to a local hard drive or network share.



Confirm that the file is not set to Read Only to prevent unregistering the integration pack at a later date.

- 2. Start the **Deployment Manager**.
- 3. In the navigation pane of the Deployment Manager, expand Orchestrator Management Server, right-click Integration Packs to select Register IP with the Management Server. The Integration Pack Registration Wizard opens.
- Click Next.
- 5. In the Select Integration Packs or Hotfixes dialog box, click Add.

- 6. Locate the .OIP file that you copied locally from step 1, click Open, and then click Next.
- 7. In the Completing the Integration Pack Wizard dialog box, click Finish.
- 8. On the **End User Agreement** dialog box, read the Microsoft Software License Terms, and then click Accept.

The Log Entries pane displays a confirmation message when the integration pack is successfully registered.

To deploy an integration pack

- 1. In the navigation pane of **Deployment Manager**, right-click **Integration Packs**, click Deploy IP to Action Server or Client.
- 2. Select the integration pack that you want to deploy, and then click **Next**.
- 3. Enter the name of the runbook server or computers with the Runbook Designer installed, on which you want to deploy the integration pack, click Add, and then click Next.
- 4. Continue to add additional runbook servers and computers running the Runbook Designer, on which you want to deploy the integration pack. Click Next.
- 5. In the **Installation Options** dialog box, configure the following settings.
- 6. To choose a time to deploy the integration pack, select the **Schedule installation** check box, and then select the time and date from the **Perform installation** list.
- 7. Click one of the following:
 - Stop all running runbooks before installing the integration pack to stop all running runbooks before deploying the integration pack.
 - Install the Integration Packs without stopping the running Runbooks to install the integration pack without stopping any running runbooks.
- Click Next.
- 9. In the Completing Integration Pack Deployment Wizard dialog box, click Finish.
- 10. When the integration pack is deployed, the **Log Entries** dialog box displays a confirmation message.



Warning

If you did not configure a deployment schedule, the integration pack deploys immediately to the computers that you specified. If you configured a deployment schedule, verify that the deployment occurred by verifying the event logs after the scheduled time has passed.

To upgrade an integration pack

- 1. On all computers that have a runbook server or Runbook Designer installed, uninstall any earlier version of the integration pack. You can achieve this by one of following steps:Register and deploy the upgraded integration pack as described above in To register an integration pack and To deploy an integration pack.
 - Log on into each computer and uninstall the integration pack from Programs and Features in Control Panel.

- On the management server, start the Deployment Manager, and then right click on the deployed integration pack for each Runbook Server or Runbook Designer computer and click Uninstall Integration Pack or Hotfix.
- Register and deploy the upgraded integration pack as described above in "To register an
 integration pack" and "Register and deploy the upgraded integration pack as described
 above in To register an integration pack and To deploy an integration pack.
- 3. Deploy the integration pack upgrade as described above in <u>To deploy an integration</u> pack.

Perform Post-Installation Tasks

How to Install GnuPG

GnuPG is an open source program used by the standard activities **PGP Encrypt File** and **PGP Decrypt File** to encrypt and decrypt files. The following procedure describes how to install this executable program and associated file on a runbook server or computer that is running the Runbook Designer.

To install GnuPG

- 1. Download **gpg.exe** and **iconv.dll**, version 1.4.10 or later, from <u>GnuPG</u>.
- 2. Save **gpg.exe** and **iconv.dll** to the <System drive>:\Program Files (x86)\Common Files\Microsoft System Center 2012\Orchestrator\Extensions\Support\Encryption folder on each runbook server and computer that is running the Runbook Designer.

See Also

Perform Post-Installation Tasks

Migrate Opalis Policies to Orchestrator

You can migrate Policies developed for Opalis Integration Server 6.3 to System Center 2012 - Orchestrator. These migrated Policies are converted to runbooks, and might require additional authoring to function correctly. If you are using an earlier release than Opalis Integration Server 6.3, you must upgrade to version Opalis Integration Server 6.3 before you can migrate your Policies to Orchestrator.

This section describes the tasks you perform to migrate Opalis Policies to System Center 2012 - Orchestrator.

Opalis Policy migration

Policy Migration Overview

Describes the steps to successfully migrate your Opalis Policies.

• Planning your Opalis Policy Migration

Describes issues to consider before you migrate your Opalis Polices.

How to Migrate Opalis Policies to Orchestrator

Describes how to migrate your Opalis Policies to Orchestrator.

Modify Migrated Orchestrator Runbooks

Describes required modifications for newly migrated Orchestrator runbooks.

• How to Test Migrated Runbooks

Describes how to test your newly migrated Orchestrator runbooks.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Perform Post-Installation Tasks
- How to Install an Integration Pack
- How to Install GnuPG
- How to Change the Orchestrator Database
- How to Change the Orchestrator Users Group

Policy Migration Overview

Use the following guide to migrate Policies from Opalis Integration Server 6.3 to Orchestrator.

Task	Information
1. Plan your Opalis Policy Migration.	Planning your Opalis Policy Migration
2. Migrate your Opalis Policies and import them into Orchestrator.	How to Migrate Opalis Policies to Orchestrator
3. Update your newly imported runbooks.	Modify Migrated Orchestrator Runbooks
4. Test your new runbooks with Orchestrator.	How to Test Migrated Runbooks

See Also

Migrate Opalis Policies to Orchestrator

Planning your Opalis Policy Migration

The following sections provide important details when you migrate Opalis 6.3 Policies to System Center 2012 - Orchestrator.

Topic	Information
Summary of supported migration paths for Opalis Policies.	Supported Opalis Policy Migration Paths
System requirements for migrating Opalis Policies to Orchestrator.	System Requirements
Modify Opalis Policies that contain special characters.	Modify Opalis Policies that Contain Special Characters

Migrate Opalis Policies to Orchestrator

Supported Opalis Policy Migration Paths

The following table summarizes the valid migration paths of Opalis Policies to System Center 2012 - Orchestrator.

Product	Upgrade path
Opalis Robot, all versions	Not supported.
Opalis Integration Server, all versions 5.2 and earlier	Not supported.
Opalis Integration Server 5.3, 5.4, 5.45, 5.5, 5.51, 5.52,5.6, 5.6.1, 5.6.2, 6.0, or 6.2	Upgrade the earlier version of Opalis to Opalis Integration Server 6.3 and then migrate an Opalis Integration Server 6.3 Policy to Orchestrator.
Opalis Integration Server 6.3	Export Policies from Opalis 6.3 and import them into Orchestrator as runbooks.

For information about how to upgrade earlier versions of Opalis Integration Server to Opalis Integration Server 6.3, see Opalis Integration Server 6.3.

See Also

Migrate Opalis Policies to Orchestrator

System Requirements

The operating systems supported for Orchestrator differ from the supported operating systems for Opalis Integration Server 6.3. Although both products run on Windows Server 2008 R2, they cannot be installed on the same system. There is no common database support for both products.

Product	Supported operating systems and software
Opalis Integration Server 6.3	Windows Server 2003 (32-bit) Service Pack 2
	Windows Server 2003 R2 (32-bit) Service Pack
	2
	Windows Server 2008 (32-bit or 64-bit)
	Windows Server 2008 R2
	Microsoft SQL Server 2005
	SQL Server 2008
	Oracle Server versions 9.2, 10.x, and 11
System Center 2012 - Orchestrator	Windows Server 2008 R2
	SQL Server 2008 R2

Migrate Opalis Policies to Orchestrator

Modify Opalis Policies that Contain Special Characters

Occasionally Opalis Policies contain special characters. These characters are in objects and links between objects in Opalis Policies. You must remove these characters from Opalis Policies before they are migrated to Orchestrator. The following Microsoft SQL Server query runs against the Opalis database to locate Policies that contain special characters. The results provide information about the objects and links where special characters appear.

When a **Name** or **Description** field is located that contains these special characters, select the field and use CTRL+A to select its entire contents. Delete the field contents, and then replace the contents with supported characters.

```
Policies.Name,
Objects.Name,
Objects.Description

FROM
Objects join
Policies
on Objects.ParentID=Policies.UniqueID

where
ASCII(Objects.Name) < 32
or ASCII(CAST(Objects.Description as nvarchar(max))) < 32 and
Policies.Deleted is NULL
```

Migrate Opalis Policies to Orchestrator

How to Migrate Opalis Policies to Orchestrator

You can migrate existing Opalis Integration Server 6.3 Policies to System Center 2012 -Orchestrator. You have to export the Policies from Opalis Integration Server 6.3 and then import them into Orchestrator, which converts them into runbooks.

To export Opalis Policies

- 1. Identify the Opalis Integration Server 6.3 Policies that you want to migrate to Orchestrator. Export the Policies you want and any required global settings.
- 2. In the Opalis Integration Server 6.3 Client, click **Actions**, and then click **Export**.



If exports take a long time to complete, they will take a long time to import. Consider breaking up an export into several smaller exports by selecting groups of Policies in the Opalis Integration Server 6.3 Client.

To import Opalis Policies into Orchestrator

- 1. Copy the export file to the computer where Orchestrator Runbook Designer is installed.
- 2. In the Runbook Designer, click Actions, and then click Import. Select the exported Opalis file as the source for the import.
- 3. The **Import** dialog box provides a box for a password. When you import Orchestrator runbooks, leave the **Password** box blank, and then click **Finish**.

See Also

Perform Post-Installation Tasks

Modify Migrated Orchestrator Runbooks

Opalis Policies and Orchestrator runbooks do not share all settings or features. Migrated runbooks must be modified so that the runbook functions correctly in Orchestrator. For each migrated runbook, you should update the runbook to meet current requirements.

This topic provides information about how to update migrated Orchestrator runbooks.

Opalis Policy Mode setting

Opalis Integration Server 6.3 provides two modes for Policies that defined the behavior of the publish and subscribe data bus. To access this property, right-clicking a **Policy** tab in the Opalis Integration Server 6.3 Client to select Properties. The Policy Mode setting is located on the Run Behavior tab. If the Run in pipeline mode box is not selected, the workflow is configured for "legacy" mode.

Legacy mode was retained in Opalis Integration Server 6.3 to provide backwards compatibility with older versions of Opalis software. In Opalis Integration Server 6.3, the default policy mode for new workflows is "pipeline."

Orchestrator does not provide a runbook mode setting. The Policy Mode property of an Opalis Policy is now the **Job Concurrency** property in Orchestrator. Runbooks in Orchestrator use a data model compatible with Opalis Integration Server 6.3 pipeline mode.

You can import Opalis Integration Server 6.3 Policies that used legacy mode successfully into Orchestrator. However, the legacy mode is not compatible with Orchestrator. You can identify legacy mode runbooks by looking at the toolbar of the imported runbook. If the runbook is Checked In, the runbook was an Opalis Integration Server 6.3 pipeline workflow. If the runbook is Checked Out, the Opalis Integration Server 6.3 workflow was a legacy workflow.

The following table summarizes the Opalis Integration Server 6.3 workflow mode settings and show how these workflows are treated when they are migrated to Orchestrator runbooks.

Opalis policy mode	Migrated runbook property
Legacy	Runbooks show Checked Out after being imported and have to be checked in to update the runbook.
Pipeline mode	Runbooks show Checked In after being imported.

Unsupported structures

When you check in a runbook that was a legacy Policy in Opalis, the Runbook Designer shows a warning if the runbook contains a structure that is not valid. You must resolve these warnings before the runbook functions correctly.



Note

The check-in succeeds even if a warning is produced. Check out the runbook to correct the structure that is not valid.

Multiple starting points

Runbooks can only contain one activity as a starting point. If the migrated runbook contains multiple starting activities, check-in produces a warning.

Update the runbook to contain a single starting activity.

Cycles

Runbooks cannot contain smart links that originate with one runbook activity and reference an earlier runbook activity. These runbook structures are called cycles. Cycles are supported in Opalis Integration Server 6.3 legacy mode. Orchestrator does not support runbooks that contain cycles. If you check in a runbook that contains a cycle, you receive a warning.

Update the runbook so it does not contain a cycle.

Opalis ROI setting

Opalis Integration Server 6.3 provided a Policy property to track return on investment (ROI). To can view this runbook property, right-click a **Policy** tab in the Opalis Integration Server Client to select **Properties**. The ROI settings for a given runbook are located on the **ROI** tab.

The **ROI** is not supported in Orchestrator. Orchestrator ignores these settings if they are present in imported Opalis Integration Server 6.3 workflows.

Opalis legacy objects

Opalis Integration Server 6.3 provides workflow objects known as legacy objects. These objects support older versions of Opalis software for backwards compatibility. Orchestrator does not provide support for legacy objects. In the Runbook Designer, references to these legacy objects in imported runbooks are marked with an activity icon that contains a question mark (?). Orchestrator provides equivalent standard activities for the legacy objects. Update migrated runbooks to remove the legacy object and replace it with an Orchestrator equivalent. The following table lists a suitable replacement for Opalis Integration Server 6.3 legacy objects.

Opalis legacy object	Orchestrator activity or resource
Manage Text File	Orchestrator Text File Management category (Append Line, Delete Line, Find Text, Get Lines, Insert Line, Read Line, and Search and Replace Text activities)
Create Folder	Create Folder
Delete Folder	Delete Folder
Copy File	Copy File
Delete File	Delete File
Move File	Move File
Rename File	Rename File
Get File Status	Get File Status
Monitor File	Monitor File
Monitor Folder	Monitor Folder
Filter Email	Run .NET Script or Orchestrator Integration Toolkit
Process Email	Run .NET Script or Orchestrator Integration Toolkit
Read Email	Run .NET Script or Orchestrator Integration Toolkit

Opalis legacy object	Orchestrator activity or resource
Filter Exchange Email	Community Integration Pack
Process Exchange Email	Community Integration Pack
Read Exchange Email	Community Integration Pack

Opalis Policy objects not supported in Orchestrator

There are a small number of Opalis Integration Server Policy objects that do not have an equivalent Orchestrator runbook activity. The Runbook Designer marks these references to unsupported objects with a question mark (?). The following table lists the unsupported Policy objects.

Opalis object	Details
Send Page	Infrequently used and out-of-date.
Purge Event Log	Infrequently used and out-of-date.
Send Pop-Up	Unsupported.
Monitor Event Log Capacity.	Infrequently used and out-of-date. Replaced by functionality found in System Center 2012 – Operations Manager.
Monitor Performance	Infrequently used and out-of-date. Replaced by functionality found in Operations Manager.
Disconnect Dial-Up	Infrequently used and out-of-date.
Get Dial-Up Status	Infrequently used and out-of-date.
Wait	Only meaningful in Opalis Integration Server 6.3 legacy mode runbooks. Junction is the closest Orchestrator activity.

Opalis Policies that use missing objects

Run the following SQL Server query against either the Opalis data store or the Orchestrator database to identify the Opalis Policies that contain objects that are no longer available in Orchestrator. This query returns both the Opalis Policy name and the name of the object in the Policy. Any Policy identified by this query must be updated after it has been imported into Orchestrator to remove the reference to the deprecated object.

```
Select

policies.[Name] as [Policy Name],

objects.[Name] as [Object Name]
```

```
From
      [Objects] objects join
      [Policies] policies
      on objects.[ParentID]=policies.[UniqueID]
Where
  objects.objecttype = '2081B459-88D2-464A-9F3D-27D2B7A64C5E' or
  objects.objecttype = '6F0FA888-1969-4010-95BC-C0468FA6E8A0' or
  objects.objecttype = '8740DB49-5EE2-4398-9AD1-21315B8D2536' or
  objects.objecttype = '19253CC6-2A14-432A-B4D8-5C3F778B69B0' or
  objects.objecttype = '9AB62470-8541-44BD-BC2A-5C3409C56CAA' or
  objects.objecttype = '292941F8-6BA7-4EC2-9BC0-3B5F96AB9790' or
  objects.objecttype = '98AF4CBD-E30E-4890-9D26-404FE24727D7' or
  objects.objecttype = '2409285A-9F7E-4E04-BFB9-A617C2E5FA61' or
  objects.objecttype = 'B40FDFBD-6E5F-44F0-9AA6-6469B0A35710' or
  objects.objecttype = '9DAF8E78-25EB-425F-A5EF-338C2940B409' or
  objects.objecttype = 'B5381CDD-8498-4603-884D-1800699462AC' or
  objects.objecttype = 'FCA29108-14F3-429A-ADD4-BE24EA5E4A3E' or
  objects.objecttype = '7FB85E1D-D3C5-41DA-ACF4-E1A8396A9DA7' or
  objects.objecttype = '3CCE9C71-51F0-4595-927F-61D84F2F1B5D' or
  objects.objecttype = '96769C11-11F5-4645-B213-9EC7A3F244DB' or
  objects.objecttype = '6FED5A55-A652-455B-88E2-9992E7C97E9A' or
  objects.objecttype = '9C1DF967-5A50-4C4E-9906-C331208A3801' or
  objects.objecttype = 'B40FDFBD-6E5F-44F0-9AA6-6469B0A35710' or
  objects.objecttype = '829A951B-AAE9-4FBF-A6FD-92FA697EEA91' or
  objects.objecttype = '1728D617-ACA9-4C96-ADD1-0E0B61104A9E' or
  objects.objecttype = 'F3D1E70B-D389-49AD-A002-D332604BE87A' or
  objects.objecttype = '2D907D60-9C25-4A1C-B950-A31EB9C9DB5F' or
  objects.objecttype = ^{1}6A083024-C7B3-474F-A53F-075CD2F2AC0F' or
  objects.objecttype = '4E6481A1-6233-4C82-879F-D0A0EDCF2802' or
  objects.objecttype = 'BC49578F-171B-4776-86E2-664A5377B178'
```

See Also

Migrate Opalis Policies to Orchestrator

How to Test Migrated Runbooks

After you migrate and update your migrated runbooks, test the migrated runbooks to verify that they function correctly. You test runbooks with the Runbook Tester, located in the Runbook Designer. To perform testing successfully, your imported runbook must satisfy the following criteria:

- The runbook does not produce any warnings when checked in.
- The runbook does not use any legacy or missing Opalis Integration Server 6.3 Policy objects.
- The runbook does not contain an **Invoke Runbook** activity that has to run as part of testing.

Observe the runtime characteristics of the runbook. In most cases, the behavior is unchanged from the behavior in Opalis Integration Server 6.3. If there are differences in behavior, these are typically associated with differences in behavior between legacy mode and pipeline mode in Opalis Integration Server 6.3.

See Also

Migrate Opalis Policies to Orchestrator

How to Configure the Orchestrator Web Service to use HTTPS

Use the following steps to configure Secure Sockets Layer (SSL) for the System Center 2012 - Orchestrator web service and Orchestration console.

To configure the Orchestrator web service to use Secure Sockets Layer (SSL)

- Request and install a certificate on the computer where you installed the Orchestrator web service. For guidance about requesting and installing a certificate, see <u>How to</u> <u>implement SSL in IIS</u> on the Microsoft Support website.
- Configure SSL on the machine that hosts the web service and Orchestration console.
 The default port for the web service is port 81, and the default port for the Orchestration console is port 82. You should configure the ports as appropriate for your installation.

You can configure the bindings by performing the following steps:

- a. Open Internet Information Services (IIS) Manager.
- b. In the **Connections** pane, expand the Orchestrator web server, expand **Sites**, and then click **Microsoft System Center 2012 Orchestrator Web Service**.
- c. In the **Actions** pane, click **Bindings**.
- d. In the Site Bindings dialog box, click Add.
- e. In the **Add Site Binding** dialog box, in the **Type** box, select **https** and select your **SSL certificate**.
- f. Specify the **Port** to use. The default of 443 is recommended.
- g. Click OK.
- h. Click Close.
- 3. In the Microsoft System Center Orchestrator 2012 Orchestration Web Service pane, under IIS, double-click SSL settings.

- 4. In the SSL Settings pane, select Require SSL.
- Click Apply.
- 6. Repeat the procedure for **Microsoft System Center 2012 Orchestrator Orchestration Console** using a different port. Port 444 is recommended.

For more information about securing Internet Information Services (IIS) 7, see http://go.microsoft.com/fwlink/p/?LinkId=231416.

To update the Orchestration console web.config file

- 1. On your Orchestrator web server, locate the **web.config** file at **C:\Program Files** (x86)\Microsoft System Center 2012\Orchestrator\Orchestration Console.
- 2. Open web.config in an editor.
- Locate the service URI key, and update the key to connect to the web service through HTTPS. For example: change <add key="ScoServiceUri" value="http://<domain>:81/Orchestrator2012/Orchestrator.svc/"/> to <add key="ScoServiceUri" value=" https://<domain>:443/Orchestrator2012/Orchestrator.svc/"/>.



If you used a port for the web service other than 443, then use that port number.

See Also

Perform Post-Installation Tasks

Troubleshoot Your Orchestrator Installation

The latest troubleshooting information for System Center 2012 - Orchestrator is available in the release notes at Release Notes for System Center 2012 - Orchestrator. The following information provides additional instructions and caveats that you can use during installation to resolve problems you might experience.

Orchestrator log files

If you experience problems during installation, installation log files are located in the folder C:\Users\%USERNAME%\AppData\Local\SCO\LOGS.

If you experience problems when you are running Orchestrator, the product log files are located in the folder C:\ProgramData\Microsoft System Center 2012\Orchestrator\.

Windows Firewall

When you deploy additional Runbook Designer applications to your environment, you might see a failed installation message. To correctly install the Runbook Designer, enable the following firewall rules as they apply to your operating system and deployment configuration.

Windows Firewall with Advanced Security for Windows Server 2012 R2

By default, **Windows Firewall with Advanced Security** is enabled on all Windows Server 2008 R2 computers, and blocks all incoming traffic unless it is a response to a request by the host, or it is specifically allowed. You can explicitly allow traffic by specifying a port number, application name, service name, or other criteria by configuring Windows Firewall with Advanced Security settings.

If you are running Windows Server 2012 R2, enable the following rules to allow all Monitor Event activities to function correctly:

- Windows Management Instrumentation (Async-In)
- Windows Management Instrumentation (DCOM-In)
- Windows Management Instrumentation (WMI-In)

Automated deployment

When a runbook server or Runbook Designer is installed behind a firewall, specific firewall rules are required between the remote computers that are used to deploy the runbook server and Runbook Designer. An additional rule is required for the remote connection between the Runbook Designer and the runbook server to allow the Orchestrator management service to accept remote connections. If you are using the **Monitor WMI** task, the runbook server requires a special firewall rule on the computer that uses PolicyModule.exe.

Enable the following firewall rules on your computer:

Firewall rule between the Runbook Designer and the Orchestrator management server

Operating system	Firewall rule
64-bit	%ProgramFiles (x86)%\Microsoft System Center 2012 R2\Orchestrator\Management Server\OrchestratorManagementService.exe
32-bit	%ProgramFiles%Microsoft System Center 2012 R2\Orchestrator\Management Server\OrchestratorManagementService.exe

Firewall rules between remote computers

Operating system	Firewall rules	
Windows Server 2008 R2	 File and Printer Sharing Windows Management Instrumentation (WMI) Program rule for OrchestratorRemotingService to accept remote connections. This rule must be enabled through the Advanced Firewall mode: 	
	 %SystemRoot%\SysWOW64\OrchestratorRemotingService.exe (for a 64-bit operating system) 	

Operating system	Firewall rules	
	 %SystemRoot%\System32\OrchestratorRemotingService.exe (for a 32-bit operating system) 	

Firewall rules between the runbook server and the computer that uses PolicyModule.exe

Operating system	Firewall rule
64-bit	%ProgramFiles (x86)%\Microsoft System Center 2012 R2\Orchestrator\Runbook Server\PolicyModule.exe
32-bit	%ProgramFiles\Microsoft System Center 2012 R2\Orchestrator\Runbook Server\PolicyModule.exe

For more information about adding firewall rules, see Add or Edit a Firewall Rule.

RunbookService fails to start after computer reboot

When you reboot your runbook server, the RunbookService attempts to connect to the orchestration database. If the database is not available, the RunbookService fails. The event log message is **This computer was unable to communicate with the computer providing the server.** Typically, this can occur when the SQL server and the runbook server are installed on the same computer.

To solve this problem. you can manually start the RunbookService, or configure the RunbookService to make multiple attempts during startup to connect to database before failing.

Cannot restart runbook service if you uninstall with an account without administrator permissions

If you attempt to uninstall Orchestrator while logged in with an account that is a member of OrchestratorSystemGroup but is not an administrator, uninstall removes all accounts from OrchestratorSystemGroup. If you stop the runbook service and attempt to restart the service, the services fails because the user account does not have the correct permissions to retrieve the orchestration database connection. An account that is an administrator or a member of the OrchestratorSystemGroup is required to retrieve the orchestration database connection.

To solve this problem, an administrator can add the user back to OrchestratorSystemGroup.

HTTP errors when starting the Orchestration console

If you get HTTP errors when starting the Orchestration console, try the following mitigations:

- HTTP error 404.3 Not Found: Ensure that both .NET Framework 4.5 and Windows
 Communication Foundation (WCF) HTTP Activation are installed. You can find the
 instructions in Single-Computer Requirements for System Center 2012 R2 Orchestrator.
- HTTP error 404.17 Not Found: Ensure that Windows Communication Foundation (WCF)
 HTTP Activation is installed. You can find the instructions in <u>Single-Computer Requirements</u>
 for System Center 2012 R2 Orchestrator.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Deploying System Center 2012 Orchestrator
- Deployment Overview
- Plan Your Orchestrator Deployment
- Install Orchestrator
- Perform Post-Installation Tasks

Administering System Center 2012 - Orchestrator

System Center 2012 - Orchestrator is a workflow management solution for the datacenter that you can use to automate the creation, monitoring, and deployment of resources in your environment. This guide describes tasks that an administrator might have to perform to maintain a production Orchestrator environment. It assumes that you have a fully deployed Orchestrator environment as described in Deploying System Center 2012 - Orchestrator.

In This Guide

- Accessibility for People with Disabilities
 Describes accessibility options in Orchestrator.
- How to Change the Orchestrator Database

Describes how to change the location of the orchestration database.

- How to Change the Orchestrator Users Group
 - Describes how to remove and add members to the Orchestrator Users Group.
- How to Configure Orchestrator Database Connections
 - Describes how to use DBSetup to configure connections to an Orchestrator database.
- How to Configure Runbook Servers to Optimize Performance of .NET Activities
 Describes how to configure a runbook server to perform optimally with activities that require Microsoft .NET libraries.
- How to Configure Runbook Throttling

Describes how to use the Runbook Server Runbook Throttling tool to change the maximum number of runbooks that can run on a runbook server.

How to View Orchestrator Data by Using PowerPivot

Describes how to view Orchestrator data by using Microsoft Excel PowerPivot.

Orchestrator Logs

Describes the types of logs available in Orchestrator and how to configure them.

Runbook Server Properties

Describes the runbook server properties.

How to Back up Orchestrator

Describes how to back up an Orchestrator database.

How to Use the Integration Toolkit to Extend Orchestrator Capabilities

Provides a link to the MSDN library content that helps you create new integration packs for Orchestrator.

Other Resources for this Product

- TechNet Library main page for Orchestrator Release Candidate
- Getting Started with System Center 2012 Orchestrator
- Deploying System Center 2012 Orchestrator
- Integration Packs for System Center 2012 Orchestrator Release Candidate
- Using Runbooks in System Center 2012 Orchestrator
- Using the Orchestration Console in System Center 2012 Orchestrator
- Runbook Activity Reference for System Center 2012 Orchestrator

Accessibility for People with Disabilities

Microsoft is committed to making its products and services easier for everyone to use. The following topics provide information about the features, products, and services that make Microsoft System Center 2012 - Orchestrator more accessible for people with disabilities.

In This Section

Accessibility Features of Orchestrator

Describes the accessibility features of Orchestrator.

Accessibility Features of Orchestrator Help

Describes the accessibility features of the Orchestrator Help.

Accessibility Products and Services from Microsoft

Describes the accessibility products and services that are available from Microsoft.

See Also

Orchestrator Release Candidate

Accessibility Features of Orchestrator

In addition to accessibility features and tools in Windows, the following feature makes Orchestrator more accessible for people with disabilities.

For information about using the Orchestrator Command Line Install tool, see Install with the Orchestrator Command Line Install Tool in Deploying System Center 2012 - Orchestrator.



The information in this section may apply only to users who license Microsoft products in the United States. If you obtained this product outside of the United States, you can use the subsidiary information card that came with your software package or visit the Microsoft Accessibility website for a list of Microsoft support services telephone numbers and addresses. You can contact your subsidiary to find out whether the type of products and services described in this section are available in your area. Information about accessibility is available in other languages, including Japanese and French.

Using Keyboard Shortcuts in the Orchestration Console

To do this	Use this keyboard shortcut
Navigate to a control.	Tab or any arrow key
Activate the edit mode.	Enter
Exit the edit mode.	Esc
Select a control.	Enter
Browse runbooks.	Ctrl+Alt+B
Browse runbook servers.	Ctrl+Alt+V
Browse events.	Ctrl+Alt+E
View summary.	Ctrl+Alt+S
View runbooks.	Ctrl+Alt+R
View jobs.	Ctrl+Alt+J
View instances.	Ctrl+Alt+I

To do this	Use this keyboard shortcut
View privacy policy.	Ctrl+Alt+P
Provide feedback.	Ctrl+Alt+F
Open help.	Ctrl+Alt+H
Open query menu.	Ctrl+Alt+Q
Start runbook.	Ctrl+Alt+Y
Stop runbook.	Ctrl+Alt+Z

See Also

Accessibility for People with Disabilities

Accessibility Features of Orchestrator Help

Orchestrator Help includes features that make it accessible to a wider range of users, including those who have limited dexterity, low vision, or other disabilities. In addition, Orchestrator Help is available on the Web at Accessibility for People with Disabilities.

Keyboard Shortcuts for Using the Help Window

By using the following keyboard shortcuts in Help, you can quickly accomplish many common tasks.

To do this	Use this keyboard shortcut
Display the Help window.	F1
Switch the cursor between the Help topic pane and the navigation pane (tabs such as Contents , Search , and Index).	F6
Change between tabs (for example, Contents , Search , and Index) while in the navigation pane.	ALT + Underlined letter of the tab
Select the next hidden text or hyperlink.	TAB
Select the previous hidden text or hyperlink.	SHIFT+TAB
Perform the action for the selected Show All,	ENTER

To do this	Use this keyboard shortcut
Hide All, hidden text, or hyperlink.	
Display the Options menu to access any Help toolbar command.	ALT+O
Hide or show the pane containing the Contents , Search , and Index tabs.	ALT+O, and then press T
Display the previously viewed topic.	ALT+O, and then press B
Display the next topic in a previously displayed sequence of topics.	ALT+O, and then press F
Return to the specified home page.	ALT+O, and then press H
Stop the Help window from opening a Help topic (useful if you want to stop a Web page from downloading).	ALT+O, and then press S
Open the Internet Options dialog box for Windows Internet Explorer, where you can change accessibility settings.	ALT+O, and then press I
Refresh the topic (useful if you have linked to a Web page).	ALT+O, and then press R
Print all topics in a book or a selected topic only.	ALT+O, and then press P
Close the Help window.	ALT+F4

Procedures

To change the appearance of a Help topic

- 1. To customize the colors, font styles, and font sizes used in Help, open the Help window.
- 2. Click Options, and then click Internet Options.
- 3. On the General tab, click Accessibility. Select Ignore colors specified on Web pages, Ignore font styles specified on Web pages, and Ignore font sizes specified on Web pages. You also can choose to use the settings specified in your own style sheet.
- 4. To change the colors used in Help, see "To change the color of the background or text in Help". To change the font, see "To change the font in Help."

To change the color of the background or text in Help

1. Open the Help window.

- 2. Click **Options**, and then click **Internet Options**.
- On the General tab, click Accessibility. Then, select Ignore colors specified on Web pages. You also can choose to use the settings specified in your own style sheet.
- 4. To customize the colors used in Help, on the General tab, click Colors. Clear the Use Windows Colors check box, and then select the font and background colors that you want to use.



Note

If you change the background color of the Help topics in the Help window, the change also affects the background color when you view a Web page in Windows Internet Explorer.

To change the font in Help

- 1. Open the Help window.
- 2. Click **Options**, and then click **Internet Options**.
- 3. On the General tab, click Accessibility. To use the same settings as those used in your instance of Windows Internet Explorer, select Ignore font styles specified on Web pages and Ignore font sizes specified on Web pages. You also can choose to use the settings specified in your own style sheet.
- 4. To customize the font style used in Help, on the General tab, click Fonts, and then click the font style you want.



Note

If you change the font of the Help topics in the Help window, the change also affects the font when you view a Web page in Internet Explorer.

Accessibility Products and Services from Microsoft

Microsoft is committed to making its products and services easier for everyone to use. The following sections provide information about the features, products, and services that make Microsoft® Windows® more accessible for people with disabilities:

- Accessibility Features of Windows
- Documentation in Alternative Formats
- Customer Service for People with Hearing Impairments
- For More Information



Note

The information in this section may apply only to users who license Microsoft products in the United States. If you obtained this product outside of the United States, you can use

the subsidiary information card that came with your software package or visit the <u>Microsoft Accessibility website</u> for a list of Microsoft support services telephone numbers and addresses. You can contact your subsidiary to find out whether the type of products and services described in this section are available in your area. Information about accessibility is available in other languages, including Japanese and French.

Accessibility Features of Windows

The Windows operating system has many built-in accessibility features that are useful for individuals who have difficulty typing or using a mouse, are blind or have low vision, or who are deaf or hard-of-hearing. The features are installed during Setup. For more information about these features, see Help in Windows and the Microsoft Accessibility website.

Free Step-by-Step Tutorials

Microsoft offers a series of step-by-step tutorials that provide detailed procedures for adjusting the accessibility options and settings on your computer. This information is presented in a side-by-side format so that you can learn how to use the mouse, the keyboard, or a combination of both. To find step-by-step tutorials for Microsoft products, see the Microsoft Accessibility website.

Assistive Technology Products for Windows

A wide variety of assistive technology products are available to make computers easier to use for people with disabilities. You can search a catalog of assistive technology products that run on Windows at the Microsoft Accessibility website. If you use assistive technology, be sure to contact your assistive technology vendor before you upgrade your software or hardware to check for possible compatibility issues.

Documentation in Alternative Formats

If you have difficulty reading or handling printed materials, you can obtain the documentation for many Microsoft products in more accessible formats. You can view an index of accessible product documentation on the Microsoft Accessibility website. In addition, you can obtain additional Microsoft publications from Learning Ally (formerly Recording for the Blind & Dyslexic, Inc.). Learning Ally distributes these documents to registered, eligible members of their distribution service. For information about the availability of Microsoft product documentation and books from Microsoft Press, contact:

Learning Ally (formerly Recording for the Blind & Dyslexic, Inc.)

20 Roszel Road Princeton, NJ 08540

Telephone number from within the United States: (800) 221-4792
Telephone number from outside the United States and Canada: (609) 452-0606

Fax: (609) 987-8116 Learning Ally website

Web addresses can change, so you might be unable to connect to the website or sites mentioned here

Customer Service for People with Hearing Impairments

If you are deaf or hard-of-hearing, complete access to Microsoft product and customer services is available through a text telephone (TTY/TDD) service:

- For customer service, contact Microsoft Sales Information Center at (800) 892-5234 between 6:30 AM and 5:30 PM Pacific Time, Monday through Friday, excluding holidays.
- For technical assistance in the United States, contact Microsoft Product Support Services at (800) 892-5234 between 6:00 AM and 6:00 PM Pacific Time, Monday through Friday, excluding holidays. In Canada, dial (905) 568-9641 between 8:00 AM and 8:00 PM Eastern Time, Monday through Friday, excluding holidays.

Microsoft Support Services are subject to the prices, terms, and conditions in place at the time the service is used.

For More Information

For more information about how accessible technology for computers helps to improve the lives of people with disabilities, see the <u>Microsoft Accessibility website</u>.

How to Change the Orchestrator Database

You might have to change the location of the Orchestrator database after installation, because you might want to separate the management server and database server, move the database to a larger server or a cluster, or just reconfigure the orchestration database based on required changes in your environment. You can use standard Microsoft SQL Server methods to move the existing database to another server, but then you must configure the Orchestrator features to connect to the new server. You must perform this configuration for the management server, the web service supporting the Orchestration console, and each runbook server as described in the following procedures.

Management server and runbook servers

You can use the Database Configuration utility to change the connection settings that the management server and runbook servers installed in your environment. The settings for these

servers are stored in an encrypted file called **Settings.dat**. If you change your orchestration database settings, such as the port, user account access, or computer name, you must manually uninstall and reinstall all runbook servers, and then re-run the Database Configuration utility on the management server and all runbook servers.

To change the database settings for the management server and runbook servers

- 1. On the management server, click **Start**, point to **All Programs**, click **Microsoft System Center 2012**, click **Orchestrator**, and then click **Data Store Configuration**.
- 2. In the **Server** box, enter the name of the server that is hosting the database by using the format **<server>\<instance>,<port>**. You can click the ellipsis **(...)** button to select the computer. You do not have to include the instance if the Orchestrator database is installed on the default instance. You do not have to include the port if SQL Server is usually installed on the default port 1433.
 - If the Orchestrator database is installed on an instance called MyInstance on a computer named MySQLServer that is configured on port 12345, enter MySQLServer\MyInstance,12345.

If the Orchestrator database is installed on an instance called MyInstance on a computer named MySQLServer that is configured on port 1433, enter MySQLServer\MyInstance.

If the orchestration database is installed on the default instance on a computer named MySQLServer that is configured on port 1433, enter **MySQLServer**.

- 3. Select the authentication method to use to connect to the SQL Server:
 - **Windows Authentication** Connect to the SQL Server by using Windows Authentication.
 - SQL Server Authentication Connect to the SQL Server by using a SQL Server user
 account. Type the User Name and Password of the SQL Server user account. This
 account must have rights to create, write, and own a database and create, update,
 and delete rows in the database.
- Click Next.
- 5. In the Data Store pane, click Use an existing database.
- 6. In the **Name** list, select the database.
- 7. Click Finish.

Web Service

The web service supporting the Orchestration console does not use the **Settings.dat** file. To change the database settings for the web service, you must modify the Web.config file on the Internet Information Services (IIS) server. You can use **IIS Manager** to modify the file, but you must first decrypt it by running the aspnet_regiis.exe executable file.

To change the database settings for the Orchestrator web service

1. Log on with administrative credentials to the computer with the Orchestration console

installed.

- 2. Open a Command Prompt window with administrator credentials.
- 3. Run the following command to decrypt the Web.config file:
 - C:\Windows\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis.exe -pdf "connectionStrings" "C:\Program Files (x86)\Microsoft System Center 2012\Orchestrator\Web Service\Orchestrator2012"
- 4. To start the IIS Manager, click **Start**, point to **Administrative Tools**, and then click **Internet Information Services (IIS) Manager**.
- Expand the Sites node, and then click Microsoft System Center 2012 Orchestrator Web Service.
- 6. In the Features View, double-click Connection Strings.
- 7. In the Connections String pane, double-click OrchestratorContext.
- 8. In the **Custom** box, scroll down to the portion of the string that includes the server name (Data Source) and database name (Initial Catalog). Modify these values as required.
- 9. Click **OK** to close the dialog box.
- 10. Close IIS Manager.
- 11. Run the following command to encrypt the Web.config file:

C:\Windows\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis.exe -pef "connectionStrings" "C:\Program Files (x86)\Microsoft System Center 2012\Orchestrator\Web Service\Orchestrator2012"

See Also

Administering System Center 2012 - Orchestrator

How to Change the Orchestrator Users Group

You might want to change the Orchestrator users group after installation because of changes in your environment. For example, you might want to use a local group during installation, and then change it to a domain account later.

PermissionsConfig tool

You can change the Orchestrator Users group by using the PermissionsConfig tool, which is located on the management server in **<InstallDir>\Management Server**. The syntax of this tool is as follows:

PermissionsConfig—OrchestratorUsersGroup *GroupName***—OrchestratorUser** *UserName***-remote**

Note that the PermissionsConfig tool does not send results to standard output. To view the results of the command, check the %errorlevel% in the Orchestrator log file that is located at C:\Users\SCXSVC\AppData\Local\SCO\LOGS. The results are 1 for failure, 0 for success.

You can get an explanation of the parameters for the PermissionsConfig tool by typing the following command:

PermissionsConfig -help

The following table explains the parameters.

Parameter	Details
OrchestratorUsersGroup	The name of the group to use for Orchestrator permissions.
OrchestratorUser	If this parameter is specified with a user name, the user is granted immediate access to Orchestrator whether a member of the specified group or not. This is to prevent the requirement for the user to log off and on if the group has just been created.
Remote	Indicates that the Runbook Designer can be run from a computer other than the management server.

For example, to change the Orchestrator users group to a group that is named Orchestrator Users in a domain that is named Contoso, use the following command:

PermissionsConfig -OrchestratorUsersGroup "Contoso\Orchestrator Users" -remote



Important

You must run the PermissionsConfig tool at a command prompt with administrative credentials because it modifies group memberships. To do this, right-click the Command Prompt icon to select Run as Administrator.

See Also

Orchestrator Security Planning

How to Configure Orchestrator Database Connections

DBSetup allows you to secure an unsecured database. The common scenario is connecting to a restored backup.

This utility provides two functions:

- 1. DBSetup allows you to change the database name or credentials that are used by the management server or runbook servers to connect to the database.
- 2. DBSetup allows you to connect to a rebuilt database.

When connecting to a rebuilt database:

- This procedure can only be performed against the same database server used during the installation of the management server.
- You must have database permissions to create the database.

In contrast, DBconfig only creates a new database; it does not configure the security for the database. DBConfig configures the database schema in the database and creates the contents of **settings.dat**, which contains the connection details for the management server and runbook servers. For more information on running DBConfig, see How to Change the Orchestrator Database.

To configure Orchestrator database connections

• Run the DBsetup binary from the **Start** menu or from the **Program Files** folder.

To create a new database on a new database server

- 1. Run the System Center 2012 Orchestrator Setup Wizard and install a new management server.
- 2. On the **Configure the database server** page in the setup wizard, point to the new database server.
- 3. After you add a new DB server to your deployment, you must also run permissionsconfig, and then export and import the service master key to the new database server.

See Also

How to Change the Orchestrator Database

How to Configure Runbook Servers to Optimize Performance of .NET Activities

If a runbook contains an activity that references the .NET libraries, the first reference to the .NET libraries takes additional time to initialize. This delay can be as much as 30 seconds. All remaining activities that reference the .NET libraries run immediately. This delay can also occur when a runbook is started on a computer without Internet access, because then Windows cannot verify the Microsoft Authenticode signature for the .NET libraries, and this causes a delay during the initialization of the activity.

The solution to removing the delay is to deactivate **generatePublisherEvidence** in **PolicyModule.exe** or to create a profile for the service account.

To deactivate generatePublisherEvidence in policymodule.exe.config

- On the runbook server where runbooks that contain an activity referencing the .NET libraries run, locate the file C:\Program Files (x86)\Microsoft System Center 2012\Orchestrator\Runbook Server\policymodule.exe.config.
- 2. Add the following code to policymodule.exe.config:

To create a profile for the service account

 On the runbook server where runbooks run that contain an activity referencing the .NET libraries, log on to the computer that is using the service account credentials. A profile is created on first logon.

See Also

Administering System Center 2012 - Orchestrator

How to Configure Runbook Throttling

By default, each runbook server is configured to simultaneously run a maximum of 50 runbooks. You can change this number by using the Runbook Server Runbook Throttling tool. In most cases, you can keep this default setting, but you should consider the resource requirements of the runbooks on a particular server when considering whether to change it. If the server has a number of runbooks with high resource requirements, you might run fewer runbooks simultaneously on the runbook server. If they are simple runbooks with minimal requirements, you might consider increasing the number of simultaneously run runbooks.

To configure the maximum number of runbooks that a runbook server processes

- Navigate to the folder where by default the Runbook Server Runbook Throttling tool is stored: <System Drive>:\Program Files (x86)\Microsoft System Center 2012\Orchestrator\Management Server.
- 2. Type one of the following commands:
 - To apply the change to one runbook server:

aspt <RunbookServerName> <MaximumRunningRunbooks>.

For example, to set the maximum number of runbooks that RunbookServer1 runs to 40:

aspt RunbookServer1 40

To apply the change to all runbook servers:

aspt * <MaximumRunningRunbooks>.

For example, to set the maximum number of runbooks that all runbook servers run to 40:

aspt * 40

3. Restart the Orchestrator Runbook Service.

See Also

Administering System Center 2012 - Orchestrator

How to View Orchestrator Data by Using PowerPivot

You can use Microsoft PowerPivot for Microsoft Excel to create reports for System Center 2012 - Orchestrator. You configure PowerPivot to use the Orchestrator web service as a data feed, filter the Source Tables for the data you want to use, and import the tables into the PowerPivot worksheet. PowerPivot lets you create relationships between tables, and manipulate the data to fit your requirements. By using the PivotTable feature in PowerPivot, you can generate a report that uses any of the data contained within the PowerPivot workbook.

Connect the Orchestrator web service to PowerPivot for Excel

You must install PowerPivot for Excel to enable the product.

PowerPivot for Excel requires Excel 2010 (64-bit or 32-bit).

To install PowerPivot

1. Follow the instructions found at Install PowerPivot for Excel.

Use PowerPivot to configure a connection to Orchestrator web service. Orchestrator uses the Open Data Protocol (OData), which PowerPivot can consume.

Note

The OData provider in PowerPivot does not support the data contained in the Runbook Diagram box. Attempts to add a Runbook Diagram table will fail.

To create a connection to an Orchestrator feed

- 1. Open Excel.
- Click the PowerPivot tab above the ribbon.
- 3. Click PowerPivot Window on the ribbon. A PowerPivot for Excel book opens.
- 4. Click From Data Feeds on the ribbon. A Table Import Wizard opens.
- Enter the Orchestrator web service URL in the **Data Feed URL** box. The web service URL is on port 81 of the Orchestrator SQL Server. For example, http://orchestrator:81/Orchestrator2012/Orchestrator.svc.
- 6. Click Test Connection.
- 7. If the test connection is successful, click **OK** and proceed to the next step.

If the test connection fails, do the following:

- a. Click OK.
- b. Click **Advanced**. The **Advanced** dialog box opens.
- c. In the Security section, change Integrated Security to Basic.
- d. Change Persist Security Info to True.
- e. Enter your User ID and Password in the appropriate boxes.
- f. Click Test Connection.
- g. Click OK and click OK.
- 8. Click Next.
- 9. Select the check boxes of the table or tables that you want to import.
- 10. To filter columns, select a table, click **Preview & Filter**, clear any boxes to exclude, and then click **OK**.
- 11. Click Finish. The data is imported.
- 12. Click Close.

Create a Summary of Runbook Results

The following procedure describes the steps to create a pivot table containing a list of all runbooks and the count of results, grouped by the runbook server that ran the runbook instance.

Note

For this example, the orchestration database must contain results from at least one runbook for PowerPivot to import a table.

To create a connection to the data feed

- 1. Open Excel.
- 2. Click the **PowerPivot** tab above the ribbon.
- 3. Click PowerPivot Window on the ribbon. A PowerPivot for Excel book opens.
- 4. Click From Data Feeds on the ribbon. A Table Import wizard opens.
- 5. Enter the Orchestrator web service URL in the **Data Feed URL** box.
- 6. Click Next.
- 7. Select the check boxes of the **Runbooks**, **Runbookinstances**, and **RunbookServers** tables.
- 8. Click **Finish**. The data is imported.
- Click Close.

To create relationships in PowerPivot

- 1. In the **PowerPivot for Excel** window, select the **RunbookInstance** tab.
- 2. Right-click the header of the **Runbookld** column to select **Create Relationship**.
- 3. In the **Related Lookup Table** list, select **Runbooks**, and in the **Related Lookup Column** list, select **Id**, and then click **Create**.
- 4. Right-click the header of the RunbookServerId column to select Create Relationship.
- 5. In the **Related Lookup Table** list, select **RunbookServers**, and in the **Related Lookup Column** list, select **Id**, and then click **Create**.

For additional information about PowerPivot relationships, see <u>Introduction to PowerPivot relationships</u>.

To create a pivot table

- 1. In the **PowerPivot for Excel** window, click **PivotTable** on the ribbon, and select **PivotTable**.
- 2. In the Create PivotTable dialog box, select New Worksheet, and then click OK.
- In the PowerPivot Field List, under RunbookServers, click and drag Name to the Row Labels box.
- 4. In the **PowerPivot Field List**, under **Runbooks**, click and drag **Name** to the **Row Labels** box.
- 5. In the **PowerPivot Field List**, under **RunbookInstances**, click and drag **Status** to the **Column Labels** box.
- 6. In the **PowerPivot Field List**, under **RunbookInstances**, click and drag **RunbookId** to the **Sum Values** box.

7. Right-click **Runbookld** to select **Summarize by**, and then click **Count**.

You can now modify the default labels and format your table for presentation.

For more information about the workflow of a runbook and an explanation of runbook jobs and runbook instances, see <u>Orchestrator Architecture</u> in the <u>Getting Started with System Center 2012 - Orchestrator</u>.

For more information about PowerPivot for Excel, see Introducing PowerPivot for Excel.

See Also

Administering System Center 2012 - Orchestrator

Orchestrator Logs

This section provides information on common troubleshooting issues and the available tools that can assist in identifying root problems.

Log files

In Orchestrator, different logs are available that provide information about Orchestrator runbooks and servers. The following table lists the available types of log files, with links to the appropriate sections that describe the logs in more detail.

Log File Topic or Section	Contents	Where data is stored	Where data is viewed
Real time log section in Runbook Logs.	Live information about a running runbook instance	Orchestration database	Log tab in Runbook Designer
Historic Log section in Runbook Logs.	Historical information about instances of a runbook	Orchestration database	Log History tab in Runbook Designer
Runbook Audit History section in Runbook Logs.	Audit information about changes to a runbook	Orchestration database	Audit History tab in Runbook Designer
Activity Events	Status information about Orchestrator management server, runbook servers, and database	Orchestration database	Events tab in Runbook Designer

Log File Topic or Section	Contents	Where data is stored	Where data is viewed
Audit Trail	Interaction of a runbook with external tools and systems	Log files	Open files in text editor
Trace Logs	Troubleshooting information about the Orchestrator environment	Log files	Open files in text editor

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Using Runbooks in System Center 2012 Orchestrator
- Designing a Runbook
- Building a Runbook
- How to Test a Runbook

Runbook logs

Every runbook generates a log when it is running. The Runbook Designer enables you to view both real-time log information and historic logs.

Real-Time Log

You can view the real-time log of a runbook on the **Log** tab in the **Log** pane of the Runbook Designer. The log has an entry for each running instance of the runbook and the time that it was started. If you expand this entry, you can view the activity that is currently running. Double-click the activity to view its details. If you have configured logging for the workbook to include Published Data, this information is included in the activity's details.

Historic Log

You can view the historic log of a runbook on the **Log History** tab in the **Log** pane of the Runbook Designer. This log has an entry for each completed instance of the runbook with its start and end times and completion status. If you double-click an entry, the **Runbook Details** dialog box opens that includes each activity in the runbook and its completion status. Double-click each activity to view its details. If you have configured logging for the workbook to include Published Data, this information is included in the activity's details.

Runbook Log Options

By default, the Runbook logs do not include Published Data for each activity. For standard activities, refer to <u>Standard Activities</u> for a list of the Published Data items available for each. For integration packs, refer to the specific integration pack guide. You can change the logging properties for an individual runbook to include this information in the logs. You can include both or either of the following two kinds of Published Data:

- Activity-specific Published Data is Published Data that is specific to a particular activity.
 Typically, this option should be enabled only for debugging purposes.
- Common Published Data is a set of data items that are common to all activities. These are as follows:
 - Activity Name
 - Activity Type
 - Activity ID
 - Activity End Time Year, Month, Day, Weekday, Hours, Minutes, Seconds
 - Activity Duration
 - Previous Activity
 - Previous Activity Name
 - Time Published Data



By default, logging options are disabled. Be aware that adding these items to the log increases the size of the data store.

Runbook Audit History

The **Runbook Audit History** tracks the changes to each activity in a runbook. This includes the user that made the change, the date and time that the change occurred, the activity that was changed, and the current and previous values of any changed properties. The audit history is maintained as long as the runbook exists and cannot be cleared.

You can view the **Runbook Audit History** on the **Audit History** tab of the Runbook Designer. The entries for the currently selected runbook will be displayed. Double-click an entry to open the **Details** dialog box where you can view a list of all activities that were part of the change. Select one of the entries to view the activity that was performed in addition to the old and new values of any properties that were changed.

Purging Runbook Logs

The **Runbook Designer** provides a **Log Purge** feature. Orchestrator log data can be purged on demand, or on a scheduled basis. A good practice is to schedule a log purge regularly during off-peak hours to maintain the database.



When logs are purged, the data is deleted from Orchestration database and cannot be recovered.

If you let the number of log entries grow indiscriminately, it can result in excessive storage requirements for the Orchestration database. To maintain the health and performance of the database, you should regularly purge old log entries. You can purge either them on an automatic schedule or manually.

Automatically Purging Runbook Logs

You can schedule the runbook logs to be purged on a regular basis by selecting frequency options and how many entries to retain. By default, runbook logs are automatically purged with the following settings.

Setting	Default value
Frequency	One time per day
Time	2:00 AM
Retain	Keep last 5000 entries

The default log purge job purges all but the last 500 log entries per runbook. This means that if an Orchestrator deployment had 20 runbooks, the default purge would keep the last 500 log entries for each runbook, for a maximum of 10,000 log entries for the database for all runbooks.

You can continue to use these defaults or change the settings by using the following process in the Runbook Designer.

To purge runbooks on demand or set the purge frequency

- In the Runbook Designer, right-click Management server to select Log Purge.
 If you want to purge all the runbooks now, click Purge Now.
 If you want to purge a specific runbook now, click the Log History tab to select the runbook to purge.
- 2. Ensure that the **Schedule Log Purge** option is selected.
- 3. Select the number of days between running the purge process and a time to run the process.
- 4. If you do not want to delete all entries, set the **Log Purge Options** to specify the log entries that you want to keep.
- 5. Click Finish.

Manually Purging the Runbook Logs

Audit History logs are a special case because these logs canno bte purged with the **Log Purge** feature. The audit history feature tracks the changes made to a runbook in the **Runbook Designer** and cannot be deleted. The only way to delete Audit History logs is to delete the runbook associated with them. However, data volumes for the Audit History logs for a given runbook are generally small, even in large deployments storage will not require more than the minimum system requirements.

There are two options to manually purge the runbook logs as shown in the following two procedures. They are performed in the Runbook Designer.

To manually purge the runbook logs for all runbooks

- 1. Right-click Management server to select Log Purge.
- 2. Set the **Log Purge Options** to specify the log entries that you want to keep.
- 3. Click Purge Now.

To manually purge all runbook log entries for a single runbook

- 1. In the **Connections** pane, select the **Runbooks** folder.
- 2. In the workspace pane, click the tab of the runbook.
- 3. In the Log History pane, click the Log History tab.
- 4. Click the **Recycle Bin** icon at the top of the pane.
- 5. When prompted whether you are sure that you want to purge the logs, click Yes.

To delete a specific set of runbook log entries for a single runbook

- 1. In the **Connections** pane, select the **Runbooks** folder.
- 2. In the workspace pane, click the tab of the runbook.
- 3. In the **Log History** pane, click the **Log History** tab.
- 4. Select a single log entry or use the key combination Ctrl+Shift to select multiple entries.
- 5. Click the red **Delete** icon at the top of the pane.
- 6. When prompted if you are sure that you want to delete the selected entries, click Yes.

Removing Orphaned Log Entries

Orphaned log files can be left in the orchestration database if the Runbook Server Service is stopped suddenly while runbook instances are running. If that happens, the runbook instance status does not get updated correctly with its ending time. These entries are removed when the log entries are purged by using one of the purging methods described above. If you want to remove orphaned entries before the next time that the entries are purged, you can manually run the **ClearOrphanedRunbookInstances** stored procedure to perform this function.

The following process describes how to run this stored procedure by using the Microsoft SQL Server Management Studio installed on the orchestration database server. You can perform this process from any computer with this tool installed that has access to the orchestration database.

To remove orphaned log entries

- 1. On the server with the orchestration database, click **Start**, point to **All Programs**, click Microsoft SQL Server 2008, and then click SQL Server Management Studio.
- 2. In the Connect to Server dialog box, in the Server name list, select (local), and then click Connect.
- 3. In the Object Explorer, expand Databases, and then click Orchestrator.



Note

If you used a different name for the Orchestrator database, select that name.

- 4. Expand Programmability, and then click Stored Procedures.
- 5. Right-click

Microsoft.SystemCenter.Orchestrator.Runtime.Internal.ClearOrphanedRunbookIns tances to select Execute Stored Procedure.

- 6. In the Execute Procedure dialog box, click OK.
- 7. On the toolbar, click **Execute**.
- 8. When you receive a message at the bottom of the Logging pane that the query has completed, close SQL Server Management Studio.

See Also

Orchestrator Logs

Activity Events

Each activity in an Orchestrator runbook has the ability to send an event whenever it fails to run or is taking too long to run. These events are presented on the **Events** tab of the Runbook Designer or can be configured to be delivered to a receiver as an SNMP trap. Runbook activity events are only sent for those activities that you specifically configure to do so.

To configure an activity to send events

- 1. Open the runbook in the **Runbook Designer**.
- 2. Double-click the activity to view its properties.
- 3. Click the Run Behavior tab.
- 4. Type a number of seconds to send an event when the activity runs too long.
- 5. Check the Report if the activity fails to run box to send an event when the activity fails.
- 6. Click **Finish** to save the activity.

Receiving Events from SNMP

In addition to viewing the events on the **Events** tab in the Runbook Designer, you can send them to an SNMP trap destination. This lets you monitor the health of the Orchestrator environment by using other tools designed to provide proactive alerting. The only requirement for such a tool is that it can receive SNMP traps. You can use the **Orchestrator Event Delivery Configuration Utility** to add and configure SNMP trap destinations for Runbook events.

Add an SNMP Trap Destination

To add an SNMP trap destination, run the **oedc** command one time for each destination that you want to add by using the following syntax:

oedc /snmp /add /ip <Targeted IP Address> /port <Targeted Port> /version <version>
/community <community>

For example, use the following procedure to send traps by using SNMP version 1 to an SNMP receiver at IP address 10.1.1.10 on port 162 and a community called public.

To add an SNMP trap destination

- 1. Open a command prompt with administrative credentials.
- 2. Navigate to *System Drive*:\Program Files (x86)\Microsoft System Center 2012\Orchestrator\Management Server.
- Type the following command: oedc /snmp /add /ip 10.1.1.10 /port 162 /version SNMP1 /community public
- Restart the Orchestrator Runbook Service and the Orchestrator Runbook Server Monitor service.

Remove All SNMP Trap Destinations

You cannot remove individual SNMP trap destinations. Instead, you must remove all destinations, and then add back any that you want. To remove all SNMP trap destination, run the **oedc** command with the following syntax:

oedc /snmp /clear

To remove all SNMP trap destinations

- 1. Open a command prompt with administrative credentials.
- Navigate to System Drive:\Program Files (x86)\Microsoft System Center 2012\Orchestrator\Management Server.
- 3. Type the following command: oedc /snmp /clear
- 4. Restart the Orchestrator Runbook Service and the Orchestrator Runbook Server Monitor service.

Receiving SNMP Traps

After you have configured an SNMP trap destination for Orchestrator event notifications, you can receive them by using any tool that reads SNMP traps, or you can use the **Monitor SNMP Trap** activity in a runbook to read the information. The content of SNMP traps is limited to the first 1000 characters if the content exceeds that length.

The variable bindings are listed in the following table.

Trap Enterprise ID	1.3.6.1.4.1.4217.100.100
Generic ID	(6)
Specific ID	(1)
Orchestrator Event Information IDs	Orchestrator Event Type – 1
	Orchestrator Event Summary – 2
	Orchestrator Event Details – 3
Example	Orchestrator Event Type – 1.3.6.1.4.1.4217.100.100.1
	Orchestrator Event Summary – 1.3.6.1.4.1.4217.100.100.2
	Orchestrator Event Details – 1.3.6.1.4.1.4217.100.100.3

See Also

Orchestrator Logs

Audit Trail

The *Audit Trail* is a collection of text log files that contain information about the interaction of a runbook with external tools and systems. By using the Audit Trail, you can report on configuration and change compliance of processes and identify changes made to a non-Microsoft system for audit purposes or to remediate a change that causes service interruption.

Depending on how many runbooks you invoke and how many activities those runbooks contain, the Audit Trail can consume a large amount of disk space on the computer that runs the management server and runbook server. If you enable auditing, you should implement an

archiving procedure to move the files generated by the Audit Trail to another computer on a regular basis.

Activating and Deactivating the Audit Trail

By default, the Audit Trail is not activated when you install Orchestrator. You can use the following procedure to activate it.

To activate or deactivate the Audit Trail

- 1. Open a command prompt with administrative credentials.
- 2. Navigate to *System Drive*:\Program Files (x86)\Microsoft System Center 2012\Orchestrator\Management Server.
- 3. To activate the Audit Trail, type **atlc /enable**, and to deactivate the Audit Trail, type **atlc /disable**.

Audit Trail Files

Audit Trail files are stored in comma-separated value file (.csv) format. The following table shows the details.

Log Type	File name	Conten ts	Computer	Location
Runboo k Publishe r	Computer Name_ RunbookPublisher_Timest amp.csv	Da te an d tim e tha t the run bo ok wa s sta rte d	Managem ent Server Runbook Server	System Drive:\ProgramData\Microsoft System Center 2012\Orchestrator\Audit\Managem entService System Drive:\ProgramData\Microsoft System Center 2012\Orchestrator\Audit\RunbookS ervice

Log	File name	Conten	Computer	Location
Туре		ts		
		Us		
		er		
		na		
		me		
		an		
		d		
		do		
		ma in		
		tha		
		t		
		sta		
		rte		
		d		
		the		
		run		
		bo		
		ok		
		•		
		Na		
		me		
		of the		
		the co		
		mp		
		ute		
		r		
		wh		
		ere		
		the		
		run		
		bo		
		ok		
		ran		
Activity	Computer Name_	•	Runbook	System
Runtime	ObjectRuntimeInfo_Timest		Server	Drive:\ProgramData\Microsoft
Informati	amp.csv	Da		System Center
on	,	te		2012\Orchestrator\Audit\PolicyMod
		an		ule
		d		
		tim		
		е		

Type ts tha t act ivit y ran Na me of run bo ook ser ver tha t ran the act ivit y • ID of the job pro ce ss tha t ran the act ivit	Log	File name	Conten	Computer	Location
tha t act ivit y y ran Na me of run bo ok ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit u ran the act ivit u ran the iob pro ce ss tha t ran the act ivit					
t act ivit y y ran Na me of run bo ok ser ver tha t ran the act ivit y y ID of the job pro ce ss tha t ran the act ivit was the individual to the act ivit wit the act ivit			tha		
ivit y ran Na me of run bo ok ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit					
V ran Na me of run bo ok ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit ran the act ivit ran the act ivit					
Na me of run bo ok ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit			1		
Na me of run bo ok ser ver tha t ran the job pro ce ss tha t ran the act ivit					
Na me of run bo ok ser ver tha t ran the job pro ce ss tha t ran the act ivit			rar	1	
me of run bo ok ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit			•		
me of run bo ok ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit					
of run bo ok ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit					
run bo ok ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit				•	
bo ok ser ver that t ran the act job proce ss that t ran the in the interpretation of th					
ok ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit					
ser ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit					
ver tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit					
tha t ran the act ivit y ID of the job pro ce ss tha t ran the act ivit					
ran the act ivit y ID of the job pro ce ss tha t ran the act ivit					
the act ivit y ID of the job pro ce ss tha t ran the act ivit			t		
act ivit y ID of the job pro ce ss tha t ran the act ivit					
ivit y ID of the job pro ce ss tha t ran the act ivit					
y ID of the job pro ce ss tha t ran the act ivit					
• ID of the job pro ce ss tha t ran the act ivit					
of the job pro ce ss tha t ran the act ivit					
the job pro ce ss tha t ran the act ivit					
job pro ce ss tha t ran the act ivit					
pro ce ss tha t ran the act ivit					
ce ss tha t ran the act ivit					
ss tha t ran the act ivit					
tha t ran the act ivit					
ran the act ivit				ı	
the act ivit			t		
the act ivit			rar		
ivit					
			act		
y			У		
•			•		
Ob ince			Ob		
jec			Jec		
t XM			χ ι .	1	

Log Type	File name	Conten	Computer	Location
		L		
		со		
		de		
		tha		
		t		
		act		
		ivit		
		у		
		rec		
		eiv		
		ed		
		as		
		inp		
		ut		
		dat		
		а		

When a file reaches 200 megabytes (MB) in size, a new file is created. The time stamp is included in the file name to ensure that each file name is unique. Passwords and other encrypted text fields are represented by five asterisks (*****) in the Audit Trail files.



Note

The ProgramData folder holding the audit files is often a hidden system folder.

See Also

Orchestrator Logs

Trace Logs

Orchestrator can create trace log messages on the management server to help you identify problems in the environment. By default, trace log messages are only written when there is an exception in the Orchestrator Management Service, but you can increase this level of logging by modifying a registry setting. Trace log files are available in directories for each Orchestrator feature under C:\ProgramData\Microsoft System Center 2012\Orchestrator. You can change these log locations by changing the registry values for the different features.



Note

The **C:\ProgramData** directory is often a hidden system folder.

Modifying Trace Log Settings

Trace log settings are configured with registry values under the key HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\SystemCenter2012\Orches trator\TraceLogger. Each of the Orchestrator services has a set of registry values as shown in the following table.

Feature	Registry key
Audit Trail tool	Atlc.exe
Data Store Configuration Utility	DBSetup.exe
Management Service	ManagementService.exe
Permissions Configuration tool	PermissionsConfig.exe
Runbooks and activities	PolicyModule.exe
Runbook Designer	RunbookDesigner.exe
Runbook Server Monitor	RunbookServerMonitorService.exe
Runbook Service	RunbookService.exe
Runbook Tester	Runbook Tester.exe

The following table shows the values that are under each key. Modify these values as appropriate to change the logging details for that feature.

Registry value name	Default value	Description
LogFolder	C:\ProgramData\Microsoft System Center 2012\Orchestrator\ManagementService.exe\Logs	Location where trace logs are stored.
LogLevel	1	Level of detail of information that is logged. Possible values are in the table below.
LogPrefix	ManagementService.exe	Prefix of the log file name. This prefix is followed by the date and time when the log file is created.
NewLogEvery	3600	Number of seconds

Registry value name	Default value	Description
		until a new log file is created.

The following table provides the possible values that you can use for the log level.

Log level	Description
1	Exception detail only. This is the default setting.
3	Exception detail with warnings and errors.
7	Full logging.

See Also

Orchestrator Logs

Runbook Server Properties

The properties for a runbook server include an optional description and the account information to use for the Runbook Service. You can modify the description but can only view the service credentials. For more information about the service account and how to change it, see the Orchestrator Security Planning topic in Deploying System Center 2012 - Orchestrator.

To view runbook server properties

- 1. In the **Connections** pane, select the Runbook Servers folder. In the right pane, right-click the runbook server to select **Properties**.
- 2. If you want to add or change the **Description** box, type a description for this runbook server, and then click **Finish**.

See Also

Administering System Center 2012 - Orchestrator

How to Back up Orchestrator

A complete backup of an Orchestrator environment consists of the following:

- Backup of the Orchestrator database.
- File backup of the Orchestrator management server.
- File backup of each Runbook server and Orchestrator web server.

System Center 2012 - Orchestrator supports Volume Shadow copy Service (VSS) for backup and restore with System Center 2012 – Data Protection Manager (DPM). VSS is a framework that allows volume backups to be performed while an application continues to run.

Registering Orchestrator with VSS

The **SCOExpressWriter** command-line utility registers an Orchestrator database as a component associated with the Orchestrator management server. This association instructs DPM to back up the Orchestrator database when it performs a backup of the management server. Without this registration DPM must perform an individual backup of each component.

You must run **SCOExpressWriter** on the management server being registered, and you must be logged on with a user account that is a member of the local Administrators group.

The usage of this command-line utility is as follows:

```
SCOExpressWriter {/register | /unregister}
```

To register the Orchestrator database used by the local management server, run the following command:

SCOExpressWriter /register

Orchestrator Servers

Orchestrator management server, Runbook servers, and web servers do not persist any data. Runbooks and their settings are stored entirely in the Orchestrator database and accessed by these servers as required. Management servers and Runbook servers have a settings dat file that includes configuration details to connect to the Orchestrator database. Orchestrator web servers have a web.config file with this same information. These files are backed up with standard file backups which are supported by DPM.

Orchestrator Database

The Orchestrator database is a standard SQL Server database that is supported by DPM. You should make sure to backup the service master key and store it in a secure off-site location. For more information see BACKUP SERVICE MASTER KEY (Transact-SQL).

Migrate Orchestrator Between Environments

This topic describes how to automatically move Orchestrator between environments. This could be useful when you want to just move to a new SQL Server 2008 R2 or if you want to move some or all of the components of Orchestrator.

The following processes and scripts enable you to easily move between environments. They are based on a full migration of all System Center 2012 - Orchestrator components to a new SQL Server 2008 R2 with a restored Orchestrator database.

The following steps are required to enable an automatic migration of Orchestrator to a new environment:

- Backup SQL Server service master key in environment A
- Backup the Orchestrator database in environment A
- Deploy SQL Server 2008 R2 in environment B
- Restore SQL Server service master key in environment B
- Restore Orchestrator database in environment B
- Deploy Orchestrator components in environment B



See http://go.microsoft.com/fwlink/?LinkId=246817 for information on using the Sqlcmd utility.

Back up SQL Server service master key in environment A

Back up the SQL Server 2008 R2 service master key using the following procedure as described in http://go.microsoft.com/fwlink/?LinkID=243093. This is a one-time operation.

Create a batch script with the following command:

```
Sqlcmd -Q"BACKUP SERVICE MASTER KEY TO FILE = 'C:\BACKUP\MASTER KEY.BAK' ENCRYPTION BY
PASSWORD = 'password'"
```

Where 'password' is the password that will be used to protect the service master key in the file that is created. If the password is lost, the service master key cannot be recovered from the file.

Back up the Orchestrator database in environment A

Back up the entire Orchestrator database. You can perform the backup when the system is running; however it is best to perform the backup when all runbook authors have checked in any pending changes to their runbooks. Pending changes are cached on the Runbook Designer and are not backed up with a database backup.

To back up the Orchestrator database

- 1. In SQL Server Management, right-click the Orchestrator database, click Tasks, and then click Back up.
- 2. Configure the backup settings as required in your organization.

- 3. Click Script, and then click Script Action to New Query Window.
- 4. Click **Execute** to test the backup script.
- 5. Create a batch file with this script. Your batch file will be similar to the following:

```
Sqlcmd -Q "BACKUP DATABASE Orchestrator TO DISK=N'C:\BACKUP\OrchestratorDB.bak'"
```

Deploy SQL Server 2008 R2 in environment B

Deploy SQL Server to environment B. See http://go.microsoft.com/fwlink/?LinkID=246815 for information about creating a Sysprep image of SQL Server 2008 R2.

Restore the SQL Server service master key in environment B

Restore the Microsoft SQL Sevver 2008 R2 service master key by using the procedure described at http://go.microsoft.com/fwlink/?LinkID=243093. This will enable decryption of Orchestrator data on the new SQL server.

Create a batch script with the following command:

```
Sqlcmd -Q "RESTORE SERVICE MASTER KEY FROM FILE = 'C:\BACKUP\MASTER_KEY.BAK' DECRYPTION
BY PASSWORD = 'password';"
```

Restore the Orchestrator database in environment B

Use the following steps to create a batch script to run on the new SQL Server computer to restore the Orchestrator database.

To create the batch file

- In SQL Server Management, right-click the Orchestrator database, click Tasks, and then click Restore.
- 2. Configure the restore settings as required in your organization.
- Click Script, and then click Script Action to New Query Window.
- 4. Click **Execute** to test the restore script.
- 5. Create a batch file with this script. Your batch file will be similar to the following:

```
Sqlcmd -Q"RESTORE DATABASE [Orchestrator] FROM DISK =
N'C:\BACKUP\OrchestratorDB.bak'WITH FILE = 1, NOUNLOAD,
STATS = 10"
```

Deploy Orchestrator components in environment B

Deploy Orchestrator components (management server, Web features, runbook servers, and Runbook Designers) using the silent install commands of Orchestrator setup. See <u>Install with the Orchestrator Command Line Install Tool</u> for more information on deploying Orchestrator through the command line.

The following example installs all of Orchestrator on a computer with SQL Server 2008 R2 and .NET Framework 4:

%systemdrive%\sco\setup\setup.exe /Silent /ServiceUserName:%computername%\administrator
/ServicePassword:password /Components:All /DbServer:%computername% /DbPort:1433
/DbNameNew:OrchestratorSysPrep /WebConsolePort:82 /WebServicePort:81 /OrchestratorRemote
/UseMicrosoftUpdate:1 /SendCEIPReports:1 /EnableErrorReporting:always

Sample migration scripts and commands

Backup SQL Server master service key sample

```
Sqlcmd -Q "BACKUP SERVICE MASTER KEY TO FILE ='C:\BACKUP\MASTER_KEY.BAK' ENCRYPTION BY PASSWORD = 'password'"
```

Backup Orchestrator database sample

Sqlcmd -Q "BACKUP DATABASE Orchestrator TO DISK=N'C:\BACKUP\OrchestratorDB.bak'"

Restore SQL Server master service key sample

```
Sqlcmd -Q "RESTORE SERVICE MASTER KEY FROM FILE =
'c:\temp backups\keys\service master key' DECRYPTION BY PASSWORD = 'password'"
```

Restore Orchestrator database sample

```
Sqlcmd -Q "RESTORE DATABASE [Orchestrator] FROM DISK =
N'C:\BACKUP\OrchestratorDB.bak'WITH FILE = 1, NOUNLOAD, STATS = 10"
```

Install Orchestrator from batch file sample

%systemdrive%\sco\setup\setup.exe /Silent /ServiceUserName:%computername%\administrator /ServicePassword:password /Components:All /DbServer:%computername% /DbPort:1433 /DbNameNew:OrchestratorSysPrep /WebConsolePort:82 /WebServicePort:81 /OrchestratorRemote /UseMicrosoftUpdate:1 /SendCEIPReports:1 /EnableErrorReporting:always

Best Practices Analyzer

The Microsoft System Center 2012 - Orchestrator Best Practices Analyzer is a tool that looks at the configuration data in an Orchestrator deployment and identifies settings that may cause issues within your environment. It performs the following functions:

- Gathers information about an Orchestrator deployment
- Determines if the configurations are set according to the Microsoft recommended best practices
- Reports on collected configurations, indicating settings that differ from recommendations
- Indicates potential problems in the deployment

Installation

The Orchestrator BPA must be installed on the Orchestrator Management server. It depends on having the Microsoft Baseline Configuration Analyzer 2.1

(http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=16475) installed on the Orchestrator Management Server prior to installation.

Install process:

- 1. Install Microsoft Baseline Configuration Analyzer 2.1
- Run Microsoft.SystemCenter.2012.Orchestrator.BestPracticesAnalyzer.msi

Uninstall process:

1. Remove Microsoft System Center 2012 Orchestrator – Best Practices Analyzer (BPA) from the Programs and Features panel within Windows Control Panel.

Usage

1. Run the Microsoft Baseline Configuration Analyzer 2.1 from the Start menu in windows.



Note

To learn how to run the Microsoft Baseline Configuration Analyzer scans from PowerShell cmdlets, please see the MBCA 2.1 help file.

- Select System Center 2012 Orchestrator BPA from the product dropdown within the MBCA 2.1 user interface.
- 3. Click Start Scan.
- Review the results and recommendations.

Orchestrator rules

System Center 2012 - Orchestrator configuration checks:

Orchestrator Component	Rule	Description
Management Server	Check log purge is set to the default value	This rule checks that the default log purging for Runbooks is set to the default values for running every one day and to keep the last five hundred entries. Please refer to http://go.microsoft.com/fwlink/?LinkID=239613 for more information about setting the purging policy for Runbook logs.
Management Server	Check the refresh interval for permissions in the Orchestration console	This rule checks that the default refresh interval for generating the cache that allows access to runbooks from the Orchestration Console is set to six hundred seconds. Please refer to http://go.microsoft.com/fwlink/?LinkID=239946 for more information on how the refresh cache is set up.
Management Server	Check logging on ManagementService.exe	This rule checks that the default trace logging on the ManagementService.exe is set to the default of 1. This can impact performance if a different value is configured. Please refer to http://go.microsoft.com/fwlink/?LinkID=239530 for more information on configuring trace logs.
Management Server	Check logging on PermissionsConfig.exe	This rule checks that the default trace logging on the PermissionsConfig.exe is set to the default of 1. This can impact performance if a different value is configured. Please refer to http://go.microsoft.com/fwlink/?LinkID=239530 for more information on configuring trace logs.
Management Server	Check logging on Runbooks	This rule checks if logging is enabled on Runbooks. This can impact performance if logging is enabled on frequently used Runbooks. Please refer to http://go.microsoft.com/fwlink/?LinkID=239614 for more information on enabling logging on Runbooks.
Management Server	Check memory on Management Servers	This rule checks that the memory allocated to the Management Server is greater than 2048MB. If the Management Server has less than 2048MB you should monitor its

Orchestrator Component	Rule	Description
		performance to ensure it is meeting the expected goals in the environment. Please refer to http://go.microsoft.com/fwlink/?LinkID=242545 for information on the Orchestrator Management Pack.
Runbook Server	Check that maximum concurrent runbooks is set to the default value of 50.	This rule checks that the maximum number of concurrent runbooks that are configured to run on a Runbook Server is set to fifty. This can impact performance if a different value is configured. Please refer to http://go.microsoft.com/fwlink/?LinkID=239560 for more information on configuring Runbook throttling.
Runbook Server	Check logging for PolicyModule.exe	This rule checks that the default trace logging on the PolicyModule.exe is set to the default of 1. This can impact performance if a different value is configured. Please refer to http://go.microsoft.com/fwlink/?LinkID=239530 for more information on configuring trace logs.
Runbook Server	Check logging for RunbookService.exe	This rule checks that the default trace logging on the RunbookService.exe is set to the default of 1. This can impact performance if a different value is configured. Please refer to http://go.microsoft.com/fwlink/?LinkID=239530 for more information on configuring trace logs.
Runbook Server	Check memory on Runbook Servers	This rule checks that the memory allocated to the Runbook Servers is greater than 2048MB. If the Runbook Server has less than 2048MB you should monitor its performance to ensure it is meeting the expected goals in the environment.
Web components	Check memory on the server hosting the Orchestration Console	This rule checks that the memory allocated to the Orchestration Console server is greater than 2048MB. If the server has less than 2048MB you should monitor its performance to ensure it is meeting the expected goals in the environment. Please refer to

Orchestrator Component	Rule	Description
		http://go.microsoft.com/fwlink/?LinkID=242545 for information on the Orchestrator Management Pack.
Web components	Check users group managing runbook access	This rule checks that the windows group that is used to manage access to runbooks is configured as a domain group if the web components are not installed on the Management Server. The group must be a domain group in order for users to have access through the web service and Orchestration console when the web components are installed on a separate server than the Management Server. Please refer to http://go.microsoft.com/fwlink/?LinkID=239561 for more information on configuring the Orchestrator users group.
Designer	Check memory on the computer hosting the Orchestrator Designer	This rule checks that the memory allocated to the Orchestrator Designer is greater than 2048MB. If the computer has less than 2048MB you should monitor its performance to ensure it is meeting the expected goals in the environment. Please refer to http://go.microsoft.com/fwlink/?LinkID=242545 for information on the Orchestrator Management Pack.

How to Benchmark

Orchestrator runbook activities can be thought of as having two distinct types of code: platform code and domain code. The term *domain code* is used to identify code within a runbook activity that is typically not associated with the Orchestrator platform itself (with notable exceptions, such as **Invoke Runbook**, **Junction**, and others). For example, the **Invoke Web Service** standard activity would contain Orchestrator platform code (the "plumbing" of the activity) as well as domain code unique to invoking a SOAP-based web service. The platform code will be very similar for most activities, since it is built on a common framework. However, there will potentially be great variation in domain code for different activities.

Data Logging

Another aspect of runbook performance is data logging. For the purpose of understanding performance consider two logging configurations: Default logging and Common Published Data logging. Default logging results in approximately 524 bytes of data being written to the Orchestrator database each time an activity is run. Logging of common published data writes approximately 6,082 bytes of data (12 times the default logging level). There is a notable difference in performance between these logging levels.

Consider the scenario where the same runbook activity is run twice, once with data logging at the default level and once with logging of common published data enabled. The domain code should take the same amount of time to complete. However, the platform code will take longer to run with common published data logging enabled. Essentially, the platform code has to support logging 12 times more data with common published data enabled than it did when running at the default logging level.

The Standard Activity **Compare Values** can been used to create benchmarks of an Orchestrator environment.

To create a runbook that can be used to benchmark your Orchestrator environment

- 1. Create a new runbook.
- 2. Add a **Compare Values** activity from the Standard Activity palette. Double-click the activity to configure it.
- 3. Click the **General** tab and configure this activity to compare strings (the default value).
- 4. Click the **Details** tab, type the value **STRING** in the **Test** box and select **is empty**.
- 5. Click **Finish** to save the updates to the activity.
- 6. Right-click the activity and select Looping.
- 7. Select the **Enable** checkbox and enter the number **0** (zero) for **Delay between attempts**.
- 8. Click the **Exit** tab.
- Change the default exit condition. Click Compare Values, check the Show Common Published Data checkbox, and select Loop: Number of attempts. Click OK to save this change.
- Select value from the updated exit condition and type the number 10000 (ten-thousand).
 Click OK to save this change.
- 11. Click **Finish** to save these updates.
- 12. Click **Check In** to save the changes to the Orchestrator database.

This simple one-activity runbook will run a **Compare Values** activity 10,000 times. **Compare Values** is a very simple activity whose domain code is quite minimal. This runbook can be invoked under a variety of circumstances to characterize the overall performance of a given Orchestrator runtime environment.

This runbook can be used to experiment with different configurations of Orchestrator. For example, supposed you wanted to determine the performance of four Runbook Servers deployed to different data centers.

Data Center	Logging Configuration	Platform Code Run Time (seconds)	ms/Activity	Scale Factor
Location 1	Default logging	819	82	1.0 (reference)
Location 1	Logging common published data	2012	201	2.5
Location 2	Default logging	1229	123	1.5
Location 2	Logging common published data	3686	369	4.5
Location 3	Default logging	2457	426	3.0
Location 3	Logging common published data	4422	442	5.4
Location 4	Default logging	1474	147	1.8
Location 4	Logging common published data	2654	265	3.2

Notice the significant decrease in platform performance caused by logging of common published data. The worst scenario appears to be logging of common published data at Location 2. On the surface, this appears to be a clear and relevant conclusion.

However, it should be noted that these figures reflect the overhead of the platform code, not the domain code. Domain code runtimes can be significantly longer. For example, the **Create VM from Template** activity in the Virtual Machine Manager Integration Pack may run for several minutes as the VM is created. Expanding on the previous example, consider the platform code costs on a runbook activity that takes 1 minute to run (1 minute = 60,000 milliseconds) regardless of location.

Data Center	Logging Configuration	Platform Code Run Time (seconds)	% Domain Code	% Platform Code
Location 1	Default logging	819	98.6%	1.4%
Location 1	Logging common published data	2012	96.7%	3.3%
Location 2	Default logging	1229	98.0%	2.0%
Location 2	Logging common published data	3686	93.9%	6.1%

Data Center	Logging Configuration	Platform Code Run Time (seconds)	% Domain Code	% Platform Code
Location 3	Default logging	2457	95.9%	4.1%
Location 3	Logging common published data	4422	92.6%	7.4%
Location 4	Default logging	1474	97.5%	2.5%
Location 4	Logging common published data	2654	95.6%	4.4%

A clearer picture begins to emerge from the data. The scenario where logging of common published data is enabled at Location 2 continues to be the worst performer. However, the platform code and logging only accounts for 6% of the total runtime. While this is a significant figure, the best-case scenario is 1.4%. Essentially, the time spent in the domain code in the example far outweighs the time spent running platform code. To put this in perspective, if you were able to completely eliminate the platform code costs, you would only see runbook performance improvements in the range of 1.4 to 7.4%.

Of course most real-world scenarios will be different. Activity behavior may change depending on what the domain code is told to do. For example, a **Clone VM from Template** activity may take one minute to clone a VM from Server Template A, but take 5 minutes to clone a VM from Server Template B. Also, Runbook Servers may reside on different networks with different performance characteristics which can potentially impact both domain code performance as well as Orchestrator data logging performance.

To summarize:

- Make careful decisions about when to log published data.
- Carefully consider the impact of logging common published data. Remember that the number
 of times activities run determines the volume of logged data. A runbook with a small number
 of activities run many times can result in more data logging than a larger runbook run a small
 number of times.
- Do not enable logging of activity specific published data in production environments.
- Develop an understanding of how much time your runbooks spend running domain code compared to running platform code.
- Estimate platform code costs using the techniques outlined in this document. Use as a reference in considering where to make improvements in runbook performance.
- Use the techniques outlined in this document to gain a deeper understanding of the relative performance of your different runtime environments. Identify opportunities for improvement by making normalized comparisons of your measurements.

How to Recover a Database

The Orchestrator database can be backed up and restored using most standard MS SQL Server database backup/restore mechanisms. This includes Microsoft SQL Server Backup, DPM SQL Server backup, and others. Orchestrator provides a VSS Writer that will discover the database server that is associated with the Management Server and back up the database when the Management Server is backed up.

However, there are a few key considerations when restoring.

Orchestrator Cryptography

System Center 2012 Orchestrator provides a set of services for encryption and decryption of runbook properties and published data. These services are based on Microsoft SQL Server 2008 R2 cell-level encryption. The Orchestrator database has a database encryption key that is created during its installation. This key is generated using a random passphrase. When a full database backup is performed, the key is backed up with the database. Likewise, the key is restored when the database is restored.

However, the encryption services also depend on the MS SQL Server Service Master Key. The service master key should be backed up and stored in a secure, off-site location. Creating this backup should be one of the first administrative actions performed on the server. The procedure for doing this is documented for Microsoft SQL Server 2012 (http://go.microsoft.com/fwlink/?LinkId=249148).

The database key is essentially paired with the service master key on the database server targeted by the installer. If either the database key or the service master key is lost, encrypted data stored in the data is likewise lost. This would include the license key, either entered by the user or an automatically created trial license.

To perform a backup

Back up the Microsoft SQL Server service master key using the procedure for <u>backing up</u> the service master key for <u>Microsoft SQL Server 2012</u>. This is a one-time operation. Note "password" is the password that will be used to protect the service master key in the file that is created. If the password is lost, the service master key cannot be recovered from the file.

```
BACKUP SERVICE MASTER KEY TO FILE = 'path_to_file'

ENCRYPTION BY PASSWORD = 'password'
```

2. Back up the entire Orchestrator database. The backup may be performed when the system is running, but it is best to perform the backup when all runbook authors have checked in any pending changes to their runbooks. Pending changes are cached on the Runbook Designer and are not backed up with a database backup.

To restore the database

- 1. If you are restoring to the same database server from which the backup was taken, and the service master key has not changed, simply restore the backup.
- 2. If you are restoring to a different database server with a different service master key, or you are restoring to the same database from which the backup was taken but the service master key has changed, the service master key must be restored to match the one used during the database backup. Use the procedure for restoring the service master key for Microsoft SQL Server 2012 (http://go.microsoft.com/fwlink/?LinkId=249149).

```
BACKUP SERVICE MASTER KEY TO FILE =
'c:\temp backups\keys\service master key'
         ENCRYPTION BY PASSWORD = '3dH85Hhk003GHk2597jheij4'
```



Note

If there are multiple databases using this service master key for encryption on your Microsoft SQL Server, all of these databases could be affected by this change. Consulting with your DBA before performing this administrative task is strongly recommended.

- Restore the database from the backup.
- 4. On the Orchestrator Management Server, run the Data Store Configuration utility from the Start menu.
- 5. Provide the connection details to connect to the new database. Note: Do not use "localhost" or ".". Explicitly specify the database server name and database name.
- 6. Restart the Management Service.
- 7. Run the Data Store Configuration utility on each Runbook Server. This utility is not located in the Start menu on Runbook Servers. It can be found in <SC2012OrchestratorInstallDir>\Microsoft System Center 2012\Orchestrator\Runbook Server. Note: For Runbook Servers installed on the same server as the Management Server one doesn't need to run the Data Store Configuration utility a second time. Running it once will update the configuration for both the Management Server and Runbook Server at the same time.
- 8. Restart the Runbook Server(s).
- 9. Follow the Web Components Recovery Process to update the Web Components to connect to the new database.

Database Sizing and Performance

Database sizing is the key to understanding the performance of System Center 2012 -Orchestrator. The runbook servers, management server, and web components all depend on the Orchestrator database for their operations. Problems with Orchestrator deployments can arise from an incomplete understanding of the types of data in the database and how to manage them.

Because the Runbook Designer communicates with the Orchestrator database (through the management server), poor database performance will impede that communication.

The Orchestrator operator experience is based on two components: The **Orchestration Console** and the Web Service. The **Orchestration Console** is a Silverlight-based application that depends on the Web Service for its connection to the Orchestrator database. The Web Service is an IIS application that connects to the database. Consequently, the Web Service and **Orchestration Console** are both dependent on the performance of the Orchestrator database.

Additionally, while the **Orchestration Console** is dependent on the Web Service, it also has logic unique to its function as a user interface and its own performance characteristics.

Key Concepts

Configuration Data and Log Data

At a high level the Orchestrator database contains two kinds of data:

- Configuration Data
 - The Orchestrator infrastructure contains configuration data. This data is not a concern in the context of database growth because the storage requirements for this type of data are small.
- Log Data

Orchestrator creates different types of log data, all of which can be viewed and managed in the **Runbook Designer**. The storage requirements for this data can vary in size and be large.

The following table lists the types of log data that can be stored in the Orchestrator database. Orchestrator also stores data in separate log files (outside of the database) for audit trails and tracing. For more information about all the types of log data, see <u>Orchestrator Logs</u>.

Type of Log Data Location in Runbook Designer		Managed by Log Purge?
Runbook logs	Log and Log History tabs	Yes
Activity (Platform) events	Events tab	Yes
Audit history	Audit History tab	No

Platform Code and Domain Code

Orchestrator runbook activities contains two distinct types of code:

- Platform Code
 - This is common code shared by most activities, and is used to run common tasks performed by Orchestrator activities. Platform code generates Common Published Data.
- Domain Code

Runs a variety of tasks that are specific for the actions for each activity, that are typically not associated with the Orchestrator platform itself. Potentially, there can be great variation between platform code and domain code.

The logging data generated for a given activity can contain data elements that are single or multi-valued. Every activity produces a single record of single-value data. Domain code can produce multiple records of multi-value data and is therefore responsible for determining what the activity does with the common published data it has received from prior activities.

Essentially, Orchestrator runbooks are designed to pass data between discrete elements of domain code. Also, domain code can optionally generate Activity-specific Published Data.

All runbooks have core similarity in that they run activities that consist of domain code and platform code, they loop workflows and they branch. Branching is when a runbook calls other runbooks to do a specific task. When a runbook is first invoked it consists of a single thread. When this thread encounters a runbook activity whose links require a branch, additional threads are created, one for each branch. Each thread takes as input the common published data from the activity that created the branch. This data is correlated back to the prior activities in the runbook to update the common published data that the activities subscribe to.

Domain code potentially affects database performance more than multi-threading generated by branching. This is because domain code can potentially generate large amounts of activityspecific published data.

Logging Options

The **Logging** tab on the **Properties** for a runbook allows you to optionally store logging entries. The term default logging refers to having neither of the two published data options selected, which amounts to 524 bytes generated for each activity. The logging options provide for two categories of common published data:

Common Published Data

The set of data items common to all activities. For a list, see the Runbook Log Options section in Runbook logs.

This logging option generates 6082 bytes for each activity.

Activity-specific Published Data

The set of data that is specific to the activity that is optionally created by domain code. This logging option generates 6082 bytes in addition to the bytes logged by specific activities.



∏ Tip

This option is selected primarily for debugging purposes. Leave unchecked to limit logging growth.

Setting logging options can significantly affect performance and increase database growth. Consider the scenario where the same runbook activity is run twice, first with data logging at the default level (no published data options selected) and then set with common published data selected. The domain code should take the same amount of time to complete. However, the

platform code will take longer to run because it has to support 12 times the amount of common published data logging than it does with just default logging.

Purging Logs

The default options specified for the **Log Purge** feature in the **Runbook Designer** is configured to provide the best user experience for an out-of-the-box Orchestrator deployment. Changing these values can change the performance characteristics of the environment, and should be implemented gradually and high-watermarked, so that the impact of the change can be evaluated.

For more information on automatic and manual purging of logs, see the <u>Purging Runbook Logs</u> section of Runbook logs.

Creating Performance Benchmarks

To create a simple runbook to test logging growth you can use the Standard Activity **Compare Values** to create benchmarks of an Orchestrator environment.

The following procedure creates a simple runbook that runs a **Compare Values** activity 10,000 times. **Compare Values** is a very simple activity whose domain code is quite minimal. This runbook can be invoked under a variety of circumstances to characterize the overall performance of a given Orchestrator runtime environment.

To create a runbook that can be used to benchmark your Orchestrator environment

- 1. Create a new runbook.
- 2. Add a **Compare Values** activity from the Standard Activity palette. Double-click the activity to configure it.
- 3. Click the **General** tab and configure this activity to compare strings (the default value).
- 4. Click the **Details** tab, type the value **STRING** in the **Test** box and select **is empty**.
- 5. Click **Finish** to save the updates to the activity.
- 6. Right-click the activity and select **Looping**.
- 7. Select the **Enable** checkbox and enter the number **0** (zero) for **Delay between attempts**.
- 8. Click the Exit tab.
- Change the default exit condition. Click Compare Values, check the Show Common Published Data checkbox, and select Loop: Number of attempts. Click OK to save this change.
- Select value from the updated exit condition and type the number 10000 (ten-thousand).
 Click OK to save this change.
- 11. Click Finish to save these updates.
- 12. Click **Check In** to save the changes to the Orchestrator database.

This runbook can be used to experiment with different configurations of Orchestrator. For example, you can create the benchmark runbooks to determine the performance of four Runbook Servers deployed to different data centers.

Data Center	Logging Configuration	Platform Code Run Time (milliseconds)	Milliseconds per Activity	Scale Factor
Location 1	Default logging	819	82	1.0 (reference)
Location 1	Logging common published data	2012	201	2.5
Location 2	Default logging	1229	123	1.5
Location 2	Logging common published data	3686	369	4.5
Location 3	Default logging	2457	426	3.0
Location 3	Logging common published data	4422	442	5.4
Location 4	Default logging	1474	147	1.8
Location 4	Logging common published data	2654	265	3.2

Notice the significant decrease in platform performance caused by logging of common published data. The worst scenario appears to be logging of common published data at Location 2. On the surface, this appears to be a clear and relevant conclusion.

However, it should be noted that these figures reflect the overhead of the platform code, not the domain code. Domain code runtimes can be significantly longer. For example, the **Create VM from Template** activity in the Virtual Machine Manager Integration Pack may run for several minutes as the VM is created. Expanding on the previous example, consider the platform code costs on a runbook activity that takes 1 minute to run (1 minute = 60,000 milliseconds) regardless of location.

Data Center	Logging Configuration	Platform Code Run Time (milliseconds)	% Domain Code	% Platform Code
Location 1	Default logging	819	98.6%	1.4%
Location 1	Logging common published data	2012	96.7%	3.3%
Location 2	Default logging	1229	98.0%	2.0%
Location 2	Logging common	3686	93.9%	6.1%

Data Center	Logging Configuration	Platform Code Run Time (milliseconds)	% Domain Code	% Platform Code
	published data			
Location 3	Default logging	2457	95.9%	4.1%
Location 3	Logging common published data	4422	92.6%	7.4%
Location 4	Default logging	1474	97.5%	2.5%
Location 4	Logging common published data	2654	95.6%	4.4%

A clearer picture begins to emerge from the data. The scenario where logging of common published data is enabled at Location 2 continues to be the worst performer. However, the platform code and logging only accounts for 6% of the total runtime. While this is a significant figure, the best-case scenario is 1.4%. Essentially, the time spent in the domain code in the example far outweighs the time spent running platform code. To put this in perspective, if you were able to completely eliminate the platform code costs, you would only see runbook performance improvements in the range of 1.4 to 7.4%.

Of course most real-world scenarios will be different. Activity behavior may change depending on what the domain code is told to do. For example, a **Clone VM from Template** activity may take one minute to clone a VM from Server Template A, but take 5 minutes to clone a VM from Server Template B. Also, Runbook Servers may reside on different networks with different performance characteristics which can potentially impact both domain code performance as well as Orchestrator data logging performance.

Determining Database Growth

Your database administrator for the Orchestrator database can use the following guidelines for determining database file growth strategy:

- In general, the database files will not increase in size with each invocation of a runbook. The
 files will grow when the data contained within them reaches a certain high watermark
 configured by your database administrator, at which time the file will generally be expanded.
- Each time a runbook activity runs it should be counted individually, which should be considered when looping features can cause a single activity to run multiple times.
- To determine the storage needed for each invocation of the runbook, multiply the number of activities in the runbook by the number of bytes added to the database according the selected logging level. These values are as follows:
 - 524 bytes
 Default logging configuration.
 - 6082 bytes
 Common published data logging level.

- 6082 bytes + data logged by activity
 Activity-specific published data logging level.
- By default, Orchestrator purges all but the most recent 500 logs for each runbook nightly at 2:00 am. To determine the storage required for each invocation of the runbook, multiply the storage needed for each invocation of the runbook by 500. If you change the Log Purge setting, multiply each invocation by the estimated number of invocations per day, week, or month as needed.

The following table shows growth and performance estimates for the logging level configurations.

Logging Level	DB Growth Factor	Performance Factor	Recommended for Production
Default	1	1	Yes
Common published data	11.6x	2.5x	Limited use with planning
Activity-specific published data	Greater than 11.6x	Greater than 2.5x	No

Examples

Example 1

The following table describes the database sizing considerations for a deployment of Orchestrator.

Runbook Name	Number of Activities	Logging Level	Invocations per Day
Runbook 1	50	Default	100
Runbook 2	25	Default	50
Runbook 3	12	Common published data	24
Runbook 4	8	Common published data	500

Using the database sizing described above, you can estimate the storage requirements for the runbooks.

Runbook Name	Bytes per Invocation	Storage in MB Default Log Purge (500 invocations)	Invocations per Month	Storage in MB One Month (Not Default Log Purge)	% of DB storage after 30 Days
Runbook 1	26,200	12.5	3,000	74.5	9%
Runbook 2	13,100	6.2	1,500	18.7	2%
Runbook 3	72,984	34.8	720	50.1	6%
Runbook 4	48,656	23.2	15,000	696.0	83%
		Total: 76.7 MB		Total: 839.8 MB	

This example clearly illustrates the importance of making sound decisions for data logging. Runbook 4 contains only eight activities, but when configured at the Common Published Data Logging level, it consumes most of storage in the database because of the high frequency of invocation. Based on these results you may prefer to reduce the logging level of Runbook 4 to the Default logging configuration.

Example 2

The following table describes the database sizing considerations for another deployment of Orchestrator.

Runbook Name	Number of Activities	Logging Level	Invocations per Day
Runbook 1	50	Default	100
Runbook 2	25	Default	50
Runbook 3	12	Common published data	24
Runbook 4	8	Default	500

Recalculating the storage figures for the updated configuration produces significantly different results.

Runbook Name	Bytes per Invocation	Storage in MB Default Log Purge (500 invocations)	Invocations per Month	Storage in MB One Month (Not Default Log Purge)	% of DB storage after 30 Days
Runbook 1	26,200	12.5	3,000	74.5	37%
Runbook 2	13,100	6.2	1,500	18.7	9%
Runbook 3	72,984	34.8	720	50.1	25%
Runbook 4	4,192	2.0	15,000	60.0	29%
		Total: 55.5 MB		Total: 203.8 MB	

While there is very little change in the default logging configuration (500 log entries per runbook), the 30-day storage requirements have changed greatly. Clearly the storage cost of using Common Published Data logging for Runbook 4 should be carefully considered since this change results in a 76% reduction in database storage requirements for 30 days of data.

Summary

Use the following guidelines to manage database sizing and performance:

- Enable logging of Common Published Data only if needed.
- Remember that the number of times activities run determines the volume of logged data. A small runbook with a few of activities run several times can result in more data logging than a larger runbook run a fewer number of times.
- Do not enable logging of Activity-specific Published Data in production environments, and should only be used for debugging purposes.
- Develop an understanding of how much time your runbooks spend running domain code compared to running platform code.
- Estimate platform code costs using the techniques outlined in this document. Use as a reference in considering where to make improvements in runbook performance.
- Identify opportunities for improvement by making normalized comparisons of your measurements.

See Also

Orchestrator Logs
Runbook logs
Orchestrator Architecture

How to Recover Web Components

The Web Service database reference does not get modified by the Database Configuration Utility (only the installer performs this task). You will need to manually modify it.

Web Components Recovery Process

To do this, you will need to complete the following actions:

To modify the Web Service database reference

- 1. Open a Command Prompt using Run as administrator.
- 2. Execute the following command (assuming the default installation path):

```
C:\Windows\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis.e
xe -pdf "connectionStrings" "C:\Program Files (x86)\Microsoft
System Center 2012\Orchestrator\Web Service\Orchestrator2012"
```

- 3. Open IIS Manager and navigate to the Orchestrator2012 virtual application.
- 4. Open up Connection Strings and then modify OrchestratorContext. Locate the segment that starts with "provider=System.Data.SqlClient;provider connection string" and then modify the Data Source and Initial Catalog attributes according to your new SQL Server and Database Catalog name respectively, then click OK.
- 5. If you want to re-encrypt the connection strings, you can execute the following command at the command prompt:

```
C:\Windows\Microsoft.NET\Framework\v4.0.30319\aspnet_regis.e
xe -pef "connectionStrings" "C:\Program Files (x86)\Microsoft
System Center 2012\Orchestrator\Web Service\Orchestrator2012"
```

Feature Performance Considerations

Despite the great deal of variance in their design and visual complexity, runbooks are essentially very simple. Runbooks all essentially do three things: Run activities, manage published data, and branch.

Orchestrator Feature Functions

Additionally, runbook activities can be thought of as having two distinct types of code: Platform code and domain code. The term *domain code* is used to identify code called within a runbook activity typically not associated with the Orchestrator product itself. For example, the **Invoke Web Service** standard activity would contain Orchestrator platform code (the "plumbing" of the activity) as well as domain code unique to invoking a web service. The platform code will be unique but

similar for most activities, since it is built from a common framework. However, there will potentially be great variation in domain code for different activities.

Essentially, Orchestrator runbooks are designed to pass data between discrete elements of domain code.

While technically not mandatory, every activity generally consumes published data created by prior runbook activities. What a given activity does with published data it subscribes to is entirely up to the domain code.

All runbook activities create published data, which is referred to as *Common Published Data*. Domain code will generally create published data, generally referred to as *Activity Specific Published Data*. This data will be as unique to a given activity as the domain code itself. Also, it's not required that domain code produce published data.

The data produced by a given activity can contain data elements that are single or multi-valued. For example, every activity produces a single record of single-value data referred to as common published data. Domain code can produce multiple records of multi-value data.

Branching is a fundamental concept for Orchestrator. A given runbook activity will create a branch if it is the origin of two or more links whose filter conditions indicate there is data to pass to the activity at the end of the link. When a runbook is first invoked it consists of a single thread. When this thread encounters a runbook activity whose links require a branch, additional threads are created, one for each branch. Each thread takes as input the published data from the activity that created the branch. This data is correlated back to the prior activities in the runbook (hence the ability to subscribe to published data from prior activities).

The operator experience is based on two components: The Orchestration Console and the Web Service. The Orchestration Console is a Silverlight application that depends on the Web Service for its connection to the Orchestrator database. The Web Service is an IIS application that connects to the database. Hence the Web Service and Orchestration Console are very dependent on the performance of the Orchestrator 2012 database.

Parts of the Orchestrator 2012 database are new to the product and directly support the Web Service. However, parts of the Web Service depend on the legacy Opalis 6.3 database structure. Additionally, while the Orchestration Console is dependent on the Web Service, it also has logic unique to its function as a user interface that will have its own performance characteristics.

How to Use the Integration Toolkit to Extend Orchestrator Capabilities

The System Center 2012 Service Pack 1 (SP1) Orchestrator Integration Toolkit is a set of tools to help you create new integrations for Orchestrator. Using wizard-based applications, you can easily create new workflow activities and Integration Packs that extend the capabilities of the product. You can also create custom workflow activities using the Orchestrator SDK and C#, and then package them into an Integration Pack using this toolkit. For complete documentation for the Orchestrator Integration Toolkit, see the MSDN Library.

Using the Orchestration Console in System Center 2012 - Orchestrator

The Orchestration console is a web-based tool with which an operator can perform the following functions in System Center 2012 - Orchestrator:

- View a list of runbooks and runbook servers.
- View the current running status and history of runbooks.
- View high-level definition of runbooks.
- Start and stop runbooks.
- View events that runbook servers and the management server create.

The functions that you can perform in the Orchestration console is a subset of the functions in the Runbook Designer, but you can run the Orchestration console from any computer with a browser without requiring installation of a separate tool. It is intended for users who are required to manage the operation of runbooks but are not required to modify them. To provide users with this access, you have to give them permission to specific runbooks or folders. This process is described in Runbook Permissions.

Orchestration console topics

Overview of Orchestration Console

Describes the basic features of the Orchestration console.

Orchestration Console Browser Requirements

Describes the minimum browser requirements for using the Orchestration console.

How to Start the Orchestration Console

Describes how to start the Orchestration console.

How to Work With Runbooks in the Orchestration Console

Describes how to start, stop, and manage runbooks from the Orchestration console.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Getting Started with System Center 2012 Orchestrator
- Deploying System Center 2012 Orchestrator
- Administering System Center 2012 Orchestrator
- Integration Packs for System Center 2012 Orchestrator Release Candidate
- Using Runbooks in System Center 2012 Orchestrator
- Runbook Activity Reference for System Center 2012 Orchestrator

Overview of Orchestration Console

The Orchestration console is comprised of a single webpage with multiple panes that are described in the following sections.

Navigation pane

The navigation pane is the left pane in the Orchestration console where you can click the workspace that you want to use. Depending on the workspace you click, you can view specific data and use specific options. The following workspaces are available in the navigation pane.

Runbooks workspace

The **Runbooks** workspace lets you start and stop runbooks. You can also view information such as the jobs and instances created for each runbook and their definition.

Summary

The **Summary** tab is displayed for any folder or runbook selected in the **Runbooks** workspace. This tab displays summary information for the jobs and instances of the selected runbook or for all of the runbooks in the selected folder. The statistics that are displayed are updated every 10 minutes so that activity performed within that time might not be reflected in the numbers until they are updated.

Each column in the **Summary** displays the number of jobs and instances that finished with a particular status (Succeeded, Warning, or Failed) within the last hour, the last day, and the last week. For instances, the number of instances that are currently in progress are also displayed. For jobs, the number of jobs that have been created and that are currently queued are also displayed.

Runbooks

The **Runbooks** tab is displayed when you select a folder in the **Runbooks** workspace. It lists the runbooks contained in the selected folder and specifies the status of any running jobs and instances from each. To select one of these runbooks and control their actions, click an option in the **Actions** pane. If you have a large number of runbooks, you can refine the list by specifying a filter.

Jobs

The **Jobs** tab is displayed when you select a folder or runbook in the **Runbooks** workspace. This tab lists the jobs created for a given runbook and the completion status. For a folder, it lists the jobs created for all runbooks in the folder and their completion status. A job is a request for a runbook server to run a runbook and is created every time a runbook receives a request to run. If a runbook starts with a monitor, it creates a job that runs continuously until the runbook is

stopped. In this case, the status of the job shows an hourglass that indicates it is currently running.

Instances

The **Instances** tab is displayed when if you select a folder or runbook in the **Runbooks** workspace. For a runbook, this tab lists the instances that have been created for the runbook and their completion status. For a folder, it lists the instances that have been created for all runbooks in the folder and their completion status. An instance is a running copy of a runbook and is created each time that a runbook runs. If a runbook starts with a monitor, it creates an instance that continues to run until the monitor condition is met. In this case, the status for the instance shows an hourglass. When the monitor condition is met, the instance continues with the subsequent activities and then shows a completion status. The runbook then creates a new instance that also runs until the monitor condition is met.

Runbook Servers workspace

The **Runbook Servers** workspace lets you view the status of current and completed jobs and instances for each runbook server.

Jobs

The **Jobs** tab lists the jobs that have been run on the runbook server and their completion status. A job is a request for a runbook server to run a runbook and is created every time a runbook receives a request to run. If a runbook starts with a monitor, it creates a job that runs continuously until the runbook is stopped. In this case, the status of the job shows an hourglass, which means that it is currently running.

Instances

The **Instances** tab lists the instances that have been created on the runbook server and their completion status. An instance is a running copy of a runbook and is created each time that a runbook runs. If a runbook starts with a monitor, it creates an instance that continues to run until the monitor condition is met. In this case, the status for the instance shows an hourglass. When the monitor condition is met, the instance continues with the subsequent activities, and then shows a completion status. The runbook then creates a new instance that also runs until the monitor condition is met.

Events workspace

The **Events** workspace lets you view log events. By default, log events include all events for the management server and all runbook servers. To limit the events, click **Filter** and provide criteria to limit the events displayed. If an event is specific to a runbook server, it includes the name of the server in the **Source** box. In this case, you can select the event, and then click **View**

Runbook Server in the Actions pane. Clicking View Runbook Server opens the Jobs tab in the Runbook Servers workspace for that runbook server.

Starting and stopping runbooks

In addition to viewing the current status of a runbook, you can also start and stop a runbook from the Orchestration console. When you start a runbook, a job is created and waits for an available runbook server to process the runbook. If the first action in a runbook is a monitor, the job runs continuously, potentially producing multiple instances of a runbook, until the runbook or job is stopped. When a runbook server is available, the job provides an instance of the runbook to the runbook server to process. A running runbook has at least one job and one or more instances associated with it.

When you stop a runbook, the runbook, all jobs, and all instances associated with the runbook are stopped.

Stopping jobs

A job is a request for a runbook to run. A job is created only when you request a runbook to run. If the first action in a runbook is a monitor, the job runs continuously until the runbook or job is stopped. An hourglass indicates the status of a running job. An instance is a running copy of a runbook.

You cannot start a job; you can only start a runbook.

When you view an instance, you can choose to stop the associated job. Stopping the job stops the instance, the job, any other associated instances, and the runbook.

See Also

Orchestration Console Browser Requirements

How to Start the Orchestration Console

How to Work With Runbooks in the Orchestration Console

Orchestration Console Browser Requirements

The Orchestration console can be accessed from any browser that supports Microsoft Silverlight 4. To access the system requirements for Silverlight, see <u>Get Microsoft Silverlight</u>.

Authentication

The Orchestration console requires authentication by using your domain credentials so that it can identify the runbooks and folders that you should have permission to access. If your browser is configured for automatic logon, you are not prompted for a name and password. Your browser supplies this information automatically each time you connect to the Orchestration console. If you are using a browser that does not support automatic logon or if your browser is configured to not perform automatic logon, you are prompted for a name and password each time that you connect to the Orchestration console. You can continue to type your name and password each time, or you can configure your browser to perform automatic logon.

To configure Internet Explorer for automatic logon

- 1. In Internet Explorer, click Tools, and then click Internet Options. The Internet Options dialog box opens.
- 2. On the **Security** tab, select **Local intranet**, and then click **Custom Level**.
- 3. Scroll down to User Authentication, and under Logon, select Automatic logon only in Intranet zone. Click OK.
- 4. In the Internet Options dialog box on the Security Settings tab, with Local intranet still selected, click Sites.
- 5. Click Advanced.
- 6. Type the URL of the Orchestration console server (for example, http://OrchSrv.contoso.com) in the Add this Web site to the zone box, and then click Add.



Note

If the Require server verification (https:) for all sites in this zone is selected, you have to specify https for the address, and your web server has to be configured to support Secure Sockets Layer (SSL).

7. Click **OK** for this and the remaining dialog boxes.

See Also

Overview of Orchestration Console

How to Start the Orchestration Console

How to Work With Runbooks in the Orchestration Console

How to Start the Orchestration Console

The Orchestration console is a web-based tool and is accessed through your web browser. In addition to starting it on the **Start** menu on the computer where the web service is installed, there are two methods for starting the Orchestration console as described in the following procedures.

To start the Orchestration console in a browser

- 1. Open your browser.
- 2. In the address bar, type http://ccomputer name:<port number> where computer name is the name of the server where the web service is installed, and port is the port number selected during configuration of the web service. By default, the port is 82.

To start the Orchestration console in the Runbook Designer

1. Click the Orchestration Console button on the toolbar.



If the URL has not been set for the Orchestration console, you will receive an error message. Use the following procedure to set the URL.

To set the Orchestration console URL in the Runbook Designer

- 1. Select Options, and then select Orchestration Console.
- 2. In the **URL** box, type **http://<computer name>:<port number>** where computer name is the name of the server where the web service is installed, and port is the port number selected during configuration of the web service. By default, the port is 82.
- Click Finish.

See Also

Overview of Orchestration Console

Orchestration Console Browser Requirements

How to Work With Runbooks in the Orchestration Console

How to Work With Runbooks in the Orchestration Console

Use the following steps to run a runbook, to view the status of the jobs and instances of a runbook, and to stop a job in the Orchestration console.

To run a runbook in the Orchestration console

- 1. Click Runbooks to open the Runbooks workspace.
- 2. If the runbook is located in a folder, select the folder in the **Runbooks** pane.
- 3. Click the **Runbooks** tab in the results pane.
- 4. Select the runbook, and then in the Actions pane, click Start Runbook.
- 5. If the runbook requires parameters, they are listed in the **Runbook Parameters** pane.

Click the **Value** column for each runbook and type a value for the runbook to use.

6. If you want to run the runbook on a server other than its default, click a server in the Available Runbook Server(s) pane, and then click the right arrow to add the server to the Selected Runbook Server(s) pane.



Note

If you add multiple servers to the **Selected Runbook Server(s)** pane, the runbook runs only on the first server if it is available. The other servers are backup servers where the runbook runs only if the primary server is not available.

7. Click Start.

To view the status of a runbook in the Orchestration console

- 1. Click Runbooks to open the Runbooks workspace.
- 2. If the runbook is located in a folder, select the folder in the **Runbooks** pane.
- 3. Select the **Runbooks** tab in the results pane.
- Select the runbook.
- 5. To view all jobs that the runbook created, in the **Actions** pane, select **View Jobs**.
- 6. To view all instances that the runbook created, in the Actions pane, select View Instances.

To stop a runbook in the Orchestration console

- 1. Click **Runbooks** to open the **Runbooks** workspace.
- 2. If the runbook is located in a folder, select the folder in the **Runbooks** pane.
- 3. Click the **Runbooks** tab in the results pane.
- 4. Select the runbook, and then in the **Actions** pane, click **Stop Runbook**.
- 5. Click **OK** to the message to confirm that you want to stop the jobs.
- 6. If the runbook was started successfully, you receive a confirmation message that the job was stopped. If the runbook has no running jobs, you receive a message that no job was running.

See Also

Overview of Orchestration Console Orchestration Console Browser Requirements

Orchestration Console Browser Requirements

Using Runbooks in System Center 2012 - Orchestrator

Welcome to the System Center 2012 - Orchestrator Runbook Guide. This document describes Orchestrator runbooks. This document provides information about the tools available in Orchestrator and guidance on how to automate tasks and processes for your IT environment. Topics include how to write, test, and deploy a runbook with System Center 2012 - Orchestrator.

For more information on building custom solutions with Orchestrator in System Center 2012, see System Center 2012 Integration Guide - Orchestrator.

Orchestrator Runbooks

Runbook Concepts

Provides basic information and terminology for runbooks, activities, and workflows.

• <u>Tools</u>

Describes tools to build and start runbooks.

Design and Build Runbooks

Describes how to design and create a runbook.

Deploy and Start Runbooks

Describes how to deploy runbooks to your Orchestrator environment.

Runbook Samples

Provides samples and step-by-step guidance on how to build a runbook.

Other resources for this product

TechNet Library main page for System Center Orchestrator 2012

Getting Started with System Center 2012 - Orchestrator

Deploying System Center 2012 - Orchestrator

Administering System Center 2012 - Orchestrator

Integration Packs for System Center 2012 - Orchestrator Release Candidate

Using the Orchestration Console in System Center 2012 - Orchestrator

Runbook Activity Reference for System Center 2012 - Orchestrator

Runbook Concepts

The power of System Center 2012 - Orchestrator lies in providing runbooks and the individual activities that make up a runbook. Runbooks contain the instructions for an automated task or process. The individual steps throughout a runbook are called activities. Within the runbook,

additional controls provide information and instructions to control the sequence of activities in the runbook. Runbooks, activities, and each runbook control have configurable properties. You modify these properties to configure the behavior that your runbook requires.

The topics in this section provide detailed information about the attributes and features related to runbooks.

Runbook Concepts

Runbooks

Provides configuration information for runbooks.

Activities

Describes categories of activities and their attributes.

Workflow Control

Describes tools to manage a runbook.

Other resources for this product

- TechNet Library main page for Orchestrator for System Center 2012
- <u>Using Runbooks in System Center 2012 Orchestrator</u>
- Tools
- Design and Build Runbooks
- Deploy and Start Runbooks
- Runbook Samples

Runbooks

Runbooks let you use a wide range of customization options. This section provides details for all properties and permission settings for runbooks.

Runbook Attributes

Runbook Properties

Provides configuration information for individual runbooks.

• Runbook Permissions

Describes access rights and permissions for single and multiple runbooks.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Using Runbooks in System Center 2012 Orchestrator

- Runbook Concepts
- Activities
- Workflow Control

Runbook Properties

A runbook is essentially a series of activities that are using data, performing tasks, and publishing data for use by other activities in the runbook. Each runbook has a collection of configurable properties. These properties let you customize the behavior of a runbook.

To view the properties of a runbook

- 1. In the Runbook Designer, in the **Connections** pane, click the **Runbooks** folder.
- 2. If the runbook is stored in a folder, select the appropriate folder under **Runbooks**.
- 3. In the **Runbook Designer** Design workspace, right-click the tab of a runbook to select **Properties**.
- 4. To close the Runbook Properties dialog box, click Finish.

A summary of the runbook properties and how to configure them follows.

General

On the **General** tab of the **Runbook Properties** dialog box, you can customize a name and description for the runbook. You can also associate a schedule with the runbook. After you assigned a schedule to the runbook, the runbook only runs on the dates and times that you specified in the schedule.

To create a schedule

- 1. In the Runbook Designer, in the **Connections** pane, expand the **Global Settings** folder.
- 2. Right-click the **Schedules** folder to select **New** to select**Schedule**.
- On the General tab of the New Schedule dialog box, in the Name box, enter a name for the schedule.
- 4. On the **Details** tab of the **New Schedule** dialog box, select the date and time to start the runbook.

To configure the schedule for specific days of the week

- a. On the **Details** tab of the **New Schedule** dialog box, click **Days of the week**, and then select the days on which you want to start the runbook.
- Under Occurrence, select the week of the month to start the runbook.
 For example, if you want to start the runbook every Monday, under Days of the week, select Monday, and under Occurrence, select First, Second,

To configure the schedule for specific days in the month

- a. On the **Details** tab of the **New Schedule** dialog box, click **Days of the** month.
- b. In the **Days of the month** box, enter the date or dates on which you want to start the runbook.

Separate multiple dates with a comma.

For example, if you want to start the runbook on the first and 15th of every month, enter **1**, **15** in the **Days of the month** box.

To configure the schedule for specific hours

- a. On the **Details** tab of the **New Schedule** dialog box, select **Hours**.
- b. In the **Schedule Hours** dialog box, select the hours on which you want to start the runbook.

You can both allow and deny the start of a runbook during any period. For example, if you want to start a runbook only outside business hours, select the hours of 9 A.M. to 5 P.M. for Monday, Tuesday, Wednesday, Thursday, and Friday, and then click **Denied**.

- 5. On the **Exceptions** tab of the **New Schedule** dialog box, add any date exceptions for the runbook, and then click **OK**.
- 6. Click Finish.

Important

The scheduled date and time to start a runbook is based on the system clock of the runbook server. This enables schedules to function in virtual machine environments and to continue to run even when the system clock adjusts for daylight savings time.

To associate a schedule with a runbook

- 1. In the **Runbook Properties** dialog box, on the **General** tab, click the ellipsis (...) button to browse for a **Schedule**.
- 2. Select a schedule, click **OK**, and then click **Finish**.

Runbook Servers

This tab displays the list of runbook servers assigned to run this runbook. If the list is empty, the runbook uses the setting defined in the **Runbook Servers** folder found in the **Connections** pane

of the Runbook Designer. If the runbook server that uses the Primary role is available, the runbook runs on it. If the primary runbook server is not available, each runbook server that uses a Standby role is checked until one is found that can run the runbook.

You can override the default behavior and assign a primary and any number of standby runbook servers to a runbook. It is useful to assign a specific runbook server to a runbook if the runbook requires access to a specialized resource, such as a backup device.

To assign primary and standby runbook servers to a runbook

- In the Runbook Properties dialog box, on the Runbook Servers tab, select Override default Runbook Server roles to configure primary and standby runbook servers.
- 2. Click Add.
- 3. Select a runbook server, and then click **OK**.
 - The first runbook server that you added becomes the primary runbook server.

4. To add more runbook servers, click Add, and select another runbook server.

- All additional runbook servers are added as standby runbook servers.
- 5. When you are finished adding runbook servers, click **Finish**.

Logging

This feature controls what data is logged to the orchestration database. If stored in the orchestration database, this data is visible in views such as the **Log** pane in the Runbook Designer and in the Orchestration console. This information does not affect the availability of Published Data in a running runbook.

Published Data includes data specific to each activity. For detailed information about the Published Data available in each standard activity, see the Runbook Activity Reference for System Center 2012 - Orchestrator. For detailed information about published data available in integration packs, see System Center Orchestrator 2012 Integration Packs.

Common Published Data is a set of data items that are common to all activities. These items are as follows:

- Activity Name
- Activity Type
- Activity ID
- Activity End Time Year, Month, Day, Weekday, Hours, Minutes, Seconds
- Activity Duration
- Previous Activity
- Previous Activity Name

Caution

When you turn on logging, the size of the orchestration database increases.

Event Notifications

You can enable event notification for the runbook. Notifications appear in views such as the **Log** pane in the Runbook Designer and in the Orchestration console.

If you want to be notified when a runbook runs for more than a specified length of time, enter a value in the **seconds** box.

If you want to be notified if the runbook does not run, click the Runbook fails to run option.

For more information about Event Notifications, see Orchestrator Logs.

Job Concurrency

The job concurrency setting lets you set the maximum number of simultaneous jobs, so that you can carry out multiple requests for the same runbook at the same time. This setting applies to the individual runbook. A runbook server can run 50 runbooks at the same time. If you select a job concurrency setting over 50, your environment requires more runbook servers or the requests to start a runbook will gueue.

The following limitations apply:

- You cannot run simultaneous requests for runbooks that start with Monitoring activities. If you
 try to change the maximum number of simultaneous requests for these runbooks, the
 Runbook Designer resets the Maximum number of simultaneous jobs value to 1 and
 displays an error message.
- A runbook server runs simultaneous requests for runbooks up to the maximum processing limit. To change the maximum processing limit, see How to Configure Runbook Throttling.
- Do not create simultaneous requests for runbooks that contain Modify Counter activities.
 When you run different copies of the same runbook at the same time that modify (set, reset, increment, or decrement), a Counter can cause the Counter value to become unreliable. You can read the value of Counters in runbooks that run at the same time.
- Do not run simultaneous requests for runbooks that interact with a non-Microsoft product, such as a ticketing or system-monitoring tool, unless you have a good understanding of how the tool handles parallel processing. If the non-Microsoft application cannot handle parallel processing, or if you do not know, leave the maximum number of simultaneous requests at a value of 1.
- Plan the use of multiple requests carefully. Before you change the maximum number of simultaneous runbook requests, consider the tasks performed by the runbook. Verify that each runbook instance can finish successfully. For example, if your runbook creates a folder, copies files into it, and then deletes the folder when it is finished, one instance of the runbook might delete the folder before other instances are finished with it. In this case, you should keep the maximum number of simultaneous requests for this runbook a value of 1 to avoid conflicts.

Returned Data

Returned Data defines the data that a runbook returns when it finishes. Each Returned Data definition can contain either a single or multiple parameter values. To populate the data definitions, end the workflow with a Return Data activity that contains the return values.

You access the Returned Data values through Published Data in one of several ways.

- Invoke the runbook from another runbook by using the Invoke Runbook activity. The parent runbook can access the child runbook's Returned Data as Published Data from the Invoke Runbook activity.
- View the Published Data from the Runbook Designer or Orchestration console.
- Use the Orchestrator web service to return the Published Data programmatically.

For more information about the standard activities Invoke Runbook, Initialize Data, and Return Data, see the Runbook Activity Reference for System Center 2012 - Orchestrator.

To define the Returned Data for a runbook to return, use **Add**, **Edit**, and **Remove** to create each parameter.

See Also

Runbook Activity Reference for System Center 2012 - Orchestrator Orchestrator Logs

Runbook Permissions

Runbook access permissions are set through the Runbook Designer. By default, only users in the Orchestrator Users Group have full access to a runbook. You give access to additional users to run, start, stop, view, and change runbooks at either the folder level or the individual runbook level.

To view or modify the permissions of a runbook

- 1. In the Runbook Designer, in the **Connections** pane, click the **Runbooks** folder.
- 2. In the **Runbook Designer** Design workspace, right-click the tab for a runbook to select **Permissions**.
- 3. To give another user or security group access to the runbook, click the **Add** button, and select the user or security group from the local computer or from the domain.
- If the user or security group should be able to view and run the runbook, select the Allow check box next to Read.

If the user or security group should be able to change the runbook, select the **Allow** check box next to **Write**.

If the user or security group should be able to change permissions for the runbook, select the **Allow** check box next to **Full Control**.

5. To close the **Permissions for Runbook** dialog box and save any changes, click **OK**.

See Also

Runbook Concepts

Activities

In System Center 2012 - Orchestrator, activities are the building blocks of runbooks. In general, individual activities perform three actions:

- Access Published Data
- Perform some action
- Publish new data

All activities, regardless of origin or type, share common behaviors. This section describes the types of activities available in Orchestrator and their common behaviors.

Activities

Standard Activities

Describes standard activities available in Orchestrator.

Monitoring Activities

Describes specialized activities that monitor environment states and event logs.

Customized Activities

Describes customization options available in Orchestrator.

Common Activity Properties

Describes configurable properties common to all activities.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Using Runbooks in System Center 2012 Orchestrator
- Runbook Concepts
- Runbooks
- Workflow Control

Standard Activities

An installation of System Center 2012 - Orchestrator includes a set of standard activities. Using these activities, you can create powerful workflows to automate tasks and processes.

For detailed reference information about each standard activity, see <u>Standard Activities</u> in theRunbook Activity Reference for System Center 2012 - Orchestrator.

Standard activities are organized into categories to help you locate the appropriate activity for the task that you want to perform. The following table shows the categories for standard activities.

Activity collection	Description
Email	Supports sending email notifications.
File Management	Manages file interaction such as copying and moving files.
Monitoring	Reacts to system-level events.
Notification	Supports other notification types such as Syslog files and pop-up notifications.
Runbook Control	Manages how runbook logic behaves.
Scheduling	Performs schedule-based activities.
<u>System</u>	Runs system commands such as running a program.
Text File Management	Manipulates text files.
<u>Utilities</u>	Supports working and manipulating data within a workflow.

See Also

Activities

Monitoring Activities

Monitoring activities are a specialized group of activities that are triggered by a state or event of a task outside of a runbook. For example, a monitor can wait for a particular event to occur in an event log, check the IP status of a certain computer, or run repeatedly on a predefined schedule.

An Orchestrator activity cannot trigger a monitoring activity. A monitoring activity is a start condition within a runbook. The Monitor Folder activity waits for the files within a specified folder

to change. When a file changes, the Monitor Folder activity triggers the next activity in the runbook workflow.

Runbooks that start with a monitoring activity load the monitoring activity and wait for the trigger condition to occur. When the monitor activity detects the trigger condition, a runbook instance is created to run the remaining activities. The monitor continues to run and waits for the trigger event to reoccur. Runbooks that start with monitors continue to run until you stop them from the Runbook Designer or the Orchestration console.

For a list of standard monitoring activities, see <u>Monitoring</u> in the <u>Runbook Activity Reference for</u> System Center 2012 - Orchestrator.

See Also

Monitoring

Runbook Activity Reference for System Center 2012 - Orchestrator

Customized Activities

System Center 2012 - Orchestrator provides two options for extending standard activities. Integration packs (IP) are Microsoft and products of other companies that contain additional activities specific to a product or technology. For more information about the currently available IPs, see **System Center Orchestrator 2012 Integration Packs** on Microsoft TechNet. If the functionality that you require is not available in an IP, you have the alternative option of using the Orchestrator Integration Toolkit. This toolkit lets you build an activity to meet your requirements. For more information about the Orchestrator Integration Toolkit, see System Center 2012 — Orchestrator SDK in the MSDN Library.

See Also

Activities

Common Activity Properties

All activities have properties. The Properties dialog box for each activity has multiple tabs that provide access to the settings for the activity. The particular set of tabs varies between activities, but there are several common property types.

Details

This tab contains various properties specific to an activity. Many activities require you to at least enter a computer name, IP address, file name, file path, or file folder location. Details on these

options are provided for each activity in the Runbook Activity Reference for System Center 2012 -Orchestrator.

Run Behavior

This tab contains the properties that determine how the activity handles multi-value Published Data. It also defines the notifications created if the activity fails or runs for an excessive period.

Published Data Behavior

By default, Published Data is passed as multiple individual outputs. You can alternatively specify that all values be flattened into a single comma-delimited value (.csv) file.

When you enable the Flatten feature, you also choose a multi-value formatting option.



The Flatten feature does not flatten data across multiple instances of the same activity. It only flattens multiple values returned from a single instance of the activity.

Flatten behavior	Description
Separate with line breaks	Each item is on a separate line. This is the format for the output text files.
Separate with	Each item is separated by one or more characters, for example, a semicolon (;).
Use CSV format	All items are in comma-separated value format (.csv file), which is useful for importing into spreadsheets or databases.

Event Notifications

Some activities are expected to take a limited amount of time to finish. If the activity does not finish within the specified period, the activity can be stalled or another issue prevents the activity from finishing. You can define the number of seconds to wait for completion of the activity, after which a platform event is sent to report the delay in completion. You can also choose whether to generate a platform event if the activity returns a failure. For more information about event notifications, see Orchestrator Logs.

Event notification setting	Description
Report when the activity runs for more than	Enter the number of seconds of run time to elapse before generating a notification.
Report if the activity fails to run	Select this option to generate a run failure

Event notification setting	Description
	notification.

Security Credentials

The settings on the Security Credentials tab let you specify the account that runs the Runbook Server Service. This is useful when the activity performs activities with resources on a remote computer.

Important

Note that the account used to start the runbook must have permission on the local computer to run successfully.

Important

If you use the <u>Invoke Runbook</u> activity and you modify Security Credentials, the account you use must be a member of the Orchestrator System group to run successfully.

Caution

If permissions on the Orchestrator installation path are changed and the activity's Security Credentials has a custom user account that does not include **Read/Execute** permissions to **ExecutionData.dll** on the runbook server, the activity will fail.

Option	Behavior
Use the security of the account assigned to the service	Select this option to run the activity with the account used by the runbook server. For more information, see Orchestrator Security Planning.
This account	Select this option to run this activity with another account. Specify the account user name and password to run this activity. Verify that the account has the credentials to perform this action. If the credentials you provided fail validation, the account assigned to the runbook server account is used.

See Also

Orchestrator Security Planning

Runbook Activity Reference for System Center 2012 - Orchestrator

Workflow Control

When you build runbooks in System Center 2012 - Orchestrator, it is important to understand the underlying logic of the workflow engine. By using this logic, you can create workflows to automate resource-based jobs and complex data processing tasks.

Workflow Control

The workflow control provides the following controls.

- Starting Point
- Smart Links
- Embedded Loops

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Using Runbooks in System Center 2012 Orchestrator
- Runbook Concepts
- Runbooks
- Activities

Starting Point

A runbook can only have one starting point. A starting point is an activity that automatically runs when the runbook is started. Each activity in the runbook runs after the previous activity in the workflow is completed.

If a runbook starts with any activity other than a monitor activity, the runbook begins processing and attempts to run to completion. If the runbook starts with a monitoring activity, the monitor loads and waits for the trigger condition. When the condition is met, a runbook instance is created to run the remaining activities in the runbook. The monitor continues to run and waits for another occurrence of the trigger condition. Runbooks that start with monitors continue to run until you stop them from the Runbook Designer or Orchestration console.

See Also

Monitoring Activities

Smart Links

The links that connect individual activities in a runbook are called smart links. Smart links in System Center 2012 - Orchestrator support precedence between two activities. Smart links invoke the next activity in the runbook as soon as the previous activity finishes successfully. Smart links also provide filtering capabilities for the data so you can limit the data passed to subsequent activities in the workflow.

Creating and configuring smart links

You can modify the smart link condition properties by double-clicking the smart link. Use the following procedure to enable or disable smart links.

To create a smart link

- 1. In the **Runbook Designer** Design workspace, click and drag two activities from the **Activities** pane to the **Runbook Designer** Design workspace.
- 2. In the **Runbook Designer** Design workspace, hover the mouse cursor over one of the activities, click the Right Arrow, and then drag it to the destination activity.

A line is created between the two activities indicating a smart link is created.

To disable a smart link connection while preserving configured properties

To disable the smart link, right-click the smart link to toggle Enable.
 The smart link changes to a dashed line indicating that it is disabled.

To enable a smart link connection

To enable the smart link, right-click a disabled smart link to toggle Enable.
 The smart link changes to a solid line to indicate that it is enabled.

General Tab

In the **Link Properties** dialog box, on the **General** tab, you can add Name and Description values to the smart link. These properties are not required, but are useful in identifying the purpose of the smart link. These properties are not displayed unless you configure the runbook option to show link labels.

To add a smart link label from the Runbook Designer

- To view the smart link name, right-click a smart link to select Properties.
- In the Properties dialog box, on the General tab, in the Name box, enter a descriptive name.
- Click Finish.

To display smart link names in the runbook

- 1. On the Runbook Designer menu, click Options, and then click Configure to open the Configuration dialog box.
- Select Show link labels.
- Click Finish.

Include and Exclude Tabs

Orchestrator lets you configure conditions for passing data to the following tasks in the runbook. By using link conditions, you can build branching capabilities into your runbooks. For example, a runbook must stop a database server before backing it up. If the database server stops correctly, the runbook starts the backup application. If the database does not stop correctly, an email is sent to the administrator to escalate the issue.

On the **Include** tab, you can specify the conditions that enable data to flow to the next activity in the runbook. The Exclude tab specifies the conditions that cause data to be excluded from the next activity in the runbook.

Important

The rules of the smart link **Exclude** tab supersede the rules on the smart link **Include** tab.

Important

The rules on each tab are joined by using an **or** condition. Only one of the conditions defined on a tab must be true for the condition to be true.

Use the following procedure to add or remove a condition to a smart link.

To add a smart link condition

1. Right-click a smart link to select **Properties** to open the **Link Properties** dialog box.

Important

To change the values that make up the rule, you have to select each underlined portion of the smart link condition.

- 2. Click the listed activity in the condition to open the Published Data dialog box.
- 3. Select the Show common Returned Data box to display properties that are common to all activities.
- 4. Select a property from the Published Data and click OK. The criteria expression is changed depending on the type of data that the property returns.
- 5. To change the different parts of the expression, select the underlined text, and then either select or type in an appropriate value. For more information about criteria, see Smart Link Criteria.
- 6. Click Finish.

To remove a smart link condition

- 1. In the Link Properties dialog box, click either the Include tab or Exclude tab.
- 2. To select the condition that you want to remove, click to the right of the link condition on the word **or**, and then click **Remove**.
- 3. Click Finish.

Options Tab

In the **Link Properties** dialog box, on the **Options** tab, you can specify different link colors on your branches to make them easier to read. For example, you can select green for the **Pass** branch and red for the **Fail** branch to identify the difference logic paths.

On this tab, you can also specify a delay before the activity runs.

Use the following procedure to configure these settings.

To configure smart link colors

- 1. Click Color, and then click the color of the smart link that you want.
- 2. Click **Width** of the smart link line in pixels to specify the width.
- 3. Click Finish.

To configure smart link activity delay

- 1. In the **Trigger delay** box, type the number of seconds that you want the smart link to wait before invoking the next step in the runbook.
- 2. Click Finish.

Smart Link Criteria

Link criteria can be created for any data published from the activity that initiates the link. The type of criteria depends on the type of data returned from the particular property. The following sections provide details on the different types of data that activities can return.

Activity Completion Status

When you add a new criteria to the link, it will default to the completion status of the activity. This status returns one of the following values:

- success
- warning
- failed

Each time you create a new link, it creates a default criteria specifying that the activity's completion status must return success. If you want the next activity to run regardless of whether the first activity successfully finished, you should delete or change criteria.

Binary Values

Some properties return a value of true or false. You can set a criteria of equals or does not equal, and the value prompts you for the two possible values.

Text Values

Certain published data properties return text that you can compare to an expected value or pattern. The following table shows the different criteria that can be used.

Condition	Description
contains	The specified text appears somewhere in the value of the Published Data item.
does not contain	The specified text does not appear somewhere in the value of the Published Data item.
starts with	The value of the Published Data item starts with the specified text.
ends with	The value of the Published Data item ends with the specified text.
matches pattern	The value of the Published Data item matches the specific regular expression.
does not match pattern	The value of the Published Data item matches the specific regular expression.
equals	The value of the Published Data item exactly matches the specified text.
does not equal	The value of the Published Data item does not match the specified text.



Text values are not case-sensitive.



Important

The regular expression criteria have a slightly different behavior than other regular expressions when using the ^ character specifying the starting position in the text and the \$ character specifying the ending position in the text. You must specify a wildcard in

addition to these operators. For example, with the string "This is some sample text", text\$ returns a false, but .*text\$ returns true. Similarly, ^This returns false, but ^This.* returns true.

Numeric Values

Certain published data properties return numeric data that you can compare to an expected value. The following table shows the different criteria that can be used.

Condition	Description
equals	The value of the Published Data item is exactly equal to the specified value.
does not equal	The value of the Published Data item does not equal the specified value.
is less then	The value of the Published Data item is less than the specified value.
is greater then	The value of the Published Data item is greater than the specified value.
is less than or equal to	The value of the Published Data item is less than or equal to the specified value.
is greater than or equal to	The value of the Published Data item is greater than or equal to the specified value.
is between	The value of the Published Data item is between two specified values.

See Also

Workflow Control

Embedded Loops

In System Center 2012 - Orchestrator, looping can be configured for any runbook. By using loops, you can build automatic retries and monitor at any location in a runbook.

Each activity can create a loop so that you can retry operations if they fail or test the output information of the activity for valid data. You can also use these mechanisms to build wait conditions into your workflows.

When a loop is configured for an activity, it continues to run with the same input data until a desired exit looping criteria is reached. The exit criteria is built in a similar way as smart link configurations. You can use any published data item from the activity as part of the exit or do not exit configuration. Included in the common published data are special data items such as **Loop:**Number of attempts and **Loop:** Total duration that let you use information from the loop itself in the looping conditions.

Loops run one time for each incoming piece of data that is passed to the activity. For example, consider a runbook that uses a **Query Database** activity followed by **Append Line**. If the **Query Database** activity returned three rows, the **Append Line** activity would run three times. If you have a loop on the **Append Line** activity, it would run three separate loops. After the first data item has looped through the **Append Line** activity, the next item goes through **Append Line** and loops until it exits, and then the third begins. After all three items have been processed, the next activity in the runbook runs.

Configuring Looping

Use the following procedure to configure looping.

To configure looping

- Right-click an activity in the runbook to select Looping. The Looping Properties dialog box opens.
- 2. On the **General** tab, click **Enable**.
- 3. In the **Delay between attempts** box, type the number of seconds to pause between each attempt to run the activity.

Exit and Do Not Exit Conditions

The rules on the **Exit** tab specify the conditions that determine whether the loop exits. The rules on the **Do Not Exit** tab specify the conditions that cause the loop to continue.



Important

The rules on the **Do Not Exit** tab supersede the rules on the **Exit** tab.

The rules within each tab are joined by using an **Or** condition. Only one of the conditions on a tab must be true for the entire tab to be true.

Use the following procedure to add or remove an **Exit** condition.

To add an exit condition

1. In the **Looping Properties** dialog box, click either the **Exit** tab or **Do Not Exit** tab, and then select the condition listed in the box, or click **Add** to add a condition.



Important

To change the values that make up the rule, you have to select each underlined

portion of the link condition.

- 2. Click the listed activity in the condition to open the **Published Data** dialog box.
- 3. Check the **Show common Returned Data** box to display properties that are common to all activities.
- 4. Select a property from the published data, and then click **OK**. The criteria expression is changed depending on the type of data that the property returns.
- 5. To change the different parts of the expression, select the underlined text and either select or type in an appropriate value. For more information about criteria, see **Smart Link Criteria**.
- 6. Click Finish.

To remove an exit condition

- 1. In the Looping Properties dialog box, click either the Exit tab or the Do Not Exit tab.
- 2. To select the condition you want to remove, click **Or** to the right of the link condition, and then click **Remove**.
- 3. Click Finish.

See Also

Workflow Control

Tools

To create and test a runbook in System Center 2012 - Orchestrator, use the Runbook Designer and the Runbook Tester.

- Runbook Designer
 - Create, manage, and run runbooks.
- Runbook Tester

Step through a runbook to test its functionality.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Using Runbooks in System Center 2012 Orchestrator
- Runbook Concepts
- Design and Build Runbooks
- Deploy and Start Runbooks
- Runbook Samples

Runbook Designer

The **Runbook Designer** is the tool that you use to create, manage, and run runbooks in System Center 2012 - Orchestrator. The **Runbook Designer** is intended for users who must create or modify runbooks. Users who only have to run runbooks and view their status should use the Orchestration console which is documented in the <u>Using the Orchestration Console in System Center 2012</u> - Orchestrator.

Runbook Designer Panes

The Runbook Designer interface is organized into the following four panes.

Pane	Description
Connections	The folder structure where you can organize workflows in the Orchestrator system and edit permissions on folders. Also provides access to Runbook Servers and Global Settings .
Runbook Designer workspace	The workspace where you build Orchestrator runbooks. The runbooks in the folder selection in the Connections pane are listed as tabs across the top of the workspace. When you select a tab in a runbook, it is displayed in the Runbook Designer workspace.
Activities	Contains all the activities available (either standard activities or activities available from integration packs) for use in runbooks. You drag activities from the Activities pane into the Design workspace, and then link them together to form runbooks.
Log	Logs showing the activity and history for the current runbook. For more information, see Orchestrator Logs.

Sorting Activities by Activity Name and Category Name

System Center 2012 - Orchestrator lets you sort activities alphabetically by activity name, or by category name. By default, activities are sorted by category, such as Runbook Control, Email,

File Management, Monitoring, Notification, Scheduling, System, Text File Management, and Tools.

Use the following steps to sort activities by their activity name and category name.

To sort activities alphabetically by activity name

In the Activities pane, right-click a category name to select All Activities.
 The activities are sorted alphabetically by activity name.

To sort activities alphabetically by category name

In the Activities pane, right-click a category name to select Default.
 The activities are sorted alphabetically by category name.

Changing Icons

You can change the default size of each activity icon from small to large. Use the following steps to change the icon size.

To change the icon size

• In the **Activities** pane, right-click an activity name to select **Small** or **Large** depending on the size of icon that you want to view.

See Also

Tools

Runbook Tester

Runbook Tester lets you test runbooks in a debugging environment. You can run an entire runbook, step through it one activity at a time, or add breakpoints to stop the simulation at any activity you select. You start Runbook Tester from the toolbar above the central Design workspace in the **Runbook Designer**. When you click the Runbook Tester button, the Runbook Tester starts and loads the current runbook. The button is only enabled if the runbook is not currently running. You must stop the runbook before you can test it.

Important

Runbook Tester actually performs each activity within the workflow. The steps are not performed in a simulated or virtualized environment. All the connections referenced in the runbook are live and fully functional, so any activities that modify or destroy data in connected systems cause that data to be modified or destroyed. For example, if you use

the Query Database activity to DROP TABLE ImportantTable, it actually deletes the ImportantTable from the instance of Microsoft SQL Server.



Important

Note that the account used to start the runbook must have permission on the local computer to run successfully. These permission requirements also apply when testing the runbook with the Runbook Tester. To successfully test your runbook, start the Runbook Designer as Administrator. By association, the Runbook Tester runs as Administrator and uses the higher-level security token.

Runbook Tester panes

The Runbook Tester interface is organized into the following four panes.

Pane	Description
Run Time Properties	Displays run-time information, including resolved published data items, variables, and computer groups, about the activity that is currently being processed by Runbook Tester. Information appears in this pane when the runbook runs with breakpoints or in stepthrough mode.
Design Time Properties	Displays design-time information about each activity in the runbook when the runbook runs without breakpoints and is not run in stepthrough mode. To view the design-time properties of an activity, click an activity in the runbook.
	You cannot edit the information that appears in the Design Time Properties pane.
Workspace	Displays the active runbook. You can select each activity to view its information in the Design Time Properties pane or to set a breakpoint on it.
Log	Displays information about each activity in the runbook as it runs. You can click the Show Details link to show the configuration details and published data from the activity.

Pane	Description
Resource Browser	Displays the counters, variables, computers groups, and schedules that the runbook in the workspace uses.

See Also

Tools

Design and Build Runbooks

This section provides details about how to design, build, and test, runbooks by using System Center 2012 - Orchestrator.

Runbooks

Designing a Runbook

Provides design guidance for building runbooks.

Building a Runbook

Describes how to create a runbook, how to add and link activities, and how to configure runbook properties.

• How to Test a Runbook

Describes how to test a runbook.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Using Runbooks in System Center 2012 Orchestrator
- Runbook Concepts
- Tools
- Deploy and Start Runbooks
- Runbook Samples

Designing a Runbook

When you plan a new runbook, you should start with a defined process that you want to automate. This process determines your choice of runbook activities. Specifically, determine the following:

- When and how often is the runbook going to run?
- What steps make up the workflow?
- What activities reflect the steps in my workflow?
- What type of data is required to begin the workflow?
- What data are generated from each activity?
- What results are produced at the end of the workflow?
- How are the runbook results reported?

Consider the following points as you design your runbook:

- 1. Failure and Warning links It is important to handle all results from an activity. An activity provides a default success string, but does not provide a default failure case. Consider if you should reverse an activity or write the result to a log file.
- 2. Replace the default strings When you look at the workflow in a runbook, the labels should identify what the individual activities are doing. Rename links and activities labels to a descriptive name.
- 3. Link colors Change the color of your links when there is a condition or branch. It is common to use GREEN as success and RED for warning or failed. You should use standard associations, but not use too many colors or you lose their descriptive purpose.
- 4. Limit the number of activities per runbook Too many activities in a single runbook make it difficult to administer and troubleshoot. Consider splitting a runbook into several subtasks and create child runbooks for each of those subtasks. You can invoke the child runbooks from a parent runbook. You can reuse these child runbooks in other workflows.
- 5. Runbook logs By default, logging options are disabled for runbooks. When you enable logging, the data significantly increases the size of your database. As an alternative, you can log to an external system or file.

See Also

Design and Build Runbooks

Building a Runbook

This topic describes the basic process for building a System Center 2012 - Orchestrator runbook.



Note

For a list of topics that contain more details about the information covered here, see Runbook data processing.

Step	Description
1. Create a runbook.	Create an empty runbook in the Runbook
	Designer.

Step	Description
2. Add activities.	Click and drag activities from the Activities pane into the runbook. Include a start point and an end point for the runbook.
3. Link activities.	Create and configure smart links between each of the activities to create a complete workflow.
4. Configure runbook properties.	Configure the properties for the runbook.
5. Check in the runbook.	Save your changes and check in the runbook.

To create a new runbook

- 1. On the computer where the Runbook Designer is installed, click **Start**, point to **All** Programs, click System Center 2012 - Orchestrator, and then click Runbook Designer.
- 2. In Runbook Designer, in the **Connections** pane, click the **Runbooks** folder.
- 3. In the **Connections** pane, click the **Create a new runbook** icon.
- 4. In the Runbook Designer Design workspace, right-click the Runbook tab, and then select Rename.
- 5. In the Confirm Check out dialog box, click Yes.
- 6. Enter a name for the runbook, such as **Sample Runbook**, and press Enter.

To add and configure activities to your runbook

- 1. In the **Activities** pane, drag an activity to the Design workspace of your runbook.
- 2. In the Activities pane, double-click an activity to open the Properties dialog box for that activity.



Note

For information about specific properties of standard activities, see the Runbook Activity Reference for System Center 2012 - Orchestrator.

To add and configure links in a runbook

- 1. To create a link, click and drag the arrow of an activity to another activity.
- 2. On the newly created link, double-click the link to open the Link Properties dialog box.



For information about the properties of links, see the Runbook Activity Reference for System Center 2012 - Orchestrator.

To define the properties of a runbook

- Right-click the Runbook tab to select Properties. The Runbook Properties dialog box opens.
- 2. Configure the settings on the **General** tab. The following tables provide the configuration instructions.
- 3. Click **Finish** to save your settings.

To check in your runbook

• In Runbook Designer, click the **Check In** icon on the toolbar.

Runbook data processing

- Data Manipulation
- Published Data

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Using Runbooks in System Center 2012 Orchestrator
- Designing a Runbook
- How to Test a Runbook

Data Manipulation

With System Center 2012 - Orchestrator, you can manipulate string data from text files, returned data, or other sources, and convert it into a usable form. You can also perform simple arithmetic operations, such as calculating sums and differences, and performing division and multiplication operations. For example, you can extract text from a text file by using a **Text File Management** activity, trim leading and trailing spaces from the text, and then retrieve specific parts of the text that you can pass to other activities as returned data items.

Data Manipulation

- Computer Groups
- Counters
- Functions
- Regular Expressions
- Schedules
- Variables

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Using Runbooks in System Center 2012 Orchestrator
- Building a Runbook
- Published Data

Computer Groups

System Center 2012 - Orchestrator is designed to interact with all of your data center systems. Computer groups let you target selected activities against a set of similar computer systems instead of a single computer. By configuring the activities in your runbook to use a computer group, you have the flexibility to add computers dynamically by adding them to the computer group.

You can create computer groups by using Active Directory queries, and you can manage the list of computers in a group outside of Orchestrator. For example, if you have a computer group that is created from an Active Directory query that retrieves all instances of Microsoft SQL Server, when an instance of SQL Server is added to your Active Directory system, it is automatically included in that group.

Managing Computer Groups

To use computer groups in your activities, create a computer group, and then add computers to it. You can also organize your computer groups into folders. Use the following steps to create a new folder.

To create a folder

- 1. In the **Connections** pane in the Runbook Designer, click the **Computer Groups** folder or a subfolder.
- 2. Right-click to select **New**, and then click **Folder**.

Use the following procedure to add a computer group. To add computers by using an Active Directory query or a System Center 2012 Configuration Manager collection, use the Active Directory Integration Pack or the Integration Pack for System Center 2012 Configuration Manager.

To add a computer group

- 1. In the Connections pane, right-click the Computer Groups folder or a subfolder.
- 2. Select **New**, and then click **Computer Group** to open the **New Computer Group** dialog box.
- 3. In the New Computer Group dialog box, on the General tab, in the Name and

Description boxes, type a name and description of the computer group.

- 4. Click the **Contents** tab. The list displays all the computer entries that make up this computer group.
- 5. Click Add to open the Add Computer to Computer Group dialog box.
- 6. Enter the name of the computer that you are adding, or click the ellipsis (...) button next to the **Computer** box, and then select the applicable computer. Click **OK** to add the computer.
- 7. To add more computers to the group. repeat the previous two steps.

To modify settings

- 1. To modify the settings of an entry you added, click the entry on the **Contents** tab, and then click **Modify**.
- 2. To remove an entry on the **Contents** tab, click the entry, and then click **Remove**.

Using a Computer Group in an Activity

Any standard activity that requires you to identify a **Computer** name in the **Configuration Properties** dialog box, such as the **Send Event Log Message** activity, can use a computer group. Other activities can use the **Computer Group** where you define a remote system or computer.

Use the following procedure to use a computer group.

To use a computer group

- 1. Right-click the applicable activity from your runbook, select **Properties** on the menu, and then select the **Details** tab to open the **Activities Properties** dialog box.
- 2. In the **Computer** box, right-click to open a menu, select **Subscribe**, and then select **Computer Group** to open the **Select Computer Group** dialog box.
- 3. Select the computer group, and then click **OK**.

A placeholder {computer group name} is inserted next to the computer name in the Computer box.

When the activity runs, it runs on each computer in the group.

See Also

Data Manipulation

Counters

When building runbooks in System Center 2012 - Orchestrator, you might find that there are values that must be incremented, such as keeping track of the number of backup attempts that a

runbook made. Counters let you modify and check the status of a number that you can use to keep track of important statistics. You create a counter in the Connections pane in the Runbook Designer, and then get and modify it by using the Get Counter Value and Modify Counter activities. Each of these activities presents the value of the counter as Published Data so that it can be used by other activities and links.

The access permissions for counters can be modified, but the Runbook server does not enforce these permissions.



Warning

You cannot run multiple, simultaneous jobs for runbooks that contain Modify Counter activities because simultaneous jobs of the same runbook that modify (set, reset, increment, or decrement) a counter can cause the counter value to become unreliable. You can, however, read the value of counters in runbooks that run simultaneously.



Important

Orchestrator does not support moving multiple counters with multiple-selection. To move more than one counter to another folder, you must move each counter individually.

Use the following procedures to create a counter and to organize counters.

To create a counter

- 1. In the Connections pane, double-click the Global Settings folder, right-click the Counters folder or a subfolder of the Counters folder to select New, and then click Counter to open the New Counter dialog box.
- 2. In the **Name** box, type a name for the counter.
- 3. In the **Description** box, type a description that explains the purpose of the counter.
- 4. In the **Default Value** box, type the starting value of the counter. This value is the starting value of the counter when it is created or reset.
- 5. To modify a counter, double-click the counter. To remove a counter, right-click the counter to select **Delete**.
- 6. Click Finish.

To organize counters

- 1. You can group counters in folders to organize them. To create a folder, right-click the Counters folder to select New, and then click Folder.
- 2. To move a counter to a different folder, right-click the counter to select **Move** to open the **Select a Folder** dialog box.
- 3. Select the destination folder, and then click **OK**. The counter is moved to the new folder

location.

See Also

Get Counter Value

Modify Counter

Published Data

Functions

By using System Center 2012 - Orchestrator, you can manipulate string data from text files, Published Data, or other sources, and convert it into a usable form. You can also perform simple arithmetic operations, such as calculating sums and differences, and performing division and multiplication operations. For example, you can extract text from a text file by using a **Text File Management** activity, trim leading and trailing spaces from the text, and then retrieve specific parts of the text that you can pass to other activities as returned data items.

For a complete list of the functions that you can perform, see the following Functions table.

Data Manipulation Functions

You can insert a data manipulation function into any box that lets you type text. Data manipulation functions must be enclosed in square brackets ('[' and ']'). For example:

```
[Upper('this will be inserted in upper case')]
```

When the activity runs, the text 'this will be inserted in uppercase' in the example is replaced with 'THIS WILL BE INSERTED IN UPPERCASE'.

Nested Functions

If you want to use a data manipulation function within another function, you do not have to enclose the nested function in square brackets. For example, to nest the **Field** function, type:

```
[Field(Field('username=jsmith@abcompany.com','=',2),'@',1)]
```

Functions

Functions are case-sensitive. For example, Upper('Text') will be processed, but upper('Text') will not.

Function and Definition	Usage	Parameters	Example
Upper - converts	Upper('Text')	Text - the text	Upper('this will be converted to

Function and Definition	Usage	Parameters	Example
text to uppercase.		that is being converted to uppercase.	uppercase') returns 'THIS WILL BE CONVERTED TO UPPERCASE'
Lower - converts text to lowercase.	Lower('Text')	Text - the text that is being converted to lowercase.	Lower('This Will Be Converted To Lowercase') returns 'this will be converted to lowercase'
Field - returns text in a specific position.	Field('Text', 'Delimiter', Field Number)	Text - the text that is being searched. Delimiter - the character that	Field('John;Smith;9055552211', ';', 2) returns 'Smith'
		separates each field. Field Number - the position of the field that is being returned (starting at 1).	
Sum - returns the sum of a set of numbers.	Sum(firstNumber, secondNumber, thirdNumber,)	Number - the number that is being added. You can put any set of numbers, each separated by a comma (,).	Sum(2,3,4,5) returns '14'
Diff - returns the difference of two numbers.	Diff(Number1, Number2, <precision>)</precision>	Number1 - the number that will be subtracted from.	Diff(9, 7) returns '2' Diff(9.3, 2.1, 2) returns '7.20'
		Number2 - the number that will be subtracted from Number1.	
		Precision <optional> - the number of decimal places</optional>	

Function and Definition	Usage	Parameters	Example
		that the result will be rounded to.	
Mult - returns the product of a set of numbers.	Mult(firstNumber, secondNumber, thirdNumber,)	Number - the number being multiplied. You can put any set of numbers, each separated by a comma (,).	Mult(2, 3, 4) returns '24'
Div - returns the quotient of two numbers.	Div(Number1, Number2, <precision>)</precision>	Number1 - the number that will be divided. Number2 - the number that will divide Number1.	Div(8, 4) returns '2' Div(9, 2, 2) returns '4.50'
		Precision <optional> - the number of decimal places that the result will be rounded to.</optional>	
Instr - returns the position of first occurrence of text within another text.	Instr ('SearchText', 'TextToFind')	SearchText - the text that is being searched. TextToFind - the text that you are searching for.	Instr('This is a string that is searched', 'string') returns 11
Right - returns a subset of the text from the right side of the full text.	Right('Text', Length)	Text - the full text. Length - the number of characters from the right side that will be returned.	Right('Take from the right', 9) returns 'the right'
Left - returns a subset of the text from the left side	Left('Text', Length)	Text - the full text. Length - the	Left('Take from the left', 4) returns 'Take'

Function and Definition	Usage	Parameters	Example
of the full text.		number of characters from the left side that will be returned.	
Mid - returns a subset of the text from the middle of the full text.	Mid('Text', Start, Length)	Text - the full text. Start - the starting position in the text where you want to begin returning characters. Length - the number of characters starting from the Start position that will be returned.	Mid('Take from the middle', 5, 4) returns 'from'
LTrim - trims leading spaces from text.	LTrim('Text')	Text - the text that is being trimmed of leading spaces.	LTrim(' Remove the leading spaces only. ') returns 'Remove the leading spaces only. '
RTrim - trims the trailing spaces from text.	RTrim('Text')	Text - the text that is being trimmed of trailing spaces.	RTrim(' Remove the trailing spaces only. ') returns ' Remove the trailing spaces only.'
Trim - trims leading and trailing spaces from text.	Trim('Text')	Text - the text that is being trimmed.	Trim(' Remove leading and trailing spaces. ') returns 'Remove leading and trailing spaces.'
Len - returns the length of text.	Len('Text')	Text - the text that is being measured.	Len('Measure this text') returns 17

See Also

Data Manipulation

Regular Expressions

In System Center 2012 - Orchestrator, regular expressions let you match a string to a pattern. The regular expression can contain a number of different elements that define the pattern. **Smart Link Properties** use regular expressions to perform pattern matching.

Advanced Regular Expressions

To build regular expressions, you must create an expression that contains the text that you are searching for and special characters that create a pattern, which describes how the text that you are searching for appears.

Character	Meaning
	Matches any character except a newline.
*	Matches the preceding item 0 or more times. For example, the "a*" pattern matches any string of a's in a row "a", "aaa", "aaaaaaaaaaaa", and an empty string "". To match any string of any character, use a dot followed by an asterisk. For example "a.*" matches any text that begins with the letter "a" and ends with any string of characters such as "abbb", "abcdef", or "automatic restart".
+	Matches the preceding item 1 or more times. This is like * but you must have a least 1 of the preceding item to make a match. For example, the "ab+" pattern matches "abbbbb", "ab", but does not match "a". To contrast, the "ab*" pattern matches "a".
?	Matches the preceding item 0 or 1 time. For example, the "ab?" pattern matches "a" or "ab" but does not match "abbb".
	Matches either the preceding expression or the following expression. Logical OR operator.
\$	Matches the expression at the end of the input

Character	Meaning
	or line. For example, "ab\$" matches "I took a cab" or "drab" but does not match "absolutely not".
^	Matches the expression at the beginning of the input or line. For example, "^ab" matches "absolutely not" or "abacuses are great!" but does not match "I took a cab" or "drab".
\	For characters that are usually treated as special. This indicates that the next character is literal and is not to be treated as a special character. For example, "\." means match the "." character and not just any character.
[]	A character set. Matches any one of the enclosed characters. You can specify a range of characters by using a hyphen. For example, [a-zA-Z] matches any letter of the alphabet.
[^]	An excluded character set. This is the opposite of []. If any of the characters inside the brackets exist, the regular expression match fails. You can specify a range of characters by using a hyphen. For example, [^a-zA-Z] ensures that none of the letters in the alphabet are present.
()	A group expression. This groups an expression into an item that you can apply special characters to. For example, "a*(ba)+" matches "ba" "aba" or "ababa" but does not match "abbba" or "abaa"

Examples

Expression	Meaning
[a-zA-Z]+	The text contains only letters of the alphabet.
^/*	The text begins with an asterisk.
(abc def)\$	The end of the text is either "abc" or "def".
Нау	The text begins with "Ha" followed by any two

Expression	Meaning	
	characters followed by a "y".	
Help.*	The text is "Help" followed by any number of other characters.	

See Also

Data Manipulation

Schedules

System Center 2012 - Orchestrator uses schedules to define the times when runbooks can run. For example, there are times when it is inappropriate to run some runbooks, such as backing up a runbook on a main server during regular business hours. You can create a schedule that runs according to a complex interval, such as the first and third Mondays and Thursdays of every month, except when these days fall on a holiday.

Schedules use the system clock of the Runbook server that runs the runbook. This enables schedules to function in virtual machine environments, and to continue running even when the system clock is adjusted because of the move to or from daylight savings time.

Runbooks that start before a prohibited time run until finished, even if they are still processing when the prohibited time arrives. They will not be interrupted after processing has started.

Security

The access permissions for schedules can be modified, but the runbook server does not enforce these permissions.



Note

If a runbook is scheduled to start during an hour that is skipped when the system clock is adjusted forward by one hour, that starting time is skipped, and the runbook starts at the next scheduled time. If a runbook is scheduled to start during an hour that occurs two times because the system clock is adjusted backward by one hour, the runbook starts two times.



Note

Orchestrator does not support moving multiple schedules with multiple-selection. To move more than one schedule to another folder, you must move each schedule individually.

Conditional Links

In addition to assigning a schedule to a runbook, you can use a <u>Check Schedule</u> activity to use a schedule for conditional logic in a runbook. This activity checks a particular schedule and returns a published data item with a value of true or false specifying whether the current time is within the schedule. This published data item can be used by a link to determine whether to run a particular activity or to continue to the workflow.

Creating a schedule and assigning the schedule to a runbook

Use the following procedures to create a schedule, to assign the schedule to a runbook, or remove a schedule from a runbook.

To create a schedule

- In the Connections pane, right-click the Schedules folder or a subfolder of the Schedules folder, point to New, and then click Schedule to open the New Schedule dialog box.
- 2. On the **General** tab, in the **Name** box, type a name for the schedule.
- 3. In the **Description** box, type a description that describes or explains the purpose of the schedule
- 4. Click the **Details** tab. Select the days that this schedule allows runbooks to run:

Days of week: Select this option and select the days of the week when this schedule allows runbooks to run.

Occurrence: Select the weeks of the month when the schedule allows runbooks to run.

Days of month: Select this option and select the days of the month when this schedule allows runbooks to run. Specify the days of the month by entering the number of the day. You can use hyphens to describe ranges and commas to separate entries. For example, typing 1,3 includes the first and third day of the month. Typing 1-21 includes the first through to the twenty-first day of the month. You can combine both to create complex descriptions of the days of the month. Type all to specify all days of the month. Type last to specify the last day of the month.

You cannot use **all** and **last** as part of a range of days. Additionally, if you typed a range of 5-31, this range works correctly for all months, including those with 28, 29, 30, and 31 days.

- 5. Click **Hours** to open the **Schedule Hours** dialog box.
- 6. Click and drag to select a group of hours in a week. The text at the bottom of the dialog box shows the time period that you selected. Then select one of the following:

Permit (blue): assigns the time period that you selected as a time when runbooks are allowed to run.

Denied (white): assign the time period that you selected as a time when runbooks are not

allowed to run.

- 7. Click OK.
- 8. Click the **Exceptions** tab. The list displays all the days that are exceptions to the rules defined in the **Details** tab.
- 9. Click Add to open the Date dialog box.
- 10. Specify the date and select **Allow** or **Disallow** to allow or not allow the runbook to run on that day, and then click **OK**. The entry appears in the list.
- 11. To modify an Exception entry, select it, and then click **Modify**. To remove the Exception entry, select it, and then click **Remove**.
- 12. To modify a schedule, double-click the **Schedule**.
- 13. To remove a schedule, right-click the **Schedule**, and then select **Delete**.
- 14. Click Finish.

To assign a schedule to a runbook

- 1. Right-click the runbook tab, and then click **Properties** to open the **Runbook Properties** dialog box.
- 2. On the **General** tab, click the ellipsis (...) button to open the **Select a Schedule** dialog
- 3. Select the schedule that you want to apply to the runbook, and then click **OK**.
- Click Finish.

Every time the runbook is started, it checks the schedule to verify that it is allowed to run. If it is not allowed to run, it stops and does not restart until the next time it is started.

To remove a schedule from a runbook

- Right-click the runbook tab, and then click **Properties** to open the **Runbook Properties** dialog box.
- 2. On the **General** tab, click the ellipsis (...) button to open the **Select a Schedule** dialog box.
- 3. Do not select a schedule. Click **OK**.
- 4. Click **Finish**. The schedule is removed from the runbook.

See Also

Published Data
Check Schedule

Variables

When building runbooks in System Center 2012 - Orchestrator, some settings are the same across activities. Variables let you specify a value that activities use in any runbook.

Security

The access permissions for variables can be modified, but the runbook server does not enforce these permissions.

Important

Be aware that in Orchestrator, variables that reference system variables, for example **%ProgramFiles%**, return values from a 32-bit runtime environment. This is because Orchestrator is a 32-bit application.

Note

Orchestrator does not support moving multiple variables with multiple-selection. To move more than one variable to another folder, you must move each variable individually.

Use the following procedures to create, insert, and organize variables.

To create a variable

- 1. In the **Connections** pane in the Runbook Designer, expand the **Global Settings** folder, and then click the **Variables** folder.
- 2. Right-click the **Variables** folder or a subfolder of the **Variables** folder to select **New**, and then click **Variable** to open the **New Variable** dialog box.
- 3. In the **Name** box, type a name for the variable.
- 4. In the **Description** box, type a description that explains the purpose of the variable.
- 5. In the **Value** box, type the value of the variable. This value replaces the placeholder in those activities where the variable is inserted.
- 6. If you want the variable to be encrypted (for example, to store a password for use in other runbook activities), select the **Encrypted Variable** check box.
 - For more information about best practices for using encrypted variables, see <u>Orchestrator</u> Data Encryption.
- 7. Click Finish.

Important

System Center 2012 - Orchestrator does not let you combine an encrypted variable with plain text as a parameter value in a runbook.

To insert a variable in an activity

- 1. Right-click the applicable activity from your runbook to select **Properties**, and then click the **Details** tab to open the activities properties dialog box.
- 2. In a text box, to open a menu, right-click to select **Subscribe**, and then click **Variable** to

open the **Select a Variable** dialog box.

3. Select the variable name, and then click **OK**.

A placeholder {variable} is inserted next to the computer name in the **Computer** box. When the activity runs, the placeholder is replaced with the value of the variable.

To organize variables

- 1. You can group variables into folders to organize them. To create a folder, right-click the **Variables** folder to select **New**, and then click **Folder**.
- 2. To move a variable to a different folder, right-click the variable, and then click **Move** to open the **Select a Folder** dialog box.
- Select the destination folder, and then click **OK**. The variable is moved to the new folder location.

Special Variables

You can specify special formats of variables to provide dynamic information to your runbooks. Specify the value of the variable to invoke this behavior.

NOW(): When the variable is resolved, it is set to the current date and time. You can pass arguments to this function to return specific portions of the date or time. For example, NOW(hour) returns the current hour. The following are the valid arguments for the NOW() function: day, dayofweek, dayofyear, month, year, hour, minute, second, millisecond.

%ENVVAR%: This variable returns the value of the environment variable between the percent (%) symbols. The environment variable is based on the runbook server computer where the runbook is running, and it is not case-sensitive. All system variables can be resolved. Any user variables are resolved in the context of the service account on the runbook server. If the environment variable does not exist, the text specified within the variable is returned as-is (that is, if you type %ENVVAR% and no environment variable named ENVVAR exists, the text '%ENVVAR%' is returned).

See Also

Data Manipulation

Published Data

Published Data lets an activity use information from another activity in the same runbook. Each activity has a specific set of Published Data items that it populates after it runs. Any other activity that runs after it in the workflow has access to this data. In addition to data specific to each activity, all activities publish a common set of data items that provide information such as the start

and stop time of the activity and its completion status. Link conditions also use Published Data to add filtering and decision-making capabilities to runbooks.

For example, the runbook might use a **Read Line** activity to get information from a text file. A **Send Email** activity later in the runbook has to use the information to include in the text of its mail. The **Send Email** activity could use the **Line Text** Published Data item from the **Read Line** activity to include in its mail message.

Data Types

The following table describes the categories of Published Data value types.

Published Data value type	Description
String value	Text, for example, an error message description.
Date value	Date and time information. For example, the date and time that a specific error occurred.
Number value	Numeric information. For example, the number of rows returned by a database query.
Boolean value	true or false. For example, command completed.

Date and time characteristics

Activity Start Time and Activity End Time data is saved to the databus in two fields. These formats are local time and Coordinated Universal Time (UTC), both in ISO 8601 format. By using UTC, runbooks can run in either a non-locale-specific or a non-time-zone-specific context. Only Published Data that is saved to the databus provide date and time information in UTC with ISO 8601 formatting.

The date and time values displayed in the Runbook Designer and the Orchestration console, including, but not limited to the Log History, Audit History, and Events respect the locale date and time format configured for your computer.

Published Data with multi-value types

When an activity in a runbook runs, it runs one time for each item of data that the previous activity produced. For example, the **Query Database** activity runs and retrieves three rows from the database. These three rows of data cause the next activity to run three times, one time for each row returned. This next activity does not have to subscribe to the data for this action to occur.

An activity can also retrieve information from an outside source. The **Get** activities and **Monitor** activities demonstrate this behavior. Data output from an activity might be a list of computers for

example. Data can be passed on as multiple individual outputs, which invoke the next activity as many times as there are items in the output.

You also have the option of passing on data as a single output. For information about how to configure Published Data with multiple values, see <u>Common Activity Properties</u>.

Adding Returned Data to Activity Configurations

When an activity has subscribed to Published Data, a placeholder is inserted where the value of the data will be added. An activity can only subscribe to Published Data from a previous activity in the workflow.

Use the following procedures to add Published Data to an activity, to change the Published Data subscription, and to copy and paste Published Data items.

To subscribe to Published Data of an earlier activity in the workflow

- 1. Right-click an activity from your runbook to select **Properties**, and then click the **Details** tab to open the activity's properties dialog box.
- 2. To open a menu, in the text box, right-click to select **Subscribe**, and then click **Published Data** to open the **Published Data** dialog box.
- In the Activity list, select the activity that returns the data that you want to subscribe to.
 By default, the dialog box only displays Published Data that is specific to that activity. To
 include Published Data that is common to all activities, click Show common Published
 Data.
- 4. Select the **Published Data** item that you want to use, and then click **OK**.

To change the Published Data subscription

- 1. In the text box, click the data placeholder to open the **Published Data** dialog box.
- In the Activity list, click the activity that returns the data that you want to subscribe to. By default, the dialog box only displays Published Data that is specific to that activity. To include Published Data that is common to all activities, click Show common Published Data.
- Click the **Published Data** item that you want to use, and then click **OK**. The Published Data placeholder changes to reflect the new activity and Published Data that you selected.

To copy and paste Published Data items

- 1. Find a Published Data item that has already been inserted into a box in the **Properties** dialog box of an activity.
- Select the Published Data item that you want to copy.
- Use the keyboard shortcut CTRL+C, or right-click the selected item, and then click Copy.
- 4. Open the **Properties** dialog box to which you want to copy the Published Data item.
- 5. Place your cursor where you want the Published Data item to appear and use the

keyboard shortcut, CTRL+V, or right-click the insertion point, and then click **Paste**. The Published Data item appears.

Transforming Published Data Items

You might have to modify text from a **Published Data** activity before you use it in another activity. For example, you might have to remove a portion of the text or replace it with another string. You can transform the existing Published Data content or variable items into new content according to rules that you specify by using the <u>Map Published Data</u> activity.

Common Published Data

The following table describes the Published Data items common to all activities.

Name	Description
Activity ID	The unique identifier of the activity. For example, {4BD3F27A-8F1B-4F60-8245-F69469075EF1}.
Activity name	The name of the activity as it appears in the workspace. If you customize the name of an activity in the workspace, the customized name appears here.
Activity Process ID	The process ID of the job process where the activity runs.
Activity status	The result status of running the activity, for example, Success.
Activity type	The default name of the activity. It does not change from the default even if you rename the activity in the workspace, and it can be useful in identifying an activity in runbooks where activity names and display icons have been changed.
Error summary text	A summary of the error information that the activity returns.
Runbook name	The name of the runbook.
Runbook Process ID	The process ID of the runbook module's executable program that is running on the runbook server.
	The job process contains the logic for the activity. It is started when the runbook server

Name	Description	
	starts the runbook, and it is stopped when the runbook is stopped. Each runbook runs in its own job process executable program.	
Server name	The name of the runbook server where the runbook is running.	
Activity duration	The total time that the activity was running.	
Activity end time	The time when the activity finished.	
Activity end time (year)	The year when the activity finished.	
Activity end time (month)	The month when the activity finished.	
Activity end time (day)	The day when the activity finished.	
Activity end time (weekday)	The day of the week when the activity finished.	
Activity end time (hours)	The hour when the activity finished.	
Activity end time (minutes)	The number of minutes past the hour when the activity finished.	
Activity end time (seconds)	The number of seconds past the minute when the activity finished.	
Activity end time in UTC	The time when the activity finished in UTC format.	
Activity end time in UTC (year)	The year when the activity finished in UTC format.	
Activity end time in UTC (month)	The month when the activity finished in UTC format.	
Activity end time in UTC (day)	The day when the activity finished in UTC format.	
Activity end time in UTC (weekday)	The day of the week when the activity finished in UTC format.	
Activity end time in UTC (hours)	The hour when the activity finished in UTC format.	
Activity end time in UTC (minutes)	The number of minutes past the hour when the activity finished in UTC format.	
Activity end time in UTC (seconds)	The number of seconds past the minute when the activity finished in UTC format.	

Name	Description
Activity start time	The time when the activity started.
Activity start time in UTC	The time when the activity started in UTC format.
Loop: Delay between attempts	The amount of time (in seconds) between each loop attempt.
Loop: Enabled	The setting that determines whether per-activity looping is enabled for the activity.
Loop: Loop error message	The error message if the loop is not successful.
Loop: Number of attempts	The number of iterations that the loop has been through.
	The name of the runbook to which that the activity belongs.
Loop: Total duration	The total amount of time (in seconds) that the looped activity ran.

See Also

Building a Runbook

How to Test a Runbook

After you build a runbook, you can test it before it is run in production. To test, you use the **Runbook Tester** which you start in the **Runbook Designer**. The **Runbook Tester** lets you run the runbook to view the Published Data from each activity. You can run through the entire runbook, step through each activity one at a time, or set breakpoints at certain activities.

To test a runbook

- 1. In the **Runbook Designer**, open the runbook, and on the menu bar, click **Runbook Tester**.
- 2. If prompted, click **Yes** to check out the runbook.
- 3. To run through the runbook from beginning to end without stopping, click **Run to Breakpoint**.
 - If you want to step through it one activity at a time, click **Step**.
- 4. View the **Log** pane to see the completion status of each activity. To view the details and Published Data from an activity, select the activity, and click **Show Details**.

To set a breakpoint

- 1. Select the activity on which to set the breakpoint.
- 2. Click Toggle Breakpoint.
- 3. Click Run to Breakpoint.
 - Each activity up to the breakpoint runs. The runbook pauses before running the activity with the breakpoint.
- 4. To continue through to the end of the runbook, click **Run to Breakpoint** again, or to step through it one activity at a time, click **Step**.

See Also

Design and Build Runbooks

Deploy and Start Runbooks

This section describes how to deploy and start runbooks in your environment.

Using runbooks

- Deploying Runbooks
- Running Runbooks

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Using Runbooks in System Center 2012 Orchestrator
- Runbook Concepts
- <u>Tools</u>
- Design and Build Runbooks
- Runbook Samples

Deploying Runbooks

There are tools available in Orchestrator to help you manage the versions of your runbooks. These tools are described in the following sections.

Version Control

In System Center 2012 - Orchestrator, multiple users can create and update runbooks. However, only one user at a time can make changes to a runbook. This protects your work from being overwritten by someone else with the same permission level.

To edit a runbook, you must check it out. Another user cannot edit that runbook until you either commit all changes by checking the runbook in or revert all changes by undoing the checkout.

Check In and Check Out

- Check Out: When a user is editing a runbook, the runbook is checked out and cannot be edited by anyone else. If someone else is already editing the runbook, a pop-up window opens informing you that someone is already editing the runbook.
- Check In: When a user editing the runbook performs a Check In operation, all changes that were made are committed, and other users can then edit the runbook after they check it out. Check in comments describe the changes that have been made.
- Undo Check Out: When a user editing the runbook performs an Undo Check Out operation, all changes that were made are reverted after the runbook was checked out. After the Undo Check Out operation is completed, another user can edit the runbook.

Audit Log

When a runbook has been changed and is checked in by a user, an entry appears in the Audit History log.



₩ Tip

When a runbook has been altered to a state where it is no longer functioning, you can select the Audit History tab at the bottom of the Runbook Designer to see the changes that were made and then reverse them.

To view runbook change details

- 1. In the Runbook Designer, select the Audit History tab at the bottom, double-click the entry item to open the **Details** dialog box.
- 2. In the **Name** column, click each item in the list to view the changes that were made.
- 3. When you select an item, the Action type displays beneath the Activities box. For example, Action: Modified or Action: Added. When you select the Action: Modified type, the Attribute, Old Value, and New Value are listed in the bottom text box.

See Also

Deploy and Start Runbooks

Running Runbooks

This topic describes the process for starting runbooks, viewing the results, and stopping a job in the Runbook Designer.

To start a runbook

- 1. In the Runbook Designer, in the **Connections** pane, click the **Runbooks** folder to see the available runbooks.
- 2. In the Design workspace, click a runbook tab.
- 3. If the runbook is Checked Out, select the Check In button.
- 4. In the Design workspace, right-click the runbook tab and select **Run**.
- 5. In the **Start Runbook** dialog box, go to **Available Runbook Server(s)** box and select the applicable server.
- 6. Click the Arrow button so that the server name is now in the **Selected Runbook Servers(s)** box.
- 7. Click Start.

To find events

- 1. Click **Start**, point to **All Programs**, click **Administrative Tools**, and then click **Event Viewer** to open a session.
- 2. On the **Event Viewer** menu, double-click **Windows Logs**, and then on the menu, click **Application**.

To stop a job from the Runbook Designer

- 1. In Runbook Designer, click the Monitor Runbook tab.
- 2. On the toolbar, click Stop.

See Also

Deploy and Start Runbooks

Runbook Samples

This topic provides instructions about how to build and test sample runbooks by using the <u>Standard Activities</u> found in System Center 2012 - Orchestrator.

Samples

- Creating and Testing a Sample Runbook
 - Provides step-by-step instructions about creating and testing a simple runbook.
- Monitor a Folder within a Runbook
 - Provides instructions about how to create a runbook that monitors the activity in a folder.

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- <u>Using Runbooks in System Center 2012 Orchestrator</u>
- Runbook Concepts
- Tools
- Design and Build Runbooks
- Deploying Runbooks

Creating and Testing a Sample Runbook

The following topic describes how to create and test a simple runbook. The purpose of this runbook is to detect when a text file is added to a particular folder, copy that file to another folder, read the contents of the file, append a line from the copied file to another file, and then delete the original file.

The runbook starts with a **Monitor File** activity to wait for the text file to be created. It then uses the **Copy File**, **Read Line**, **Append Line**, and **Delete File** activities to perform the other functions. A **Junction** activity is used to coordinate the activities so that the **Copy File** and **Append Line** activities are both completed before the source file is deleted.

Creating the runbook

Use the following procedures to create the runbook by using the required activities.

To create a runbook

- Click Start, point to All Programs, click Microsoft System Center 2012, click Orchestrator, and then click Runbook Designer.
- $2. \quad \text{In the \textbf{Connections} pane, right-click \textbf{Runbooks}} \ \text{to select \textbf{New}}, \ \text{and then click \textbf{Runbook}}.$
 - A **New Runbook** tab appears at the top of the **Runbook Designer** Design workspace with the name **New Runbook**.
- Right-click the New Runbook tab to select Rename.
 In the Confirm Check out dialog box, click Yes.
- 4. Type **Append and Copy Workflow** in the **Input** box, and then press Enter.

You have created a new runbook and are ready to begin adding and configuring activities.

To add and configure a Monitor File activity

- 1. With the newly created **Append and Copy Workflow** runbook open, in the **Activities** pane, expand the **File Management** category.
- 2. Click and drag the Monitor File activity to the Runbook Designer Design workspace.
- 3. Double-click the **Monitor File** activity to open its **Properties** dialog box.
- 4. In the **In folder** box, type **C:\Drop**.
- 5. In the **Filters** section, click the **Add** button.
- 6. In the **Filter Settings** dialog box, in the **Name** list, select **File Name**.
- 7. In the Relation list, select Matches Pattern.
- 8. In the **Value** box, type *.txt.
- 9. Click OK.
- 10. Click the Triggers tab.
- 11. In the **Trigger if one of the files was** section, select the **Created** check box, and then click **Finish**.

The **Monitor File** activity is created and configured to watch for any new text files that are created in the C:\Drop folder.

To add additional activities to the runbook

- 1. In the **Activities** pane, expand the **File Management** category.
- 2. Click and drag the **Copy File** activity to the **Runbook Designer** Design workspace.
- 3. Expand the **Text File Management** category.
- 4. Click and drag the **Read Line** activity to the **Runbook Designer** Design workspace.
- 5. To create a link between the **Monitor File** activity and the **Copy File** activity, click and drag the right arrow of the **Monitor File** activity to the **Copy File** activity.
- 6. To create a link between the **Monitor File** activity and the **Read Line** activity, click and drag the right arrow of the **Monitor File** activity to the **Read Line** activity.

By adding both the **Read Line** activity and the **Copy File** activity, you have created a workflow.

To configure the Copy File activity

- 1. In the **Append and Copy Workflow** runbook, right-click the **Copy File** activity to select **Properties**.
- 2. On the **Details** tab, right-click the **File** box to select **Subscribe**, and then click **Published Data** to open the **Published Data** dialog box.
 - The **Monitor File** activity is listed at the top of the **Published Data** dialog box because this is the activity just before to the selected activity.
- 3. In the Name column, select Name and path of the file, and then click OK. This

populates the **File** property of the **Copy File** activity with the name of and path to the file from the **Monitor File** activity.

- 4. In the destination **Folder** box, type **C:\Copy**.
- 5. Click Finish.

The **Copy File** activity is now configured to copy files from the source folder to the destination folder.

To configure the Read Line activity

- 1. In the **Append and Copy Workflow** runbook, right-click the **Read Line** activity to select **Properties**.
- 2. On the **Details** tab, right-click the **File** box to select **Subscribe**, and then click **Published Data** to open the **Published Data** dialog box.
- 3. In the **Activities** list, select **Monitor File**.
- 4. In the Name column, select Name and path of the file, and then click OK.
- 5. Click the ellipse (...) button to the right of the File encoding box, and then select auto.
- 6. In the **Line numbers** box, type **1-END**, and then click **OK**.
- 7. Click Finish.

The Read Line activity is now configured.

To add an Append Line activity

- 1. In the **Activities** pane, expand the **Text File Management** category.
- 2. Click and drag the **Append Line** activity to the **Runbook Designer** Design workspace to the right of the **Read Line** activity.
- 3. To create a link from the **Read Line** activity to the **Append Line** activity, click and drag the right arrow of the **Read Line** activity to the **Append Line** activity.
- 4. Right-click the **Append Line** activity to select **Properties**.
- 5. On the **Details** tab in the **File** box, type **C:\Copy\Masterlog.txt**.
- 6. Click the ellipse (...) button to the right of the **File encoding** box, and then select **auto**.
- 7. Right-click the **Text** box to select **Subscribe**, and then click **Published Data** to open the **Published Data** dialog box.
- 8. In the Name column for the Read Line activity, select Line text, and then click OK.
- 9. Click Finish.

The **Append File** activity is now configured to append files to the **Masterlog.txt** file.

To synchronize branches of a runbook

- 1. In the **Activities** pane, expand the **Runbook Control** category.
- 2. Click and drag the **Junction** icon to the **Runbook Designer** Design workspace.
- 3. To create a link from the **Append Line** activity to the **Junction** activity, click and drag the right arrow of the **Append Line** activity to the **Junction** activity.

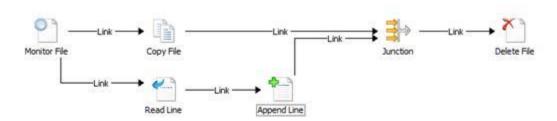
- 4. To create a link from the **Copy File** activity to the **Junction** activity, click and drag the right arrow of the **Copy File** activity to the **Junction** activity.
- 5. Right-click the Junction activity to select Properties.
- Click the ellipse (...) button next to the Return data from box, and then select Copy File.
 Click OK. This action configures the activity to return the same Published Data as the Copy File activity.
- 7. Click Finish.

The **Junction** activity is configured to coordinate the workflow so that no further activities run until both the **Copy File** activity and **Append Line** activity finish.

To add and configure the Delete File activity

- 1. In the **Activities** pane, expand the **File Management** category.
- 2. Click and drag the **Delete File** icon to the **Runbook Designer** Design workspace.
- To create a link from the Junction activity to the Delete File activity, click and drag the right arrow of the Junction activity to the Delete File activity.
- 4. Right-click the **Delete File** activity to select **Properties**.
- 5. Right-click the **Path** box to select **Subscribe**, and then click **Published Data** to open the **Published Data** dialog box.
 - In the Activity list, select Copy File.
- 6. In the Name column, select Name and path of the original file, and then click OK.
- 7. Click Finish.

The **Append and Copy Workflow** runbook is now completed. It should look similar to the following illustration.



Testing the runbook

You can test the runbook by using the Runbook Tester. This tool lets you run the entire runbook and inspect the completion status and output of each activity. The Runbook Tester runs the activities, so you must first create the folders specified for the runbook.

To test the runbook

- 1. Create a folder on the runbook server called C:\Drop.
- 2. Create a folder on the runbook server called C:\Copy.
- 3. With the **Append and Copy Workflow** runbook selected in the Runbook Designer, on the toolbar, click **Runbook Tester**.
- 4. Click **Run To Breakpoint**. The **Monitor File** activity is loaded and waits for a text file to be created in the C:\Drop folder.
- 5. Open **Notepad** and type a few lines of text. Save the file as C:\Drop\File1.txt.
- 6. Wait a few moments for the other activities to run. Ensure that each of the activities is completed successfully.
- 7. To view the Published Data and other details of an activity, click **Show Details** for the activity.
- 8. Open the C:\Drop folder and ensure that the file has been removed.
- 9. Open the C:\Copy folder and ensure that the file has been copied. Also verify that the MasterLog.txt file has the contents of the original file.

See Also

Runbook Samples

Monitor a Folder within a Runbook

This sample shows you how to create a simple monitor runbook that monitors a folder for new text files. When a file is detected, the runbook sends an event log message, and then starts another runbook.



Create and test a monitor runbook

The procedures to create, configure, and test a sample runbook that monitors a folder are described below.

To create the workflow

 In the Runbook Designer Connections pane, right-click the Runbooks folder to select New, and then click Runbook.

- 2. Right-click the **New Runbook** tab to select **Rename**.
- 3. In the Confirm Check out dialog box, click Yes.
- 4. Type a name for the runbook, such as **Monitor Runbook**, and then press Enter.
- In the Activities pane, click File Management to expand the category, and then drag the Monitor Folder activity into the Runbook Designer Design workspace.
- In the Activities pane, click Notification to expand the category, and then drag the Send
 Event Log Message activity into the Runbook Designer Design workspace, to the right
 of the Monitor Folder activity.
- 7. In the **Runbook Designer** Design workspace, move your pointer over the right side of the **Monitor Folder** activity to display the smart link arrow.
- 8. Click the smart link arrow, and then drag it to the **Send Event Log Message** activity.
- In the Activities pane, click Runbook Control to expand the category, and then drag the Invoke Runbook activity into the Runbook Designer Design workspace, to the right of the Send Event Log Message activity.
- 10. In the **Runbook Designer** Design workspace, move your pointer over the right side of the **Send Event Log Message** activity to display the smart link arrow.
- 11. Click the smart link arrow, and then drag it to the **Invoke Runbook** activity.

To configure the workflow

- 1. In the Runbook Designer Design workspace, double-click the Monitor Folder activity.
- 2. In the **Monitor Folder Properties** dialog box, click the **General** tab.
- 3. In the **Name** box, change the name of the activity to something informative, for example **Monitor C:\Monitor Folder**.
- 4. Click the **Details** tab.
- 5. On the **Details** tab, in the **Path** box, type the path of the folder you want to monitor, for example **C:\Monitor**.
- 6. Below the **File Filters** list, click **Add**.
- 7. In the **Filter Settings** dialog box, set the following:



- a. In the Name list box, select File Name.
- b. In the Relation list box, select Matches pattern.
- c. In the Value box, type *.txt.

This setting directs the monitor to look for files with the **txt** extension. This field accepts regular expression syntax.

- 8. Click OK.
- 9. Select the **Triggers** tab.
- 10. Select the **Number of files is** option, set the value in the list to **greater than**, and then type **0** in the edit box.

- 11. Click Finish.
- In the Runbook Designer Design workspace, double-click the Send Event Log Message.
- 13. In the Send Event Log Message Properties dialog box, on the Details tab, in the Properties section, set the following:



- a. In the **Computer** box, type the name of the computer to receive the Event message.
 - This is typically the computer where you are running Runbook Designer.
- b. In the **Message** box, type the message to display in the Event log, for example, File Detected.
- c. Leave the Severity level at Information.
- 14. Click Finish.



Note

In this sample, the Invoke Runbook activity is not configured. For more information about configuring this activity, see the Invoke Runbook activity in the System Center 2012 - Orchestrator Runbook Activity Reference.

To modify runbook settings

- 1. Above the Runbook Designer Design workspace, right-click the Monitor Runbook tab to select Properties.
- 2. In the Monitor Runbook Properties dialog, click the Logging tab, and then select both Store Activity-specific Returned Data and Store Common Returned Data.
- 3. Click Finish.
- 4. Right-click the Monitor Runbook tab to select Check In.

Test the runbook

In the Runbook Tester, you can test runbooks in a simulated runtime and debugging environment. You can run an entire runbook, step through it one activity at a time, or add breakpoints to stop the simulation at any activity that you select.

Use the following steps to test your runbook in the Runbook Tester.

To prepare your computer

- 1. Right-click Start to select Open Windows Explorer.
- 2. Create a C:\Monitor folder on your computer.
- 3. Create a **C:\Source** folder on your computer.
- 4. In the C:\Source folder, create a file with a txt extension, for example text.txt.

To test the runbook

- 1. In the Runbook Designer Design workspace, select the Monitor Runbook tab.
- 2. On the toolbar above the **Runbook Designer** Design workspace, click **Runbook Tester**.
- 3. In the Confirm Check out dialog box, click Yes.
- 4. In Runbook Tester, on the toolbar, click Step Over to start stepping through the runbook.



∏ Tip

To increase the size of the **Log** pane, remove the **Resource Browser** pane by selecting View on the menu, and then clearing the Resource Browser option.

- 5. In Windows Explorer, browse to the C:\Source folder.
- Copy test.txt to C:\Monitor.
- 7. Close Windows Explorer.
- 8. On the Runbook Tester toolbar, click **Next**.

After a few moments, note that the **Log** pane entry updates and shows an event for the Monitor Folder activity.

- 9. On the **Log** pane Click the **Show Details** link to see the contents of the data bus for that
- 10. Scroll down the list of properties. Note that the activity status is success indicating that the Monitor Folder activity detected the change in the folder.
- 11. On the Runbook Tester toolbar, click Next.

Notice that the Log pane changes and shows an event for the Send Event Log Message activity.

- 12. Click the Show Details link and note that the activity status is success indicating that the **Send Event Log Message** activity detected the change in the folder.
- 13. Close Runbook Tester.
- 14. On the **Runbook Designer** toolbar, click **Check In**.

See Also

Runbook Samples

Runbook Activity Reference for System Center 2012 - Orchestrator

System Center 2012 - Orchestrator includes an extensive set of standard activities that enable you to create runbooks to automate your data center procedures. This guide contains detailed information about using each of the standard activities

Standard Activities

Standard Activities

Describes all of the Orchestrator Standard Activities.

Privacy

Orchestrator is committed to protecting your privacy, while delivering software that brings you the performance, power, and convenience you want. For more information, see the Privacy Orchestrator.

For more information about the Orchestrator Release Candidate, see <u>Release Notes for System Center 2012 - Orchestrator</u>.

Other resources for this product

- 1. TechNet Library main page for System Center Orchestrator 2012
- 2. Getting Started with System Center 2012 Orchestrator
- 3. <u>Deploying System Center 2012 Orchestrator</u>
- 4. Administering System Center 2012 Orchestrator
- 5. Integration Packs for System Center 2012 Orchestrator Release Candidate
- 6. Using Runbooks in System Center 2012 Orchestrator
- 7. Using the Orchestration Console in System Center 2012 Orchestrator

Standard Activities

Activities are organized into categories to help you find the appropriate activity for the task you want to perform. The following table provides a brief description of tasks you can accomplish with each activity category.

Tasks	Categories
Run system commands.	<u>System</u>
Perform schedule-based activities.	Scheduling
Monitor processes or system-level events.	Monitoring
Manage file interactions such as copying and moving files.	File Management
Send e-mail notifications.	<u>Email</u>
Support other notification types.	Notification

Tasks	Categories
Search for or modify data within a workflow.	<u>Utilities</u>
Manipulate text files.	Text File Management
Manage workflows.	Runbook Control

See Also

Common Activity Properties
 Learn how to configure common tab settings.

• Alphabetical List of Standard Activities

View all activities in alphabetical order.

Alphabetical List of Standard Activities

All standard activities are listed below.

Α

Append Line

Apply XSLT

С

Check Schedule

Compare Values

Compress File

Copy File

Create Folder

D

Decompress File

Delete File

Delete Folder

Delete Line

Disconnect Network Path

Ε

End Process

F

Find Text

Format Date/Time

G

Generate Random Text

Get Computer/IP Status

Get Counter Value

Get Disk Space Status

Get File Status

Get Internet Application Status

Get Lines

Get Process Status

Get Service Status

Get SNMP Variable

ı

Initialize Data

Insert Line

Invoke Runbook

Invoke Web Services

J

Junction

M

Map Network Path

Map Published Data

Modify Counter

Monitor Computer/IP

Monitor Counter

Monitor Date/Time

Monitor Disk Space

Monitor Event Log

Monitor File

Monitor Folder

Monitor Internet Application

Monitor Process

Monitor Service

Monitor SNMP Trap

Monitor WMI

Move File

Move Folder

Ρ

PGP Decrypt File

PGP Encrypt File

Print File

Q

Query Database

Query WMI

Query XML

R

Read Line

Read Text Log

Rename File

Restart System

Return Data

Run .Net Script

Run Program

Run SSH Command

S

Save Event Log

Search and Replace Text

Send Email

Send Event Log Message

Send Platform Event

Send SNMP Trap

Send Syslog Message

Set SNMP Variable

Start/Stop Service

W

Write to Database

Write Web Page

Ports and Protocols of Standard Activities

Orchestrator standard activities can communicate between the runbook servers where the runbook is deployed and any resource. If you have firewalls in your environment, when you use a

standard activity, you must enable the ports between the runbook servers and resource as indicated in the following table.

Standard activity	Port on runbook server	Port on resource server	Notes
Query Database		Any port the target database requires.	
Write to Database		Any port the target database requires.	
Invoke Web Services	HTTP or HTTPS	HTTP or HTTPS	
Map Network Path			Activity uses Microsof t Window s file sharing.
Set SNMP Variable	SNMP	SNMP	
Get SNMP Variable	SNMP	SNMP	
Monitor SNMP Trap	SNMP	SNMP	
Send SNMP Trap	SNMP	SNMP	
Run Program			Activity uses Microsof t Window s file sharing and I/O

Standard activity	Port on runbook server	Port on resource server	Notes
			pipes.
Send Email	SMTP	SMTP	
Monitor Internet Applicatio n	HTTP/SMTP/POP3/FTP/DNS	HTTP/SMTP/POP3/FTP/DNS	
Get Internet Applicatio n Status	HTTP/SMTP/POP3/FTP/DNS/Cust om	HTTP/SMTP/POP3/FTP/DNS/Cust om	Custom can be anything
Send Syslog Message	syslog	syslog	

Other resources for this product

- TechNet Library main page for System Center Orchestrator 2012
- Runbook Activity Reference for System Center 2012 Orchestrator
- Alphabetical List of Standard Activities

See Also

TCP Port Requirements

System

The following table provides a brief description of tasks you can accomplish when using each System activity.

Tasks	System Activities
Run any program or command on any computer in your domain.	Run Program
Run scripts that parse data or run functions against available APIs.	Run .Net Script

Tasks	System Activities
End processes that are running on the runbook server or on a remote computer.	End Process
Start, stop, pause, or restart a Windows service.	Start/Stop Service
Restart a computer on your network.	Restart System
Save entries from an event log so that they can be used later.	Save Event Log
Send a Windows Management Instrumentation (WMI) query to a system that you specify and then return the results.	Query WMI
Open an SSH connection to a remote server and run shell commands on that server.	Run SSH Command
Query a network device for the value of variable that is assigned to the Management Information Base (MIB) address you specify.	Get SNMP Variable
Wait for an event to occur either in the Microsoft SNMP Trap Service or on a port that you specify.	Monitor SNMP Trap
Raise an SNMP event that can be detected by a network systems manager application.	Send SNMP Trap
Modify a variable that is specified by its Management Information Base (MIB).	Set SNMP Variable

Run Program

The Run Program activity runs any program or command on any computer in your domain in interactive or background mode. Use this activity to run backup applications or a batch script that runs a set of complex commands.

Configuring the Run Program Activity

Before you configure the Run Program activity, you need to determine the following:

- The command line argument or program that will run and which computer it will run on.
- You also need to determine whether a user on the target computer will need to interact with the program when it is run. The user account that will run the program or command must have administrator rights to run programs on the target computer.

Use the following information to configure the Run .NET Script activity.

Details

Settings	Configuration Instructions
Program execution	Select this mode to run a program in the same way as a Windows shortcut or the Windows Run dialog box. You can also use the ellipsis () button to browse for the computer.
Command execution	Select this mode to run a command in the same way as the Windows Command Prompt.
Computer	Type the computer where this program or command will run.
Program path	 If you selected the Program execution mode, this element appears as Program path. Type the full path to the location of the program that you want to run. Then, to pass parameters to the program, type them in the Parameters box. If you selected the Command run mode, this element appears as Command. Type the path of the command that you want to run, and include the parameters that you want to pass to the command on the same line in the Command box.
Parameters	Type the parameters that will be passed to the program that you want to run. This option is only available when you select the Program execution mode.
Working folder	Type the full path of the working folder that the program or command will use. The command or program will behave as if it was run from the working folder.

Advanced

Settings	Configuration Instructions
Execution mode	Select one of the following execution options for the program: Interactive: Select this option to display a user interface on the computer where the command or program is run. A user interface, if available, appears in a user session that is defined by the user credentials specified in the Run as boxes (User name, Password) on the Advanced tab. Background, normal priority: Select this option to run the command or program in the background with the process priority set to normal. In this mode no user interface will be displayed. Background, low priority: Select this option to run the command or program in the background with the process priority set to low. In this mode no user interface will be displayed. Some programs may not function correctly when set to low priority. If this is the case, use the Interactive or Background, normal priority settings instead.
Wait for the completion of the program	Select this option to cause the Run Program to wait for the program or command to finish running before moving to the next activity in the runbook. If you have set the Execution mode to Interactive , then the user must close the program before the Run Program activity is able to move to the next activity in the runbook.
Terminate after	Type the maximum number of minutes to wait for the program or command to complete. Set this value to 0 to have the Run Program activity to wait indefinitely for the completion of the program or command. If the time has expired and the program or command has not completed running, the Run Program activity will shut down the program or command and report a failure.

Settings	Configuration Instructions
Do not wait for the completion of the program	Select this option to cause the Run Program activity to run the program or command and not wait for it to complete. When this option is selected, the published data items generated by the Run Program activity will not be available to other activities.
User name	To use a different account name to log in to a computer and run a program, type the account number in the User name box. Note This user name only logs in to the computer where the Run Program activity runs, and uses the interactive logon type. If the program that the Run Program activity launches accesses resources on other computers, the same user name is used on the remote computer, but with the network logon
Password	Type the password associated with the user name to run the program on the remote computer.

Published Data

The following table lists the published data items.

Item	Description
Program path	The program path or command that was entered.
Program parameters	The parameters that were passed to the program. This option is only available when Program run is selected on the Details tab.
Working folder path	The path of the working folder.
Process ID	The process ID of the application that was started when the Run Program activity runs. If you are using Command run , this will be the

Item	Description
	process ID of the Windows Command Prompt application.
Program exit code	The return code of the application that was run by the Run Program activity.
Computer	The name of the computer where the application was started.
Program output	The text that was sent to the console when the program was run.
Pure Output	The unmodified output of the program.
Program output file	The name of the local file where the program output was saved.
UNC program output file	The name of the file where the program output was saved in UNC format.

Security

The Run Program activity is based on PsExec. PsExec lets you execute processes on other systems, complete with full interactivity for console applications. For more information on PsExec, go to PsExec.

The Run Program activity inherits certain security concerns from PsExec. Specifically, PsExec uses named pipes. This can be a security concern, as credentials can be sent through this tool.

A work-around for customers concerned about security vulnerabilities is to create a mapped drive to the server that is the target of the Run Program activity. This establishes a security context for the Run Program activity.

Run .Net Script

The Run .Net Script activity runs scripts written in VB.NET, JScript, C#, and Windows PowerShell. This activity is compatible with .NET CLR version 2.0 and later. Use the Run .Net Script activity to run scripts that parse data or run functions against available APIs.

Configuring the Run .Net Script Activity

Before you configure the Run .Net Script activity, you need to determine the following:

- The code you want to run.
- The libraries you want to use.

The data you want to publish.

Use the following information to configure the Run .Net Script activity.



You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

Details Tab

Settings	Configuration Instructions
Туре	Select the script language. Use the ellipsis () button to browse for the language.
Script	Type the code that will run when the activity runs.

Advanced Tab

Settings	Configuration Instructions
Namespace	Add a Namespace for each .NET namespace that will be used within your code. This allows you to call the code without using fully qualified names for each of the classes. Orchestrator recommends adding System namespace to every Run .Net Script activity.
References	Add each of the Assembly (DLL) references that contain the libraries that you want to use. Add the System.dll located in the Windows\Microsoft.NET\Framework\<.NET Version> directory.

Published Data

Add the published data items that you want this activity to publish. Every published data item that you add will be available on the Data bus. It is important to determine if a published data item will be multi-valued. The Run .Net Script activity automatically correlates multi-valued data from different items by aligning them. For example, if you choose to publish two items labeled "Name" and "Email" as Collections, the Run .Net script will try to line up each item in the Name collection with each item in the Email collection. If the collections are not equally sized, then the Run .Net Script activity will create blank values for the collection that has fewer items. For a list of data

items and the corresponding description published by this activity, see the following Published Data table.

Published Data Tab

Settings	Configuration Instructions
Name	Enter the Name of the published data. This will be the name that appears when other activities subscribe to the data published by the Run .Net Script activity.
Туре	You can select Date/Time , Integer , or String . If the type you want is not available, select String . Use the ToString method of the activity to assign a value to this published data.
Collection	If your data is multi-valued data, select Collection. When using a collection you must use the Add method to add items to the collection. If you are not using the collection you can use the assignment operator (=) to assign the value.
Variable name	Use unique naming to make sure that your variable name does not collide with existing variables within your script or with classes and keywords available in .NET. We recommend prefixing variables with "OPD_". For example, if you want to name your variable "myString", you would name it "OPD_myString". The Run .Net Script activity will automatically create a .NET Property for this item. If this variable is a collection it will be created using a List <t>, where T is the Type that you selected. If it is not a collection the property will be created using a String, Integer, or Date/Time based on the Type that you selected.</t>

Published Data

Item	Description
Standard Error	Any standard error output published by the Run .Net Script activity.

Item	Description
Namespaces	The namespaces used.
Standard Output	The standard output published by the Run .Net Script activity.
References	The Assemblies used in the activity.
Script Body	The script that was run.
Script Language	The language that was selected for the script.

End Process

The End Process activity ends processes that are running on the runbook server or on a remote computer. The End Process activity can be used to shut down an application that is not responding. The activity returns **success** if the named process is successfully ended or if the name process is not running. This activity uses a satellite license.

Configuring the End Process Activity

Before you configure the End Process activity, you need to determine the following:

- · Name or ID of the process
- Computer on which it is running

Use the following information to configure the End Process activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the computer where this process is running. Enter localhost to specify the runbook server where the runbook is being processed. You can also use the ellipsis () button to browse for the computer.
Process	Type the name or process ID of the process that you are ending. You can also use the ellipsis () button to browse for the process. Browsing is only available if you have specified a valid Computer .
End all instances	Select to end all processes that match the Process that you have specified when multiples

Settings	Configuration Instructions
	are found.
Fail if there is more than one instance	Select to cause the end process to fail if it finds more than one process matching the name you specified.
Terminate in	Type the number of seconds to wait for the process to be shut down gracefully before it is shut down forcefully.

Published Data

The following table lists the published data items.

Item	Description
Number of instances	The number of processes that matched the Process you specified.
Process ID	The process ID of each of the processes that matched the Process you specified.

Start/Stop Service

The Start/Stop Service activity will start, stop, pause, or restart a Windows service. The Start/Stop Service activity can be used to restart a service that has stopped responding or shut down a service in preparation for a backup. This activity uses a satellite license.

Configuring the Start/Stop Service Activity

Before you configure the Start/Stop Service activity you need to determine the following:

- The service name
- The computer where the service is running
- Parameters that are required to start the service.



Note

This depends on the service you are interacting with; it may not be required.

Use the following information to configure the Start/Stop Service activity.

Details Tab

Settings	Configuration Instructions
Action	Select one of the following actions that you want to take on the service:
	Start service: Start the service if it is stopped. This action is ignored if the service is already running.
	Stop service: Stop a running service. This action is ignored if the service is already stopped.
	Pause service: Pause a running service. This action is ignored if the service is already stopped or paused.
	Restart service: Stop then start a running service. If the service is already stopped it will only be started.
Computer	Type the computer where this service is running. Type localhost to specify the runbook server where the runbook is being processed. You can also use the ellipsis () button to browse for the computer.
Service	Type the name of the service. You can also use the ellipsis () button to browse for the service. Browsing is only available if you have specified a valid Computer .
Parameters	Type any parameters that are required to interact with the Service .
Action must complete in less than	Specify the maximum amount of time in which the action must complete. After the time has expired, the Start/Stop Service activity will timeout and return a failure.

Published Data

The following table lists the published data items.

Item	Description
Service display name	The name of the service as it appears in the Windows Services control panel utility.

Item	Description
Service real name	The name of the ran file that the service is running.
Service status	The current status of the service.
Service computer	The name of the computer where the service is located.
Control Parameters	The parameters that were passed to the service when it was started, stopped, paused or restarted.
Control Time Allowance	The maximum amount of time that was specified to complete the Start, Stop, Pause, or Restart of service action.
Control Action	The action that was taken on the service: Start, Stop, Pause, or Restart.

Restart System

The Restart System activity will restart a computer on your network. The Restart System activity can either wait for applications to shut down gracefully or you can configure the activity to forcefully shut down any running applications. You also can send a message to notify your users of the reason for the disruption.

Some applications may consume memory and hard disk space and will not relinquish them without restarting the system. The Restart System activity can be used to restart these systems during maintenance windows to maintain service during business hours.

Configuring the Restart System Activity

Before you configure the Restart System activity, you will need to determine the following:

- The computer you want to restart.
- Whether you want to forcefully shut down any running applications.

Use the following information to configure the Restart System activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the computer that you are restarting. You
	can also use the ellipsis () button to browse

Settings	Configuration Instructions
	for the computer.
Message	Type a message that will be displayed to users of the Computer before it is shut down.
Wait	Type the number of seconds after sending the Message to the users before the system will be shut down.
Force applications to close	Select to forcefully shut down any applications that are running when the system is restarted.

Published Data

The following table lists the published data items.

Item	Description
Computer	The computer that was restarted.
Message to display	The message that was sent to the computer before restarting.
Shutdown delay	The number of seconds of delay between the message being sent and the computer restart.
Force open apps to close	Determines whether open applications were forced to shut down when the computer was restarted. This value can be either True or False.

Save Event Log

The Save Event Log activity is used to save entries from an event log so that they can be used later. The Save Event Log activity saves the event log entries to a delimited text file in a format that you specify. The activity allows you to choose which fields will be saved and allows you to filter against the fields to only allow particular event log entries to be saved. This activity uses a satellite license.

The Save Event Log activity can be used to create audit trails of problems that occur with a particular application or specific categories of event log entries. These saved files can later be used to track the performance of servers and applications in your network.

Configuring the Save Event Log Activity

Before you configure the Save Event Log activity, you need to determine the following:

- The event log that you are saving from
- The computer where it is located
- The fields that you want to include
- The format of the file



If you require only specific entries to be saved and not the entire event log, you will need to know what fields to filter against as well as what values to filter.

Use the following information to configure the Save Event Log activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the computer where the event log is located. Type localhost to specify the runbook server where the runbook is being processed. You can also use the ellipsis () button to browse for the computer.
Event log	Type the name of the Windows Event Log where the entries that you are saving are located. You can also use the ellipsis () button to browse for the event log name. Browsing is only available if you have specified a valid Computer .
Include	Select all the event Log fields that you want to save to the file. You have the option to select Event ID, Source, Category, Description, Type, Computer, and Date/time.

Filters Tab

Settings	Configuration Instructions
Event ID	Select and type the specific event ID of the event log entry that you want to save.
Source	Select and type the value that the Source field of the event log entries will need to match.
Category	Select and type the value that the Category field of the event log entries will need to match.

Settings	Configuration Instructions
Description	Select and type the value that the Description field of the event log entries will need to match.
Туре	Select and specify the value that the Type field of the event log entries will need to match.
Computer	Select and specify the value that the Computer field of the event log entries will need to match.
Date from	Select and specify the ranges of dates that the events will need to be from to be included.

Output Tab

Settings	Configuration Instructions
File name	Type the name of the file where the event log entries will be saved. This file will be saved on the computer where the event log resides.
If the file exists	 Select the action that you want to take if a file with the same name already exists: Create a file with a unique name: Select to append a value to the filename to create a unique name that does not conflict with an existing name. Append: Select to append the entries that are being saved to the file. Overwrite: Select to overwrite the existing file with the file that is being created. Fail: Select to cause the Save Event Log activity to fail if the filename already exists.
File format	Select the format that will be used to save the event log entries to the file: CSV Delimited: Select to use the CSV format to write each log entry. TAB Delimited: Select to separate fields in each entry using the TAB character. Custom Delimited: Select to separate fields in each entry using a custom character that you specify in the Delimiter box.

Settings	Configuration Instructions
Delimiter	Type the delimiter that you want to use to separate the fields of each entry.
Create column headings	Select to save the column header information when saving a set of entries to a file. The header information contains meta data such as the field names.

Published Data

The following table lists the published data items.

Item	Description
Event log name	The name of the event log that was saved.
Computer	The computer where the event log that was saved resides.
Name and path of the file where entries are saved	The full path of the file where the event log was saved.
Number of Entries	The number of entries that were saved.

Query WMI

The Query WMI activity will send a WMI query to a system that you specify and return the results. This activity also can be used to check statistics on a remote server to create audit trails that can be reviewed later.

Configuring the Query WMI Activity

Before you configure the Query WMI activity, you need to determine the following:

- The computer you are querying.
- The WMI query statement you want to run.

Use the following information to configure the Query WMI activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer that you are

Settings	Configuration Instructions
	running the WMI query against. You can also use the ellipsis () button to browse for the computer.
Namespace	Type the name of the WMI namespace that you want to query.
WMI query	Type the WMI query that will be used to query the Computer . For more information about Windows Management Instrumentation, see Windows Management Instrumentation (http://go.microsoft.com/fwlink/?LinkId=221343).

Published Data

The following table lists the published data items.

Item	Description
Computer where the WMI query is performed	The name of the computer where the WMI query was ran.
WMI Query	The WMI query that was sent to the computer.
WMI Query Result as a string	The result of the WMI query.
WMI Namespace	The WMI namespace that you queried.

Run SSH Command

The Run SSH Command activity opens an SSH connection to a remote server and runs shell commands on that server. Use the Run SSH Command activity to run backup applications or a batch script that runs a set of complex commands on a non-Windows computer. The Run SSH Command activity can run any command in a Secure Shell.

Run SSH Command activity is based on PuTTY beta .61. The implementation of SSH in "Run SSH Command" has certain limitations:

- The Run SSH Command activity does not work against all SSH-1 and SSH-2 servers. In general, this activity functions with most SSH servers, but it does not work for all SSH server implementations.
- You must download and use the PuTTy key generation tool to create keys for the Run SSH
 Command activity. The key generation tool is available at <u>Download PuTTY a free SSH</u>
 and telnet client for Windows.

- The Run SSH Command activity supports SSH-1. Microsoft does not recommend the use of SSH-1. If you want to prevent The Run SSH Command activity from using SSH-1, you should use a key file that contains keys that do not support SSH-1. Do not use a username and password pair use a key file.
- The property Accept Host Key Change is not a recommended setting. This property should
 only be used to establish the initial connection to a computer when the key is stored on the
 runbook server. Runbooks that contain the Run SSH Command activity should be configured
 with Accept Host Key Change disabled. When you use this property it disables the
 validation of the identity of the SSH server and represents a security risk.
- You should review the list if cryptographic ciphers supported by PuTTY, which is found at Encryption algorithm selection.
- PuTTY beta .61 uses a pseudorandom number generator suitable for most cryptographic purposes. It is not recommended for the generation of long-term cryptographic keys.

For more information about PuTTY, go to <u>Download PuTTY - a free SSH and telnet client for</u> Windows.

Configuring the Run SSH Command Activity

Before you configure the Run SSH Command activity, you need to determine the following:

- Connection information for the computer that hosts the SSH server that you want to connect to.
- Commands that you want to run.
- Whether you require a key file to log into the server before you are able to run commands; this depends on your SSH server.

Use the following information to configure the Run SSH Command activity.

Details

Settings	Configuration Instructions
Computer	Type the name of the computer or IP address where the SSH server is running. You can also use the ellipsis () button to browse for the computer.
Port	Type the port number that you need to use to connect to the SSH server.
Run Command	Select this option and type the command that you want to run on the SSH server after the connection has been established.
Command Set File	Select this option and specify a file that contains a set of commands that will be run on the SSH server when the connection has been established. The command set file must use the

Settings	Configuration Instructions
	scripting language of the native shell on the SSH server.
Accept Host Key Change	Select this option to accept host key changes when they occur.
	It is recommended that you do not use this setting because it can cause a runbook to accept any change in a server, including any that are for malicious purposes. By selecting this option, you are instructing the activity to connect to any server, regardless of the host key. Only use this option for testing purposes.
Connection Timeout	Specify the amount of time, in seconds, that the Run SSH Command activity will wait for the SSH command to complete. Configure a value of 0 (zero), or leave the box blank, to wait indefinitely.
	After the timeout period has elapsed, the Run SSH Command activity times out and returns a warning. The command that you ran may continue running, regardless of whether the Run SSH Command activity times out.

Advanced

Settings	Configuration Instructions
Username	Type the username that you need to log into the SSH server.
Password	Select this option and type the password that is associated with the Username that you specified.
Key File	Select this option to specify a key file to use. You must use the PuTTY key file generator to create a key file. You can download this tool from Download PuTTY - a free SSH and telnet client for Windows .

Settings	Configuration Instructions
· ·	Type the passphrase that is associated with the key file that you specified.

The following table lists the published data items.

Item	Description
Command	The command that ran on the SSH server. This data is not available when the Command Set File option is selected.
Command Set file	The command set file that was used to run commands on the SSH server. This option is not available when the Run Command option is selected.
Computer name	The name or IP address of the SSH server.
Execution Result	The text that was published as output from the commands that were run on the SSH server.
Exit Code	The exit code published by the command. When using a command set file, this will be the exit code of the last command in the file.
Key file path	The path of the key file that was used to authenticate with the SSH server.
Port	The port used to connect to the SSH server.
Username	The username used to log into the SSH server.

Get SNMP Variable

The Get SNMP Variable activity will query a network device for the value of variable that is assigned to the Management Information Base address that you specify. You can use the Get SNMP Variable activity to retrieve information about a network device to determine if an administrator needs to be notified.

Configuring the Get SNMP Variable Activity

Before you configure the Get SNMP Variable activity, you need to determine the following:

- The IP address of the device, as well as the port number, SNMP MIB, and SNMP version
- The community string required to retrieve the variable.



You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

Use the following information to configure the Get SNMP Variable activity.

Details Tab

Settings	Configuration Instructions
IP address	Type the IP address of the device hosting the MIB variable.
Port	Type port used to communicate with the network device. The default port is 161.
Object identifier	Type the MIB identifier of the variable whose value you want to retrieve.
SNMP Version	Select the SNMP version to use when connecting to the network device.
Community string	Type the community string that will be used to authenticate against the network device. The community should have rights of Read only or higher. This field is case-sensitive and supports only alphanumeric characters.

Advanced Tab

Settings	Configuration Instructions
Timeout	Type the number of seconds the Get SNMP Variable will wait for a response from the network device. If the operation times out, then it will attempt to retry the action. The number of retries is specified in the Retry box.
Retry	Type the number of times to attempt to retrieve

Settings	Configuration Instructions
	the SNMP variable

The following table lists the published data items.

Item	Description
MIB identifier	The MIB identifier of the variable that was retrieved.
MIB value	The value of the variable that was retrieved.
Device IP address	The IP address of the device where the variable was retrieved.
Timeout	The timeout period specified in the Get SNMP variable operator interface.
Retry attempts	The number of attempts made to retrieve the SNMP Variable.
SNMP Version	The SNMP version that was specified to retrieve this variable. This value can be SNMPv1 or SNMPv2c.
Community string	The community string that was used to authenticate against this SNMP variable.
Request port	The port used to communicate to the SNMP device.

Monitor SNMP Trap

The Monitor SNMP Trap activity waits for an event to occur either in the Microsoft SNMP Trap Service, or on a port that you specify. Using filters, you can invoke your runbooks according to the device that raised the event or the enterprise, generic, or specific identifiers of the SNMP trap. Use the Monitor SNMP Trap activity to monitor a network device for critical errors, automatically create a trouble ticket, and perform level 1 diagnostics on the device.

Configuring the Monitor SNMP Trap Activity

Before you configure the Monitor SNMP Trap activity, you need to determine the following:

- Version of SNMP that you are using
- Source host IP address
- Enterprise identifier of the device
- Generic or specific identifier of the device that you are monitoring

Use the following information to configure the Monitor SNMP Trap activity.

Details Tab

Settings	Configuration Instructions
Microsoft SNMP Trap Service (SNMPv1, SNMPv2c)	Select this option to use the Microsoft SNMP Trap Service. This service is only compatible with SNMP versions SNMPv1 and SNMPv2c.
No dependency (SNMPv1, SNMPv2c, SNMPv3)	Select this option to monitor SNMP traps using a port rather than the Microsoft SNMP Trap Service.
Port	If you select the No dependency option, type the communication port number that will be monitored for SNMP traps. If you select port 162 , the Microsoft SNMP Trap Service must be disabled because it uses the same port when it runs.
Source host	Select to specify the IP address of the device where the event originates.
Enterprise identifier	Select to specify the enterprise identifier of the event raised by the device.
Generic identifier	Select to specify the generic identifier of the SNMP trap. There are six options available:
	coldStart(0): Select to filter for a cold start of the network device. This option has a numerical value of 0.
	warmStart(1):Select to filter for a warm start of the network device. This option has a numerical value of 1.
	linkDown(2): Select to filter for a severed connection to the network device. This option has a numerical value of 2.
	linkUp(3): Select to filter for a re-established connection to the network device. This option has a numerical value of 3.

Settings	Configuration Instructions
	authenticationFailure(4): Select to filter for a failed SNMP authentications to the network device. This option has a numerical value of 4.
	egpNeighborLoss(5): Select to filter for a lost connection to an EGP neighbor. This option has a numerical value of 5.
	enterpriseSpecific(6): Select to filter based on an enterprise specific ID. This option has a numerical value of 6. You must specify this option to filter based on a Specific identifier.
Specific identifier	Select to specify an enterprise specific identifier for the SNMP trap. This setting becomes active when you select the enterpriseSpecific(6) option in the Generic identifier box.

The following table lists published data items.

Item	Description
Source IP address	The IP address of the device where the trap originated.
Enterprise Id	The enterprise ID of the trap.
Generic Id	The generic ID of the trap.
Specific Id	The specific ID of the trap. The value of the specific identifier is published when using the enterpriseSpecific(6) option of the Generic identifier box. Otherwise, a value of 0 (zero) is published.
Trap port	The port where the trap was received.
Varbind count	The number of variable bindings received.
SNMP Version	The SNMP version specified for this trap.

Send SNMP Trap

The Send SNMP Trap activity will raise an SNMP event that can be detected by a network systems manager application. By using an enterprise identifier of a known network device, you can send SNMP Traps on behalf of a network device in your system. Use the Send SNMP Trap to create events for runbooks that need to be tracked using an SNMP monitoring product.

Configuring the Send SNMP Trap Activity

Before you configure the Send SNMP Trap activity you need to determine the following:

- IP address of the device where you will send your SNMP trap
- Identifiers of the trap
- The SNMP version you will use
- The agent address you want to identify as the sender of the SNMP trap information.



You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

Use the following information to configure the Send SNMP Trap activity.

You can also add more information to the SNMP trap. Each item that you add becomes a published data item.

Details Tab

Settings	Configuration Instructions
IP address	Type the name of the computer or IP address where you are sending the SNMP trap.
Port	Type the port to use to send the SNMP trap.
Enterprise identifier	Specify the enterprise identifier of the event being raised by the Send SNMP Trap activity.
Generic identifier	Specify the generic identifier of the SNMP trap. There are six options available:
	coldStart(0): Select to signify a cold start of the network device. This option has a numerical value of 0.
	warmStart(1): Select to signify a warm start of the network device. This option has a numerical value of 1.

Settings	Configuration Instructions
	linkDown(2): Select to signify a severed connection to the network device. This option has a numerical value of 2.
	linkUp(3): Select to signify a re-established connection to the network device. This option has a numerical value of 3.
	authenticationFailure(4): Select to signify a failed SNMP authentications to the network device. This option has a numerical value of 4.
	egpNeighborLoss(5): Select to signify a lost EGP peer connection to the network device. This option has a numerical value of 5.
	enterpriseSpecific(6): Select to specify an enterprise specific id. This option has a numerical value of 6. You must specify this option to specify a specific identifier.
Specific identifier	Type the enterprise specific identifier for the SNMP trap. This setting becomes active when you select the enterpriseSpecific(6) option of the Generic identifier box.
SNMP Version	Select the SNMP version to use when generating the SNMP trap.
Community string	Type the community string that will be used to authenticate against the network device. This field is case-sensitive and supports only alphanumeric characters.
	The Send SNMP Trap activity does not verify the content of community strings, nor whether the strings are received. It sends whatever data you provide, whether it is valid or not. The activity returns a status of Success if it was able to send the data, regardless of whether the data were correct or readable.

Advanced Tab

Settings	Configuration Instructions
Address	If you want to identify another computer as the

Settings	Configuration Instructions
	agent that sends the SNMP trap information, type the agent address in the box. Otherwise, leave the box blank. The activity will use the agent address of the runbook server that runs the runbook. This setting can only be used with version SNMPv1.

The following table lists the published data items.

Item	Description
Destination IP address	The IP address of the device where the trap is sent.
Enterprise Id	The enterprise ID of the trap.
Generic Id	The generic ID of the trap.
Trap port	The port where the trap was sent.
SNMP Version	The SNMP version that was specified for this trap. This value can be SNMPv1 or SNMPv2c.
Community string	The community string that will be needed to retrieve this SNMP trap.
Origin address	The address of the device that generated the trap.
Specific Id	The specific ID of the trap.

Set SNMP Variable

The Set SNMP Variable activity will modify a variable, specified by its MIB, on a network device. Use the Set SNMP Variable to update a variable that reports on the failure or success of a critical runbook.

Configuring the Set SNMP Variable Activity

Before you configure the Set SNMP Variable activity you need to determine the following:

• IP address of the device as well as the port number, SNMP MIB, and the SNMP version

• Community string required to update the variable



You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

Use the following information to configure the Set SNMP Variable activity.

Details Tab

Settings	Configuration Instructions
IP address	Type the IP address of the device hosting the MIB variable.
Port	Type port used to communicate with the network device.
Object identifier	Type the MIB identifier of the variable whose value you want to change.
Object value	Type the new value of the variable you are changing. Make sure that the new value matches the constraints that are set out by device manufacturer. This field is casesensitive and supports only alphanumeric characters.
SNMP version	Select the SNMP version to use when connecting to the network device. You can select SNMPv1 or SNMPv2c.
Community string	Type the community string that will be used to authenticate against the network device. The community should have rights of Read write or higher. This field is case-sensitive and supports only alphanumeric characters.

Advanced Tab

Settings	Configuration Instructions
Timeout	Type the number of seconds the Set SNMP Variable will wait for a response from the network device. If the operation times out, then
	it will attempt to retry the action. The number of

Settings	Configuration Instructions
	retries is specified in the Retry box.
Retry	Type the number of times to attempt to set the SNMP variable.

The following table lists the published data items.

Item	Description
MIB identifier	The MIB identifier of the variable that was set.
MIB value	The new value of the variable that was set.
Device IP address	The IP address of the device where the variable was set.
Timeout	The timeout period specified in the Set SNMP variable operator interface.
Retry attempts	The number of attempts made to set the SNMP variable.
SNMP Version	The SNMP version that was specified to set this variable. This value can be SNMPv1, or SNMPv2c.
Community string	The community string that was used to authenticate against this SNMP variable.
Request port	The port used to communicate to the SNMP device.

Scheduling

The following table provides a brief description of tasks you can accomplish when using each Scheduling activity.

Tasks	Scheduling Activities
Invoke a runbook at a scheduled time.	Monitor Date/Time
Verify that a runbook can run at its scheduled	Check Schedule

Tasks	Scheduling Activities
time.	

Monitor Date/Time

The Monitor Date/Time activity invokes runbooks at a time or interval that you specify. Use the Monitor Date/Time activity to invoke your runbooks at a specific time once a day, week, or month. You can also schedule runbooks to be invoked when a specific number of seconds have passed since it was last invoked, or immediately after the runbook is deployed.

The Monitor Date/Time activity uses the system clock of the operating system on the computer that runs the runbook server, not Coordinated Universal Time (UTC), to verify the runbook's launch time. This enables the Monitor Date/Time activity to function in virtual machine environments, and to continue running even when the system clock is adjusted because of the move into or out of Daylight Saving Time. However, if a runbook is scheduled to start during an hour that is skipped when the system clock is adjusted forward by one hour, that starting time is skipped, and the runbook starts at the next scheduled time. If a runbook is scheduled to start during an hour that occurs twice because the system clock is adjusted backwards by one hour, the runbook launches twice.

Depending on the practices in your time zone, the usual official time to change the system clocks at the start or finish of Daylight Saving Time is 2:00 A.M., or 02:00. We recommend that you configure a schedule to prevent your runbooks from being skipped or processed twice when the system clock changes.

The Monitor Date/Time activity becomes inactive when the schedule does not allow the runbook to run.

The Monitor Date/Time activity is best suited for scenarios where you need to run routines regularly that do not rely on events in other systems. For example, nightly backup procedures or periodically reading and processing mail in a customer service inbox.

Additional Use Cases

The Monitor Date/Time activity starts according to its configured interval and passes the runbook run to the Check Schedule activity. The Check Schedule activity verifies that the runbook is allowed to run at the current time.

If the runbook is permitted to run at that time, the Check Schedule activity publishes a published data value of True. It passes the runbook run to the next activity if there is a link to the next activity with a invoke condition of "Conforms to schedule from Check Schedule equals true". If the runbook is not permitted to run at that time, the Check Schedule activity publishes a published

data value of False. It passes the runbook run to the next activity if there is a link to the next activity with an invoke condition of "Conforms to schedule from Check Schedule equals false." This is useful when you want to implement conditional link branches according to the results of the Schedule verification.

Configuring the Monitor Date/Time Activity

Before you configure the Monitor Date/Time activity, you need to determine the time or interval you want to use to invoke the runbook.

Use the following the information to configure the Monitor Date/Time activity.

Details Tab

Settings	Configuration Instructions
At	Select an absolute time for the runbook to run. The Monitor Date/Time activity will invoke every day at the time that you specify.
Every [x] days [y] hours [z] minutes	Select to specify intervals of days, hours, and minutes for the runbook to run.
	Starting: Select to specify the number of minutes past the hour to invoke the runbook. This option is only available if you have specified 0 minutes and at least 1 Day or 1 hour.
	At time slices within the hour: Select to invoke the runbook at times that are multiples of minutes you have specified. This option is only available when 0 days and 0 hours are specified. For example, if minutes is set to 15 then the Monitor Date/Time activity will invoke at 0, 15, 30, 45 minutes past each hour.
	Trigger immediately: Select to invoke the runbook immediately after deploying.
Every [x] seconds	Select to specify the interval, in seconds, between each time the runbook is ran.

The Monitor Date/Time activity accepts the following inputs when configuring times and intervals:

Configuring Time and Intervals

Unit	Accepted Input
Seconds	5 - 300

Unit	Accepted Input
Minutes	0 - 59 (0 is allowed only when hours/days are also specified)
Hours	0 - 23 (0 is allowed when days/minutes are also specified)
Days	0 - 48 (0 is allowed when hours/minutes are also specified)
Time slices	1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30

This activity does not generate published data items.

Check Schedule

The Check schedule activity verifies that a runbook is allowed to run at the current time according to the permitted times or interval configured in a schedule. To use this activity, you can create a schedule and configure the permitted times, denied times, or interval at which the runbook can run. Then you can insert the activity into a runbook following a Monitor Date/Time activity and configure it to check the schedule to verify whether a runbook is allowed to run at the current time. You can also use the Check Schedule activity in a runbook that monitors systems for availability. If a problem is encountered, the Check Schedule activity can verify whether the current time is during business hours, or in or out of a maintenance window.

Configuring the Check Schedule Activity

Use the following information to configure the Check Schedule activity.

To configure the Check Schedule activity

- 1. From the **Activity pane**, drag a **Check Schedule** activity to the runbook.
- 2. Double-click the Check Schedule activity icon to open the Properties dialog box.
- Select the **Details** tab, and next to the **Schedule Template** box, click the ellipsis (...)
 button and in the **Select a Schedule** dialog box, select the **Schedule** that you want to
 verify.

Published Data

The following table lists published data items.

Item	Description
Conforms to schedule	Determines whether the current time is within the schedule specified. This value can be either True or False.

Monitoring

The following table provides a brief description of tasks you can accomplish when using each Monitoring activity.

Tasks	Monitoring Activities
Invoke a runbook when new events that match a filter appear in the Windows Event Log.	Monitor Event Log
Invoke a runbook when a service has been started or stopped.	Monitor Service
Check the status of a service on any computer.	Get Service Status
Invoke a runbook when a process has been started or stopped.	Monitor Process
Check the status of a running process on any computer.	Get Process Status
Send a ping to a remote computer or IP address and wait for a response.	Monitor Computer/IP
Send a ping to a remote computer or IP address and wait for a response.	Get Computer/IP Status
Invoke a runbook when the disk space on a computer passes a critical threshold.	Monitor Disk Space
Retrieve the current amount of available disk space.	Get Disk Space Status
Invoke a runbook when an internet application server becomes available or unavailable.	Monitor Internet Application
Check the availability of a Web, Email (POP3 or SMTP), FTP, DNS, or custom server.	Get Internet Application Status
Invoke a runbook when a Windows Management Instrumentation (WMI) event is	Monitor WMI

Tasks	Monitoring Activities
received as a result of the WMI event query you specified.	

Monitor Event Log

The Monitor Event Log activity invokes runbooks when new events that match a filter that you specify appear in the Windows Event Log. You can use the Monitor Event Log activity to run runbooks that will escalate, investigate, or correct any issues in response to events being generated to the Windows Event Log. For example, a security audit failure appears in the security log which will send an email to an administrator to notify them of the problem. The second mode invokes your runbook when the size of the Windows Event Log reaches the maximum size allowed.

Configuring the Monitor Event Log Activity

Before you configure the Monitor Event Log activity, you need to determine the following:

- Name of the event log you are monitoring
- Details about the events that will invoke the runbook

Use the following steps to configure the Monitor Event Log activity.

To configure the Monitor Event Log activity

- 1. From the **Activity pane**, drag a **Monitor Event Log** activity to the runbook.
- 2. Double-click the **Monitor Event Log** activity icon to open the **Properties** dialog box.
- 3. Configure the settings on the **Details** tab and on the **Advanced** tab. Configuration instructions are listed in the following tables.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer that stores the Windows Event Log that you want to monitor. You can also browse for the computer using the ellipsis () button. The runbook server that
	runs this activity must have the appropriate rights to monitor the Windows Event Log on

Settings	Configuration Instructions
	that computer.
Event log	Type the name of the Windows Event Log that you are monitoring. You can also browse for the Windows Event Log using the ellipsis () button. Windows includes three Event Logs by default: Application, Security, and System. The computer that you are connecting to may contain other Event Logs.
Message filters	The list shows all the filters that have been configured to filter the events that are generated in the log that you have specified. To edit or remove an item in the list, select it and click Edit or Remove as applicable.
	►To add an event filter
	 Click Add to open the Filter Properties dialog box. Select the property of the event log entry that you are filtering against. You can filter against the Category, Description, Event ID, Source, and Type that is attributed to the event. Specify the relation you are using to compare the value of the event property to the filter value. If you select Category, Description, Type, and Source you can specify Contains or Does not contain. For Event ID you can specify is different than, is equal to , is lower than, is lower than or
	equals, is more than, and is more than or equals. 4. Specify the filter value that you are comparing the event property against. For Category, Description, and Source, enter the string that is contained within the property. For Event ID, enter the numeric value that will be compared against the ID of the event. For the Type condition, select the specific type of event that you want

Settings	Configuration Instructions
	to filter for such as Error, Warning, Information, Success Audit, or Failure Audit.

The following table lists the published data items.

Item	Description
Event log name	The name of the Windows Event Log being monitored.
Computer	The name of the computer where the Windows Event Log is stored.
Log entry description	The text that is contained in the Event Log entry description.
Log Entry ID	The ID of the Event Log entry.
Log Entry source	The source of the event.
Log Entry computer	The computer where the event occurred.
Log Entry type	The type of event.
Log Entry date	The date the event was logged.
Log Entry time	The time the event was logged.

Monitor Service

The Monitor Service activity invokes runbooks when a service has been started or stopped. You can use the Monitor Service activity to monitor services on any remote computer. Use the Monitor Service activity to create runbooks that take corrective actions when services unintentionally shut down. For example, if a SQL Server service that hosts critical data stops responding, you can use a Monitor Service activity with a Start/Stop Service activity to automatically restart the service.

Configuring the Monitor Service Activity

Before you configure the Monitor Service activity, you need to determine the following:

Which computer hosts the service that you are monitoring

- Which service you want to monitor
- Whether the runbook will run when the service is started or stopped Use the following information to configure the Monitor Service activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the service that you are monitoring is located. You can also browse for the computer using the ellipsis () button. The runbook server that runs this activity must have the appropriate rights to monitor the services on that computer.
Service	Type the name of the service that you are monitoring. To open the Choose a Service dialog box, click the ellipsis () button.
Service is started	Select to invoke the Monitor Service activity when the selected service has been started.
Service is stopped or paused	Select to invoke the Monitor Service activity when the selected service has been stopped or paused. When a service is restarted using the Windows Service Control Manager it is stopped and then started in succession. This will cause the Monitor Service activity to be invoked regardless of whether you have specified to invoke when the Service is started or Service is stopped or paused.
Restart stopped service	Select the Restart stopped service box to restart a service that has stopped. You can also use the Start/Stop Service activity instead of selecting this option.
Test frequency	Select the amount of time to wait between each time that the Monitor Service activity checks the status of the service.

Published Data

The following table lists the published data items.

Item	Description
Service display name	The name of the service as it appears in the Windows Services control panel utility.
Service real name	The name of the file that the service is running.
Service status	The current status of the service.
Service computer	The name of the computer where the service is located.
Test interval	The number of seconds between each check of the service status.
Restart stopped service	Determines whether the service is automatically restarted when it is found to be stopped. This value can be either True or False.

Get Service Status

The Get Service Status activity will check the status of a service on any computer. Use the Get Service Status to check the status of service before performing another action. For example, if you have an SQL Server backup runbook that requires that SQL Server is stopped before performing the backup, you can check the status and then stop the service using the Start/Stop Service activity.

Configuring the Get Service Status Activity

Before you configure the Get Service Status activity, you need to determine the following:

- The computer where the service is located
- The name of the service

Use the following information to configure the Get Service Status activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the service that you are checking is located. You can also use the ellipsis () button to browse for the computer. The runbook server that runs this runbook must have the appropriate rights

Settings	Configuration Instructions
	to monitor the services on that computer.
Service	Type the name of the service that you are checking. You can also browse for the service using the ellipsis () button.

The following table lists the published data items.

Item	Description
Service display name	The name of the service as it appears in the Windows Services control panel utility.
Service real name	The name of the ran file that the service is running.
Service status	The current status of the service.
Service computer	The name of the computer where the service is located.

Monitor Process

The Monitor Process activity invokes runbooks when a process has been started or stopped. A process is any executable file that is running. You can use the Monitor Process activity to monitor processes on any remote computer.

The Monitor Process activity can be used to create runbooks that take corrective actions when a process has been started but has not stopped. For example, if an application that has a tendency to stop responding and remain resident in memory even though it has completed, it can be shut down automatically by using a Monitor Process activity in a runbook with a Get Process Status activity to retrieve the status of the process and an End Process activity to shut it down.

Configuring the Monitor Process Activity

Before you configure the Monitor Process activity, you will need to determine the following:

- Which computer will run the process that you are monitoring
- Which process you want to monitor
- Whether the runbook will be ran when the process is started or stopped

Use the following information to configure the Monitor Process activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the process that you are monitoring is located. You can also browse for the computer using the ellipsis () button. The runbook server that runs this runbook must have the appropriate rights to monitor the process on that computer.
Process	Type the name of the process that you are monitoring. You can also browse for the process using the ellipsis () button.
Process is started	Select to invoke the Monitor Process activity when the selected process has been started.
Process is stopped	Select to invoke the Monitor Process activity when the last running instance of the selected process has been stopped.
Test frequency	Select the amount of time to wait between each time that the Monitor Process activity checks the status of the process.

Published Data

The following table lists the published data items.

Item	Description
Computer	The name of the computer where the process is located.
Process name	The name of the process ran.
Number of instances for the process	The number of running occurrences of the process.
Test interval	The number of seconds between each check of the process status.
Invokes on process start	Determines whether the runbook will be invoked if the process is started.
Invokes on process end	Determines whether the runbook will be

Item	Description
	invoked if the process is stopped.

Get Process Status

The Get Process Status activity checks the status of a running process on any computer. Use the Get Process Status activity to check the status of a process before performing another action. For example, you can check that a process that was detected by the <u>Monitor Process</u> activity is still running before shutting it down with the <u>End Process</u> activity.



Important

The Get Process Status activity returns a status of **failed** if the named process is not running. If the activity returns **failed**, the overall status of the runbook is set to **warning** or **failed**, depending on the number of activities in the runbook.

Configuring the Get Process Status Activity

Before you configure the Get Process Status activity, you need to determine the following:

- The computer where the process is located.
- The file name that will run the process.

Use the following information to configure the Get Process Status activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the process that you are checking is located. You can also browse for the computer using the ellipsis () button. The runbook server that runs this runbook must have the appropriate rights to check the process on that computer.
Process	Type the name of the process that you are checking. You can also browse for the process using the ellipsis () button.

Published Data

The following table lists the published data items.

Item	Description
Computer	The name of the computer where the process is located.
Process name	The name of the process ran.
Process ID	The ID of the process.
Number of instances for the process	The number of running occurrences of the process.

Monitor Computer/IP

The Monitor Computer/IP activity will send a ping to a remote computer or IP address and wait for a response. You can configure the Monitor Computer/IP activity to invoke your runbook if the computer is either reachable or unreachable. The Monitor Computer/IP activity can be used to invoke runbooks that will automatically notify administrators when a vital system has become unreachable on the network.

Configuring the Monitor Computer/IP Activity

Before you configure the Monitor Computer/IP activity, you will need to determine the following:

- The computer you are monitoring.
- Whether you are waiting for the computer to become reachable or waiting for it become not reachable.



Important

You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

Use the following information to configure the Monitor Computer/IP activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer that you are monitoring. You can also browse for the computer using the ellipsis () button.
The computer is not reachable	Select to invoke the Monitor Computer/IP activity when the computer that you are

Settings	Configuration Instructions
	monitoring cannot be reached using a ping.
The computer is reachable	Select to run the Monitor Computer/IP activity when the computer that you are monitoring can be reached using a ping.
Test frequency	Specify the amount of time between each ping to the Computer .

The following table lists the published data items.

Item	Description
Computer to ping	The computer that is being monitored.
Percentage of packets received	The percentage of packets that were received back from the ping.

Get Computer/IP Status

The Get Computer/IP Status activity will send a ping to a remote computer or IP address and wait for a response. If a response is received, then the Get Computer/IP Status activity will succeed. If a response is not received, the activity will fail.

The Get Computer/IP Status activity can be used to confirm that a computer is available before performing an action on that computer. You can also use the Get Computer/IP Status activity to check the availability of a computer as part of the level 1 diagnostic step when performing problem management processes.

Configuring the Get Computer/IP Status Activity

Before you configure the Get Computer/IP Status activity, you need to determine the computer name or IP address of the computer that you are monitoring.



Important

You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

Use the following information to configure the Get Computer/IP Status activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer that you are checking. You can also use the ellipsis () button to browse for the computer.

Published Data

The following table lists the published data items.

Item	Description
Computer to ping	The computer that is being monitored.
Percentage of packets received	The percentage of packets that were received back from the ping.

Monitor Disk Space

The Monitor Disk Space activity will invoke a runbook when the disk space on a computer passes a critical threshold. You can monitor multiple drives on different computers with a single Monitor Disk Space activity. The Monitor Disk Space activity can be used to invoke runbooks that will automatically backup and purge files on a hard drive that is running out of space

Configuring the Monitor Disk Space Activity

Before you configure the Monitor Disk Space activity, you need to determine the following:

- The drives that you want to monitor
- The computer where those drives are located

The runbook server that runs this runbook must have the appropriate rights to check the process on the computer that you are monitoring.

Use the following information to configure the Monitor Disk Space activity.

Test frequency example: Monitor Disk Space activity is set to test every 30 seconds

Time	All Disks are Passed Threshold?	Result
30s	No	Do not trigger runbook
60s	Yes	Trigger runbook
90s	Yes	Do not trigger runbook

Time	All Disks are Passed Threshold?	Result
120s	No	Do not trigger runbook
150s	Yes	Trigger runbook

The following table lists the published data items.

Item	Description
Computer	The name of the computer where the drive is being monitored.
Drive	The drive that is being monitored.
Percentage of Space available	The percentage of the entire drive capacity that is available.
MB available	The number of megabytes available on the drive.
GB available	The number of gigabytes available on the drive.
Test interval	The number of seconds between each test of the disk space.

Get Disk Space Status

The Get Disk Space Status activity will retrieve the current amount of available disk space on a UNC path or local disk drive that you specify. This activity can be used to check the space of a destination folder before transferring files to that location.

Configuring the Get Disk Space Status Activity

Before you configure the Get Disk Space Status activity, you need to determine the UNC path or local drive that you want to check.

Use the following information to configure the Get Disk Space Status activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer that you are

Settings	Configuration Instructions
	checking. You can also use the ellipsis () button to browse for the computer.
Drive	Type the drive path you want to check. To specify a local drive path include the colon and backslash. For example, to specify the Local Disk (C:), type "C:\". If you specify a local drive path, the runbook server that runs the runbook will check its local drive. The runbook server that runs this runbook must have the appropriate rights to check the process on the computer on which you are checking the disk space status.

The following table lists the published data items.

Item	Description
Drive	The drive that is being monitored.
Percentage of Space available	The percentage of the entire drive capacity that is available.
MB available	The number of megabytes available on the drive.
GB available	The number of gigabytes available on the drive.

Monitor Internet Application

The Monitor Internet Application activity will invoke a runbook when an internet application server becomes unavailable or becomes available. You can monitor a Web, Email (POP3 or SMTP), FTP, or DNS server. You can also configure your external FTP or Web servers to be reachable through the internet and then automatically restart the server if it is found to be unavailable.

Configuring the Monitor Internet Application Activity

Use the following information to configure the Monitor Internet Application activity.

Note

You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

General Tab

Settings	Configuration Instructions
Name	Type a descriptive name for the activity.
Description	Type a detailed description of the actions of the activity.
Туре	Select the Type that matches the server that you want to monitor. The options include the following: • Web (HTTP) • E-mail (SMTP) • E-mail (POP3) • FTP • DNS Configuration instructions for each Details tab Type are listed in the following tables.

Web (HTTP) Details Tab

Settings	Configuration Instructions
URL	Type the URL that will be used to contact the web server.
Port	Select to specify a port to use to connect to the web server. The default port is 80.
Timeout	Type the number of seconds to wait for a response from the web server. If the timeout expires without a response, the server will be considered unavailable.
Test frequency	Specify the amount of time to wait between each connection test to the server.
Check that the page contains this string	Select and type a string to search for when the page is retrieved from the web server. When this option is selected, the server is only

Settings	Configuration Instructions
	considered available if the string can be found on the page that is specified by the URL .
Search is case sensitive	Select to make the string search case sensitive.

Email (SMTP) Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the SMTP server is located. You can also browse for the computer using the ellipsis () button.
Port	Select to specify a port to use to connect to the SMTP server. The default port is 25.
Timeout	Type the number of seconds to wait for a response from the server. If the timeout expires without a response, the server will be considered unavailable.
Test frequency	Specify the amount of time to wait between each connection test to the server.
Send test email	Select to send a test email using the SMTP server. When this option is selected, the server is only considered available if the email can be sent to the server.
То	Type the address to send the email to.
From	Type the address that the email is being sent from.

Email (POP3) Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the POP3 server is located. You can also browse for the computer using the ellipsis () button.
Port	Select to specify a port to use to connect to the POP3 server. The default port is 110.
Timeout	Type the number of seconds to wait for a response from the server. If the timeout expires

Settings	Configuration Instructions
	without a response, the server will be considered unavailable.
Test frequency	Specify the amount of time to wait between each connection test to the server.
Test connection	Select to use a username and password to test the connection to the POP3 server. When this option is selected, the server is only considered available if the credentials are successfully used to log into the server.
Username	Type the username to use to log into the POP3 server.
Password	Type the password that is associated with the Username that you have specified.

FTP Details Tab

Settings	Configuration Instructions
Computer	Enter the name of the computer where the FTP server is located. You can also browse for the computer using the ellipsis () button.
Port	Select to specify a port to use to connect to the FTP server. The default port is 21.
Timeout	Type the number of seconds to wait for a response from the server. If the timeout expires without a response, the server will be considered unavailable.
Test frequency	Specify the amount of time to wait between each connection test to the server.
Test connection	Select to use a username and password to test the connection to the FTP server. When this option is selected, the server is only considered available if the credentials are successfully used to log into the server.
Username	Type the username to use to log into the FTP server.
Password	Type the password that is associated with the

Settings	Configuration Instructions
	Username that you have specified.

DNS Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the DNS server is located. You can also browse for the computer using the ellipsis () button. This field is not required to test the availability of a DNS server.
Port	Select to use the default port of 53 to connect to the DNS server.
Port	Select to specify the port to use to connect to the DNS server.
Test DNS table IP Address	Select to specify a computer name and the IP address that should be associated with that IP address. When this option is selected, the server is only considered available if the IP address is assigned to the computer that you specify.
Test frequency	Specify the amount of time to wait between each connection test to the server.

Advanced Tab

Settings	Configuration Instructions
Trigger if test succeeds	Select to invoke the Monitor Internet Application activity when the server that you are checking becomes available.
Trigger if test fails	Select to invoke the Monitor Internet Application activity when the server that you are checking becomes unavailable.

Published Data

The following table lists the published data items.

Item	Description
Computer	The name of the computer where the Internet application resides.
Port	The port used to communicate with the Internet application.
Protocol	The protocol of the Internet application. For example, HTTP or FTP.
Server Greeting	The greeting message received from the Internet application.
Web page	The HTML of the web page that was retrieved when in Web (HTTP) mode.

Get Internet Application Status

The Get Internet Application Status activity checks the availability of an internet application server. You can check the availability of a Web (HTTP), Email (SMTP), Email (POP3), FTP, DNS, or custom server. You can also configure a server so it is available after a power outage or a restart.

Configuring the Get Internet Application Status Activity

Use the following information to configure the Get Internet Application Status activity.



Note

You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

General Tab

Settings	Configuration Instructions
Name	Type a descriptive name for the activity.
Description	Type a detailed description of the actions of the activity.
Туре	Select the Type that matches the server that

Settings	Configuration Instructions
	you want to monitor. The options include the
	following:
	Web (HTTP)
	E-mail (SMTP)
	E-mail (POP3)
	• FTP
	• DNS
	Custom
	Configuration instructions for each Details tab Type are listed in the following tables.

Web (HTTP) Details Tab

Settings	Configuration Instructions
URL	Type the URL that will be used to contact the web server.
Port	Select to specify a port to use to connect to the web server. The default port is 80.
Timeout	Type the number of seconds to wait for a response from the web server. If the timeout expires without a response, the server will be considered unavailable.
Check that the page contains this string	Select and type a string to search for when the page is retrieved from the web server. When this option is selected, the server is only considered available if the string can be found on the page that is specified by the URL .
Search is case sensitive	Select to make the string search case sensitive.

Email (SMTP) Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the SMTP server is located. You can also browse for the computer using the ellipsis () button.
Port	Select to specify a port to use to connect to the SMTP server. The default port is 25.

Settings	Configuration Instructions
Timeout	Type the number of seconds to wait for a response from the server. If the timeout expires without a response, the server will be considered unavailable.
Send test email	Select to send a test email using the SMTP server. When this option is selected, the server is only considered available if the email can be sent to the server.
То	Type the address to send the email to.
From	Type the address that the email is being sent from.

Email (POP3) Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the POP3 server is located. You can also browse for the computer using the ellipsis () button.
Port	Select to specify a port to use to connect to the POP3 server. The default port is 110.
Timeout	Type the number of seconds to wait for a response from the server. If the timeout expires without a response, the server will be considered unavailable.
Test connection	Select to use a username and password to test the connection to the POP3 server. When this option is selected, the server is only considered available if the credentials are successfully used to log into the server.
Username	Type the username to use to log into the POP3 server.
Password	Type the password that is associated with the Username that you have specified.

FTP Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the FTP server is located. You can also browse for the computer using the ellipsis () button.
Port	Select to specify a port to use to connect to the FTP server. The default port is 21.
Timeout	Type the number of seconds to wait for a response from the server. If the timeout expires without a response, the server will be considered unavailable.
Test connection	Select to use a username and password to test the connection to the FTP server. When this option is selected, the server is only considered available if the credentials are successfully used to log into the server.
Username	Type the username to use to log into the FTP server.
Password	Type the password that is associated with the Username that you have specified.

DNS Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer where the DNS server is located. You can also browse for the computer using the ellipsis () button. This field is not required to test the availability of a DNS server.
Port	Use the default port of 53 to connect to the DNS server.
Port	Select to specify the port to use to connect to the DNS server.
Test DNS table IP address	Select to specify a computer name and the IP address that should be associated with that IP address. When this option is selected, the server is only considered available if the IP address is assigned to the computer that you

Settings	Configuration Instructions
	specify.

Custom Details Tab

Settings	Configuration Instructions
Actions	Click Add or Insert to open the Action Properties dialog box. Configure the rest of the settings described in this table. Tip Click the Up or Down buttons to change the order of the actions. Click Remove to remove an action. Click Edit to edit an action.
Open port	Type the port number and the computer where the Internet application resides.
Send data	Type the data that you will send to the Internet application. To specify a file that contains the data you want to send, click Send data from file .
Receive data	Click Publish as execution data and click the name of the variable where the received data will be saved. Click Save data, specify the File where you want to save the data received from the Internet application. Click the action you want to specify in the If the Destination File Exists box. You can select Create a file with a unique name, Append data to the existing file, or Overwrite the existing file.
Close port	You must configure the Open port action before you can select this action.

You can use a sequence of actions to test a custom Internet application that is not part of the predefined list. You can perform actions such as opening and closing a port as well as communicating with the Internet application by sending and receiving information.

Published Data

Item	Description
Computer	The name of the computer where the Internet application resides.
Port	The port used to communicate with the Internet application.
Protocol	The protocol of the Internet application. For example, HTTP or FTP.
Server Greeting	The greeting message received from the Internet application. This published data is only available in FTP, Email (POP3), and Email (SMTP).
Web page	The HTML of the web page that was retrieved when in Web (HTTP) mode.
Receive variable 1	The first variable retrieved when in Custom mode.
Receive variable 2	The second variable retrieved when in Custom mode.
Receive variable 3	The third variable retrieved when in Custom mode.
Receive variable 4	The fourth variable retrieved when in Custom mode.
Receive variable 5	The fifth variable retrieved when in Custom mode.
Receive variable 6	The sixth variable retrieved when in Custom mode.
Receive variable 7	The seventh variable retrieved when in Custom mode.
Receive variable 8	The eighth variable retrieved when in Custom mode.
Receive variable 9	The ninth variable retrieved when in Custom mode.
Receive variable 10	The tenth variable retrieved when in Custom mode.

Monitor WMI

The Monitor WMI activity invokes a runbook when a WMI event is received as a result of the WMI event query that you specify. You can check for changes in devices that are attached to the server and invoke runbooks that take corrective action when errors occur.

Configuring the Monitor WMI Activity

Before you configure the Monitor WMI activity, you need to determine the following:

- The computer that you are monitoring
- The WMI event query that you want to run



Warning

A WMI event query differs from a standard WMI query.

Use the following information to configure the Monitor WMI activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer that you are monitoring for new WMI events. You can also use the ellipsis () button to browse for the computer.
Namespace	Type the name of the WMI namespace that you want to query.
WMI query	Type the WMI event query that will be used to query the computer that you specified in the Computer box.

Syntax Examples

Here is the syntax of a simple notification query: SELECT * FROM [EVENTClass] WITHIN [interval] WHERE TargetInstance ISA [object]

When you submit this WMI query, you are submitting a job to be notified of all occurrences of the event represented by [EventClass]. The WITHIN clause denotes how the test is performed, which is at an interval of seconds denoted by [interval]. The WHERE clause is used to narrow down your query and can include activities, properties of embedded activities and condition statements.

Monitor for the Addition of a Modem: The following query submits a notification job to monitor for the addition of a modem and will cause the WMI event to invoke if a modem is added. The test is performed at an interval of every 10 seconds. SELECT * FROM InstanceCreationEvent WITHIN 10 WHERE TargetInstance ISA "Win32 POTSModem"

Monitor for the Deletion of a Modem: The following query submits a notification job to monitor for the deletion of a modem and will cause the WMI event to invoke if a modem is deleted. The test is performed at an interval of every 50 seconds. SELECT * FROM __InstanceDeletionEvent WITHIN 50 WHERE TargetInstance ISA "Win32 POTSModem"

Monitor for the Modification of a Display Configuration: The following query submits a notification job to monitor for the modification of a display configuration and will cause the WMI event to invoke if the display frequency is greater than 70. The test is performed at an interval of every 20 seconds. SELECT * FROM __InstanceModificationEvent WITHIN 20 WHERE TargetInstance ISA "Win32_DisplayConfiguration" AND TargetInstance.DisplayFrequency > 70

Monitor for a Modification in a Processor value: The following query submits a notification job to monitor for a modification in a Processor value and will cause the WMI event to invoke if the CPU utilization is greater than 50. The test is performed at an interval of every 5 seconds. Select

* FROM __InstanceModificationEvent WITHIN 5 WHERE TargetInstance ISA "Win32_Processor" AND TargetInstance.LoadPercentage > 50



Tip

A query can be rejected by WMI if it is too complex or becomes resource-intensive for evaluation.

Published Data

The following table lists the published data items.

Item	Description
Computer where the WMI query is performed	The name of the computer where the WMI query was ran.
WMI Query	The WMI query that was sent to the computer.
WMI Query Result as a string	The result of the WMI query.
WMI Namespace	The WMI namespace that you queried.

File Management

The following table provides a brief description of tasks you can accomplish when using each File Management activity.

Tasks	File Management Activities
Compress files into zip archives.	Compress File
Copy files from one directory to another.	Copy File

Tasks	File Management Activities
Create new folders.	Create Folder
Decompress files contained in a zip archive file.	Decompress File
Delete files.	Delete File
Delete a folder, sub-folder, or the entire folder tree of a directory.	Delete Folder
Verify that a file exists.	Get File Status
Invoke a runbook when files in folders and subfolder change.	Monitor File
Invoke a runbook when a folder or files within a folder change.	Monitor Folder
Move a file from one directory to another.	Move File
Move a folder and its sub-folders from one directory to another.	Move Folder
Decrypt a file or an entire folder tree.	PGP Decrypt File
Encrypt a file or an entire folder tree.	PGP Encrypt File
Print text files.	Print File
Rename files.	Rename File



Caution

If permissions on the Orchestrator installation path are changed and the activity's Security Credentials has a custom user account that does not include Read/Execute permissions to ExecutionData.dll on the Runbook server, the activity will fail.

Compress File

The Compress File activity compresses files into zip archives. You can use the Compress File activity to archive log files before storage or before sending them to another location using FTP or email.

Configuring the Compress File Activity

Before you configure the Compress File activity, you need to know which files you will compress. Use the following information to configure the Compress File activity.

Settings	Configuration Instructions
Settings	Configuration Instructions
Folder	Type the path to the file, or to the folder that contains the files, that you want to compress, or click the ellipsis () button and browse for the files. You can use wildcards in filenames. You cannot browse for the folder name; you must type in the full folder name and location and include a trailing slash.
Include files in sub-folders	Select this option to include any files that are found within sub-folders of the folder that you specified.
File	Type the path and filename of the archive that you are creating. This field will only accept characters from the current system locale. If you use other characters, the activity will fail.
Store relative path in archive	Select this option to store the files within the same sub-folders that they were found in. When this option is unselected, the files will be added to the archive with the full path. For example:
	Selected:\subfolder1\file.txt, and\subfolder1\subfolder2\file.txt
	Unselected: C:\files\subfolder1\file.txt, and C:\files\subfolder1\subfolder2\file.txt
If the destination archive already exists	Select the action that you want to take if a file with the same name as the archive being created already exists in the destination folder:
	Add files to the existing archive: Select this option to add the files that you specified to the existing archive.
	Overwrite the existing archive: Select this option to overwrite the existing file with the archive that you are creating.
	Fail if the archive exists: Select this option to cause the Compress File activity to fail if the filename already exists.
	Create a unique named archive: Select this option to append a value to the filename to

Settings	Configuration Instructions
	create a unique filename that does not conflict with the existing filename.
Compression level	Select the level of compression that you want to use to compress the files into the archive. You can select one of the following levels. None
	• Low
	Medium
	High
	Higher compression levels take more time to complete but usually result in smaller files. Lower compression levels create larger archives, but take less time to complete.

The following table lists the published data items.

Item	Description
Archive name and path	The name and path of the archive file that was created.
Number of files within archive	The number of files inside the archive file.
Size of archive	The size of the archive file.

Copy File

The Copy File activity copies a file from one directory to another. You can also copy files to network shares that are available using UNC paths. Use the Copy File activity to copy important files that have been created or modified in a folder that is being monitored by the Monitor Folder activity to a backup location.

Configuring the Copy File Activity

Before you configure the Copy File activity, you need to know which files you are copying and the destination path where you will put the copies.

Use the following information to configure the Copy File activity.

Details Tab

Settings	Configuration Instructions
File	Type the path and name of the file that you want to copy. You can use the * and ? wildcards to specify the file name or path. These wildcards behave the same way as in the Windows Command Prompt.
Include sub-folders	Select this option to copy any files within the sub-folders of the path you have specified that match the filename that you have specified.
Folder	Type the path of the folder where you want the files to be copied to.
If the destination exists	Select the action that you want to take if a file with the same name already exists in the destination folder:
	Overwrite: Select this option to overwrite the existing file with the file that is being copied.
	Fail : Select this option to cause the Copy File activity to fail if the filename already exists.
	Create a file with a unique name: Select this option to append a value to the filename to create a unique name that does not conflict with an existing name.

Advanced Tab

Settings	Configuration Instructions
File age	Specify Is less than or Is more than to copy the files that are older or newer, respectively, than the number of days that you specify.
days	Enter the number of days that you will use with the File age measure.
Date of transfer	Set the date of the file at the destination to the date when it was copied to the folder.
Same as original	Set the date of the file at the destination to the date of the original file.

Item	Description
Origin folder	The path of the base folder where the file was copied from.
Destination folder	The destination folder where the file was copied to.
Total number of files to be transferred	The number of files that matched the criteria that you specified.
Number of successful file operations	The number of files that were successfully copied.
Number of failed file operations	The number of files that failed to copy.
File operation status	Determines whether the copy succeeded or failed.
File path	The path of the file that was copied.
File name	The name of the file that was copied.
Name and path of the file relative to the origin folder	The relative path of the file starting from the origin folder.
If destination exists	The option that was selected to handle the operation if the destination file already exists.
File age date options	The option that was selected to evaluate the file age.
File age days	The number of days that was provided to evaluate the file age.
Modified date option	The option that was selected for the date to be assigned to the destination file.
Name and path of the destination file	The name and path that the file was copied to.
Name and path of the origin file	The name and path that the file was copied from.
Include sub-folders	Indicates whether the Include sub-folders check box was selected.
Origin folder	The path of the base folder where the file was copied from.

Item	Description
Destination folder	The destination folder where the file was copied to.
Total number of files to be transferred	The number of files that matched the criteria that you specified.
Number of successful file operations	The number of files that were successfully copied.
Number of failed file operations	The number of files that failed to copy.
File operation status	Determines whether the copy succeeded or failed.
File path	The path of the file that was copied.
File name	The name of the file that was copied.
Name and path of the file relative to the origin folder	The relative path of the file starting from the origin folder.
If destination exists	The option that was selected to handle the operation if the destination file already exists.
File age date options	The option that was selected to evaluate the file age.
File age days	The number of days that was provided to evaluate the file age.
Modified date option	The option that was selected for the date to be assigned to the destination file.
Name and path of the destination file	The name and path that the file was copied to.
Name and path of the origin file	The name and path that the file was copied from.
Include sub-folders	Indicates whether the Include sub-folders check box was selected.

Create Folder

The Create Folder activity creates a new folder on the local file system or a network location specified using a UNC path. Use the Create Folder activity to create folders dynamically with names that represent the context in which they were created. For example, on August 25 you can create "C:\backupfolderAug25".

Configuring the Create Folder Activity

Before you configure the Create Folder activity, you need to know the name of the folder that you are creating.

Use the following information to configure the Create Folder activity.

Published Data

The following table lists the published data items.

Item	Description
Folder path	The path of the folder that was created.

Decompress File

The Decompress File activity decompresses the files contained in a zip archive file. You can extract files from zip archives that are downloaded using email or FTP.

Configuring the Decompress File Activity

Before you configure the Decompress File activity, you need to determine the following:

- The archive file name that you want to decompress.
- The files names within the archive that you want to extract.

Use the following information to configure the Decompress File activity.

Settings	Configuration Instructions
File	Type the path of the archive file that you want to extract files from.
Files to extract	Type the name of the file that you want to extract. You can use the * and ? wildcards to specify the file name. These wildcards behave in the same way as in the Windows Command Prompt.
Folder	Type the folder name to which the files will be extracted, or click the ellipsis () button and browse for it.
Reproduce tree	Select this option to extract the files to the

Settings	Configuration Instructions
	same relative paths that they were saved in. To use this feature, the relative paths must have been stored in the zip archive when it was created.
If the destination file exists	Select the action that you want to take if a file with the same name as the file being extracted exists in the destination folder:
	Create a file with a unique name: Select this option to append a value to the filename to create a unique filename that does not conflict with an existing filename.
	Overwrite: Select this option to overwrite the existing file with the file that you are extracting.
	Fail: Select this option to cause the Decompress File activity to fail if the file name already exists.

The following table lists published data items.

Item	Description
Archive name and path	The name of the archive file that was decompressed.
Number of files within archive	The total number of files that are inside the archive file.
Size of archive	The size of the archive file.
Size of the decompressed files	The total size of the files decompressed.

Delete File

The Delete File activity deletes files from the local file system or from a network location specified using a UNC path. You can purge a folder that contains old log files.

Configuring the Delete File Activity

Before you configure the Delete File activity, you need to know which files you are deleting. Use the following information to configure the Delete File activity.

Details Tab

Settings	Configuration Instructions
Path	Type the path and name of the file that you want to delete. You can use the * and ? wildcards to specify the file name. These wildcards behave in the same way as the Windows Command Prompt.
Delete files from sub-folders	Select this option to delete any files within the sub-folders of the path you have specified that match the file name that you have specified.
File age	Select the Is less than or Is more than option from the drop-down list to delete the files that are older or newer, respectively, than the number of days that you specify.
days	Type the number of days that you will use with the file age measure.

Published Data

Item	Description
File age days	The number of days that was provided to evaluate the file age.
File age option	The option that was selected to evaluate the file age.
Name and path of the file	The name and path of the file that was deleted.
File name	The name of the file that was deleted.
Name and path of the file relative to the origin folder	The relative path of the file starting from the origin folder.
File operation status	Determines whether the delete operation succeeded or failed.

Item	Description
Origin folder	The path of the base folder where the file was deleted from.
Number of failed file operations	The number of files that were not deleted.
Number of successful file operations	The number of files that were successfully deleted.
Total number of files	The number of files that matched the file that you specified.
File path	The path of the file that was deleted.
Delete files from sub-folders	Indicates whether the Delete files from sub-folders check box was selected.

Delete Folder

The Delete Folder activity deletes a folder, sub-folders, or the entire folder tree of a directory on the local file system or a network location specified using a UNC path. You can delete temporary folders that were created when a runbook runs or you can use this activity to purge data that has been recently archived.

Configuring the Delete Folder Activity

Before you configure the Delete Folder activity, you need to determine the following:

- The folder name you are targeting.
- Whether you are going to delete the entire tree; delete the sub-folders only; or delete just the directory.

Use the following information to configure the Delete Folder activity.

Settings	Configuration Instructions
Path	Type the path of the folder that you are targeting.
Delete the folder only if it is empty	Select this option to delete the folder only if there are no files or sub-folders in it.
Delete all files and sub-folders	Select this option to delete the specified folder and all sub-folders and files contained in that

Settings	Configuration Instructions
	folder.

The following table lists the published data items.

Item	Description
Folder path	The path of the folder that was deleted.
Folder pattern to match	The pattern used to find the sub-folder that was deleted.
Base Folder to start deletion from	The Path that was specified on the Details tab.
Delete folder options	The option that you selected for the delete folder operation.
Name and path of the folder	The name and path of the folder that was deleted.

Get File Status

The Get File Status activity verifies that a file exists on the local file system or a network location using a UNC path. You can check that a file is available before copying to another location or before starting any services that depend on the existence of the file. If the file does not exist, you can take corrective action using the Copy File activity to copy the file from another location.

Configuring the Get File Status Activity

Before you configure the Get File Status activity, you need to determine the file name and path name you are checking.

Use the following information to configure the Get File Status activity.

Settings	Configuration Instructions
File	Type the name and path of the file that you are checking the status of, or click the ellipsis () button and browse for it.
Include sub-folders	Select this option to copy any files within the

Settings	Configuration Instructions
	sub-folders of the path you have specified that match the filename that you have specified.
File age	Select Is less than or Is more than to specify the files that are older or newer, respectively, than the number of days that you specify.
days	Type the number of days that you will use with the File age measure.

Item	Description
Date and time the file was created	The local date and time on which the file was created.
UTC date and time the file was created	The UTC date and time on which the file was created.
File age days	The number of days that was provided to evaluate the file age.
Modified date option	The option that was selected to search for files according to a date range.
File exists	Indicates whether the file exists or not.
File name extension	The extension, or file type, of the file.
File folder	The folder that the file was found in.
File name	The name of the file.
File owner	The name of the owner of the file.
File size (bytes)	The size of the file in bytes.
Name and path of the origin file	The file name and path that was provided.
Last accessed date and time	The date and time on which the file was created in localized format.
Last accessed UTC date and time	The date and time on which the file was created in UTC format.
Last modified date and time	The date and time on which the file was

Item	Description
	created in localized format.
Last modified UTC date and time	The date and time on which the file was created in UTC format.
Include sub-folders	Indicates whether the Include sub-folders check box was selected.
File path	The source file name and path.
Encoding type (text files only)	The file encoding format used by the file, if the file is a text file.

Monitor File

The Monitor File activity invokes a runbook when files that you specify in folders and sub-folders have changed. You can monitor a file that indicates the completion of a transaction. For example, there are nightly transfers sent to your runbook server, and when the transfer is complete a file with the name "Complete" is written to the folder. This activity can automatically invoke a runbook that processes all the files in the folder when the "Complete" file is created.

Configuring the Monitor File Activity

Use the following information to configure the Monitor File activity.

Settings	Configuration Instructions
In folder	Type the path to the file that you are monitoring, or use the ellipsis () button to browse for it.
Include sub-folders	Select this option to copy any files within the sub-folders of the path you have specified that match the filename that you have specified.
Filters	Create filters with custom criteria for the files that you want to monitor. Perform the following for each filter that you want to create: 1. Click Add to open the Filter Settings

Settings	Configuration Instructions
	dialog box. 2. From the Name drop-down list, select the criteria that you want to use. The Relation and Value menu options present custom options according to the criteria that you select from the Name list.
	 Select options from the Relation and Value items.
	4. Click OK .

Triggers Tab

Settings	Configuration Instructions
Trigger if one of the files was	Select a condition to invoke the activity if the condition in the monitored file is true.
Trigger if file properties changed	Select a condition to invoke the activity if the condition in the monitored file is true.

Authentication Tab

Settings	Configuration Instructions
User name	Type the user name required to access the folder if it is on a remote computer.
Password	Type the password required to access the folder if it is on a remote computer.

Published Data

Item	Description
Change type	The type of changed that was detected on the file.
Name and path of the file	The name and path of the file that was monitored.
Include sub-folders	Indicates that the Include sub-folders check

Item	Description
	box was selected.
Notify if changed	Indicates that the Changed check box was selected.
Notify if created	Indicates that the Created check box was selected.
Notify if deleted	Indicates that the Deleted check box was selected.
Notify if renamed	Indicates that the Renamed check box was selected.
Origin Folder	The folder that the monitored file was stored in.
Notify if file attributes changed	Indicates that the Attributes check box was selected.
Notify if file creation time changed	Indicates that the Creation time check box was selected.
Notify if file last access time changed	Indicates that the Last access time check box was selected.
Notify if file last write time changed	Indicates that the Last write time check box was selected.
Notify if file security changed	Indicates that the Security check box was selected.
User name	The user name used to access the folder if it was on a remote computer.

Monitor Folder

The Monitor Folder activity invokes a runbook when the folder that you specified has changed, or if the files within that folder have been changed. You can monitor the size of log files in a folder. If the files grow too large, the Monitor Folder activity can invoke a runbook that will archive, backup, and then purge the log files to clean up the folder.

Configuring the Monitor Folder Activity

Before you configure the Monitor Folder activity, you need to determine the following:

• The folder name you are monitoring.

- What condition invokes the runbooks.
- Optionally, you may need to know what file types you want to monitor.

Use the following information to configure the Monitor Folder activity.

Details Tab

Settings	Configuration Instructions
Path	Type the path to the folder that you are monitoring. You can use the ellipsis () button to browse for the folder.
Include sub-folders	Select this option to monitor the files and folders in sub-folders in the Folder that you specified.
File Filters	Create filters with custom criteria for the files that you want to monitor. Perform the following for each filter that you want to create:
	Click Add to open the Filter Settings dialog box.
	 From the Name menu list, select the criteria that you want to use. The Relation and Value menu options present custom options according to the criteria that you select from the Name menu items.
	Select from the Relation and Value menu items.
	4. Click OK .

Triggers Tab

Settings	Configuration Instructions
Number of files is	Select this option to invoke the Monitor Folder activity if the number of files is greater than, equal to, or less than the value that you provide. Select the criteria from the drop-down list and type the value in the field.
Total file size is	Select this option to invoke the Monitor Folder activity if the total file size of the folder is greater than or less than the value that you

Settings	Configuration Instructions
	provide. Select the criteria from the first drop- down list, type the value in the field, and select the unit of measure from the last drop-down list.

Authentication Tab

Settings	Configuration Instructions
User name	Type the user name required to access the folder if it is on a remote computer.
Password	Type the password required to access the folder if it is on a remote computer.

Published Data

Item	Description
Include sub-folders	Indicates that the Include sub-folders check box was selected.
Trigger if number of files changed	Indicates that the Number of files is check box was selected.
Trigger if total file size changed	Indicates that the Total file size is check box was selected.
Number of files	The number of files given to evaluate the Number of files is option.
Number of files relation	The relation that was used to evaluate the Number of files is option.
Number of files limit	
Origin Folder	The folder that was monitored.
Total file size measure	The unit of measure selected to evaluate the Total file size is option.
Total file size relation	The relation that was used to evaluate the Total file size is option.
Total file size limit	

Item	Description
Total file size number	The number given to evaluate the Total file size is option.
User name	The user name used to access the folder if it was on a remote computer.

Move File

The Move File activity moves a file from one directory to another. You can move files to network shares that are available using UNC paths. You can also move files from a local or publicly available network folder, such as an FTP location, to an internal folder.

Configuring the Move File Activity

Before you configure the Move File activity, you need to determine the following:

- The files you are moving.
- The destination path where you will move the files.

Use the following information to configure the Move File activity.

Settings	Configuration Instructions
File	Type the path and name of the file that you want to move. You can use the * and ? wildcards to specify the filename and path. These wildcards behave the same way as in the Windows Command Prompt.
Include sub-folders	Select this option to move any files within the sub-folders of the path you have specified that match the filename that you have specified.
Folder	Type the path of the folder where you want the files to be moved to.
If the destination exists	Select the action that you want to take if a file with the same name already exists in the destination folder:
	Overwrite: Select this option to overwrite the existing file with the file that is being moved. Fail: Select this option to cause the Move File

Settings	Configuration Instructions
	activity to fail if the filename already exists.
	Create a file with a unique name: Select this option to append a value to the filename to create a unique name that does not conflict with an existing name.

Advanced Tab

Settings	Configuration Instructions
File age	Select Is less than or Is more than from the drop-down list to move the files that are older or newer, respectively, than the number of days that you specify.
days	Enter the number of days that you will use with the File age measure.
Date of transfer	Set the file date at the destination to the date when it was copied to the folder.
Same as original	Set the date of the file at the destination to the date of the original file.

Published Data

Item	Description
Origin folder	The path of the base folder where the file was moved from.
Destination folder	The destination folder where the file was moved to.
Total number of files to be transferred	The number of files that matched the file that you specified.
Number of successful file operations	The number of files that were successfully moved.
Number of failed file operations	The number of files that failed to move.
File operation status	Determines whether the move operation

Item	Description
	succeeded or failed.
File path	The path of the file that was moved.
File name	The name of the file that was moved.
Name and path of the file relative to the origin folder	The relative path of the file starting from the origin folder.
If destination exists	The option that was selected to handle the operation if the destination file already exists.
File age date option	The option that was selected to evaluate the file age.
File age days	The number of days that was provided to evaluate the file age.
Modified date option	The option that was selected for the date to be assigned to the destination file.
Name and path of the destination file	The name and path that the file was moved to.
Name and path of the origin file	The name and path that the file was moved from.
Include sub-folders	Indicates whether the Include sub-folders check box was selected.

Move Folder

The Move Folder activity moves a folder and its sub-folders from one directory to another. You can also move folders to network shares that are available using UNC paths. In addition, you can take files from a local or network folder that are made publicly available as an FTP location and move them to an internal folder.

Configuring the Move Folder Activity

Use the following information to configure the Move Folder activity.

Settings	Configuration Instructions
Source	Type the path of the folder that you want to
	move, or click the ellipsis () button to browse

Settings	Configuration Instructions
	for it. The Move Folder activity does not support the * and ? wildcards.
Destination	Type the path and name that you want to move the folder to, or click the ellipsis () button to browse for it.

The following table lists the published data items.

Item	Description
Origin name of the folder	The path of the original folder that was moved.
Destination name of the folder	The destination folder where the folder was moved to.
New folder path	The new path of the folder that was moved.

PGP Decrypt File

The PGP Decrypt File activity decrypts a file or entire folder tree using a PGP key file and passphrase that you have created. When decrypting an entire folder, the folder tree is preserved from the root folder down. For example, if you decrypt C:\Documents and Settings\Administrator\My Documents*.* and all subfolders, all files in My Documents are decrypted as well as all the files in the folders under My Documents. All files in subfolders will be in the same subfolder in the Output folder.

You can use the PGP Decrypt File activity to decrypt files that were encrypted as part of a backup operation. To use this activity you must install the Gpg executable. To install the Gpg executable, see Install GnuPG (http://go.microsoft.com/fwlink/?LinkId=219849).

Configuring the PGP Decrypt Activity

Use the following information to configure the PGP Decrypt File activity.

Settings	Configuration Instructions
Path	Type the path of the files that you want to decrypt. You can use wildcards? and * to
	decrypt. You can use wildcards? and to

Settings	Configuration Instructions
	specify the files that you are decrypting. This field will only accept characters from the current system locale. If you use other characters, the activity will fail.
Include sub-directories	Select this option to find all files that match the file name that you specified in all sub-directories under the folder that you specified in the path.
Output folder	Type the path of the folder where you want the decrypted files to be stored.
Skip	Select this option to skip decrypting a file when a file with the same name is found in the Output folder .
Overwrite	Select this option to overwrite any files with the same name as a resulting decrypted file.
Create unique name	Select this option to give the decrypted file a unique name if a file with the same name already exists.

Advanced Tab

Settings	Configuration Instructions
Keyring folder	Type the location of the keyring folder that contains the secret keyring file that you will use to decrypt the files. The secret keyring file (*.skr) may be renamed with a *.gpg extension.
Passphrase	Type the passphrase that is associated with the keyring file.

Published Data

Item	Description
Keyring folder	The path of Keyring folder that contains the key
	used to decrypt the files.

Item	Description
Output folder	The path of the folder where the decrypted files were saved.
Files to decrypt	The number of files that Orchestrator attempted to decrypt.
Files decrypted	The number of files that were successfully decrypted.
Decrypted filename	The path and filename of the resulting decrypted file.

PGP Encrypt File

The PGP Encrypt File activity encrypts a file or an entire folder tree using a PGP key file that you have created. When encrypting an entire folder, the folder tree is preserved from the root folder down. For example, if you encrypt C:\Documents and Settings\Administrator\My Documents*.* and all subfolders, all files in My Documents are encrypted as well as all files in folders under My Documents. All files that are in subfolders will be in the same subfolder in the Output folder. Use the PGP Encrypt File activity to encrypt files before backing them up.

To use this activity you must install the Gpg executable. To install the Gpg executable, see Install GnuPG.



Important

This activity supports DSS and RSA4 keys.

RSA keys are not supported by this activity.

Configuring the PGP Encrypt File Activity

Before you configure the PGP Encrypt File activity, you need to determine the following:

- The path of the files that you want to encrypt.
- The output folder where the encrypted files will be stored.

Use the following information to configure the PGP Encrypt File activity.

Details

Settings	Configuration Instructions
Path	Type the path of the files that you want to
	encrypt. You must use the full path name. You
	can use wildcards? and * to specify the files

Settings	Configuration Instructions
	that you want to encrypt. This field only accepts characters from the current system locale.
Include sub-directories	Select this option to find all the files that match the filename that you specified in all the subfolders of the folder that you specified in the path.
Output folder	Type the path of the folder where you want the encrypted files to be stored.
Skip	Select this option to skip encrypting a file when a file with the same name is found in the Output folder.
Overwrite	Select this option to overwrite any files with same name as the resulting encrypted file.
Create unique name	Select this option to give the encrypted file a unique name if a file with the same name already exists.
File extension	Type the file name extension that you want to appended to the file name when it is encrypted. The default extension is gpg.

Advanced

Settings	Configuration Instructions
Key file	Type the location of the PGP key file that you will use to encrypt the files. If you leave this field blank, the PGP Encrypt File activity uses the file that you specify in the Keyring folder field. Files can have any file name extension, but *.asc is the standard.
Keyring folder	Type the location of the folder that contains the keyring that you will use to encrypt the files. The public keyring file (*.pkr) may be renamed with a *.gpg file name extension.
	Important The PGP Encrypt File activity creates files in the keyring folder. The Orchestrator Runbook Service account,

Settings	Configuration Instructions
	or the user account used to run the runbook, requires read and write permissions on the keyring folder.
User	Type the user name that was specified when the encryption key was created. This is a required field.
Comment	Type the comment that was specified when the encryption key was created. If this field was completed when the encryption key was created, you must provide this information when using this activity.
Email	Type the email address that was specified when the encryption key was created. This is a required field.

Item	Description
Key file	The path of the key file used to encrypt the files.
Keyring folder	The path of keyring folder that contains the key used to encrypt the files.
User	The name of the user that was used to encrypt the files.
Comment	The comment that was used to encrypt the files.
Email	The email address that was used to encrypt the files.
Output folder	The path of the folder where the encrypted files were saved.
Files to encrypt	The number of files that Orchestrator attempted to encrypt.
Files encrypted	The number of files that successfully encrypted.

Item	Description
Encrypted filename	The path of the resulting encrypted file.

Print File

The Print File activity prints text files to a printer that you specify. You can use this activity to print log files for paper filing before the data is moved or deleted from a server.

Configuring the Print File Activity

Before you configure the Print File activity, you need to determine the following:

- File name you are printing.
- Printer name

Use the following information to configure the Print File activity.

Details Tab

Settings	Configuration Instructions
File	Type the path and name of the file that you want to print.
Printer	Type the path of the printer that will print the file.
Age	Specify is less than or is more than to print the files that are older or newer, respectively, than the number of days that you specify.
days	Enter the number of days that you will use with the Age measure.

Published Data

Item	Description
Origin folder	The path of the base folder where the file was printed from.
Number of successful file operations	The number of files that were successfully printed.

Item	Description
Number of failed file operations	The number of files that failed to print.
For each file:	
Name and path of the file	The path of the file that was printed.
Name of the printer	The printer that was used to print the file.

Rename File

The Rename File activity renames files on the local file system or on a network location specified using a UNC path. You can automatically rename files to a standard format according to your data center procedures.

Configuring the Rename File Activity

Before you configure the Rename File activity, you need to determine the following:

- The original file name you are renaming.
- The new name of the file

Use the following information to configure the Rename File activity.

Settings	Configuration Instructions
Folder	Type the path of the folder that contains the files that you want to rename.
Include sub-folders	Select this option to rename any files in the subfolders of the folder that you specified that match the file names that you want to rename.
Destination	This list displays all the file names that will be renamed when this activity runs. To add a filename, click Add to open the Rename Properties dialog box, and specify the Old name and then specify the file New name . To edit the list of file names, click Edit . To remove file names, click Remove .

Advanced Tab

Settings	Configuration Instructions
File age	Specify Is less than or Is more than to rename the files that are older or newer, respectively, than the number of days that you specify.
days	Type the number of days that you will use with the File age measure.
Date of rename	Select this option to set the date of the file at the destination to the date when it was renamed.
Same as original	Select this option to set the date of the file at the destination to the date of the original file.
Destination already exists	Select the action that you want to take if a file with the same name already exists in the folder: Overwrite: Select this option to overwrite the existing file with the file that is being renamed. Fail: Select this option to cause the Rename File activity to fail if the filename already exists. Create a file with a unique name: Select this option to append a value to the filename to create a unique name that does not conflict with an existing name.

Published Data

Item	Description
File path	The path of the file that was renamed.
Include sub-folders	Indicates whether this option was selected.
Destination folder	The destination folder of the file that was renamed.
If destination exists	The option that was selected for handling the filename if the file existed in the destination folder.

Item	Description
Modified date option	The option that was selected for assigning a modified date to the file in the destination folder.
File age days	The number of days used in the File age filter.
Destination date	The option that was selected for assigning a destination date to the file in the destination folder.
Total number of files to be renamed	The number of files that were renamed by the operation.
Number of successful file operations	The number of successful operations that occurred.
Number of failed file operations	The number of failed operations that occurred.
Origin folder	The folder where the file originated from.
Name and path of the destination file	The name and path of the destination file.
File name	The filename of the origin file.
Name and path of the file relative to the origin folder	The relative path of the file, relative to the origin folder.
Name and path of the origin file	The name and path of the origin file.
File operation status	The status of the rename operation.
Pattern that matched file	The pattern that the user entered that matched the file or files that were found.
Pattern file renamed to	The pattern that the file or files were renamed.

Email

The following table provides a brief description of tasks you can accomplish when using an Email activity.

Tasks	Email Activities
Send an email message.	Send Email

Send Email

The Send Email activity sends an email message using the standard SMTP protocol or an Exchange server. You can use this activity to notify an administrator of problems that have occurred with a system.



Important

If you put more than 1 MB of text directly into the message body, the activity can fail during initialization. To avoid this issue, enter no more than 1 MB of text directly into the message body or save the text to a file, and provide the file name as the message you want to send.

Configuring the Send Email Activity

Before you configure the Send Email activity, you will need to determine the following:

- Your SMTP server information
- The recipient who will receive the email message.
- The email message you want to send.

Use the following information to configure the Send Email activity.

Details

Settings	Configuration Instructions
Subject	Type the subject of the email.
Recipients	The list displays the email addresses that the email will be sent to. To add a recipient, click Add to open the Recipients Properties dialog box, specify the Email address and from the Recipient type box, select To, Cc, or Bcc, and then click OK.
	To remove a recipient, select the recipient in the Recipients and click Remove . To edit a recipient, double-click the recipient in the Recipients box.
Message	Select how you want the message to be entered for this email: Text: Type the message body. To use HTML formatting, you will need to select HTML as the

Settings	Configuration Instructions
	Format on the Advanced tab.
	File: Type the name of the file that contains the message body. To browse for the file name, click the ellipsis () button next to the Message box.
Attachments	The list displays the attachments that will be sent with the email. To add an attachment, click Add to open the Attachment Properties dialog box, specify the path of the attachment or click the ellipsis () button next to the File box, and then click OK.
	To remove an attachment, select the attachment in the Attachments box, and click Remove . To edit an attachment, double-click the attachment in the Attachments box.
Task fails if an attachment is missing	Select this box to cause the Send Email activity to fail if any of the attachments cannot be found when the email is being sent.

Advanced

Settings	Configuration Instructions
Priority	Select the priority of the email from the drop- down list. You can select Normal , Low , or High .
Format	Select the format that will be used for the message body. You can select Rich Text , ASCII , or HTML .
	Note Some SPAM filters may not allow Rich Text or HTML email.
User Id	If your SMTP server requires authentication, you will need to type the user ID that will be used to send the email.
Password	The password that is associated with the User ID.
Domain	The domain associated with the User ID.

Connect

Settings	Configuration Instructions
Email address	Type the email address that will be inserted into the From: field of the email.
Computer	Type the name of the SMTP server. You can also use the ellipsis () button to browse for the server.
Port	Select to change the port that will be used to connect to the SMTP server. The default port is 25.
Enable SSL	Select to indicate that the SMTP connection requires SSL.

Published Data

The following table lists the published data items.

Item	Description
Subject of the email	The subject of the email that was sent.
The email message Recipient	The address of the recipient of the email.
Body of the email message	The body of the email.
Name and path of the attached file	The full path of the file that was attached.
Email account	The SMTP account that was used to send the email.
Outgoing mail server (SMTP)	The name of the SMTP server used to send the email.
Outgoing mail server port number	The port used to communicate with the SMTP server.
Outgoing mail server SSL enabled	Indicates whether the mail server has SSL enabled.

Notification

The following table provides a brief description of tasks you can accomplish when using each Notification activity.

Tasks	Notification Activities
Create an entry in the Application Windows Event Log.	Send Event Log Message
Create a message on the Syslog server.	Send Syslog Message
Create an entry in the Application Windows Event Log.	Send Platform Event

Send Event Log Message

The Send Event Log Message activity creates an entry in the Windows Event Log within the Application folder. This activity can be used to create audit logs in the Windows Event Log that document any problems that occur while trying to correct issues by using an automated runbook.

Configuring the Send Event Log Message Activity

Before you configure the Send Event Log Message activity, you will need to determine the following:

- The event message you are creating.
- The severity of the event

Use the following information to configure the Send Event Log Message activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer that contains the Windows Event Log that you are writing to. You can also use the ellipsis () button to browse for the computer.
Message	Type the message text of the event log entry.
Severity	Select the severity level that is appropriate for

Settings	Configuration Instructions
	this event.
	You can select Information, Warning, or
	Error.

Published Data

The following table lists the published data items.

Item	Description
Computer	The computer where the event log message was created.
Log entry description	The description of the event log message.

Send Syslog Message

The Send Syslog Message activity creates a message on the Syslog server that you specify. You can use this activity to create audit logs on the Syslog server that document any problems that occur while trying to correct issues using an automated runbook.

Configuring the Send Syslog Message Activity

Use the following information to configure the Send Syslog Message activity.

Details Tab

Settings	Configuration Instructions
Computer	Type the name of the computer that contains the Syslog server that you are writing the message to. You can also use the ellipsis () button to browse for the computer.
Text	Type the message of the event log entry.
Priority	Select the priority from the drop-down menu that is appropriate for this message.
Facility	Select the facility from the drop-down menu that is appropriate for this message.

Published Data

The following table lists the published data items.

Item	Description
Computer	The computer where the Syslog server is located.
Priority	The priority of the message.
Facility	The facility that the message belongs to.
Message	The text of the message.

Send Platform Event

The Send Platform Event activity creates an activity event with text that you specify. You can use the Send Platform Event activity to create notifications of any problems or general information that occur in the runbook.

For more information about activity events, see Activity Events.

Activity Properties

Details Tab

Settings	Configuration Instructions
Туре	Type of event to create. You can select from the following values: Information Warning Error
Summary	Summary of the event that displays in the list in the Events tab of the Runbook Designer. This has a limit of 200 characters.
Details	Details of the event that displays when the event is opened. This has a limit of 2,000 characters.

Warning

The Runbook Designer does not warn you when you configure this activity and you exceed the limits for the **Summary** or **Details** settings . If you exceed these limits, the Runbook Designer does not allow you to check-in the runbook and a generic error is shown. The runbook server generates an error if it attempts to process data that exceeds these limits.

Published Data

Item	Description
Туре	The type of event that was generated.
Summary	The summary text of the event.
Details	The detailed description of the event.

Utilities

The following table provides a brief description of tasks you can accomplish when using each Utilities activity.

Tasks	Utilities Activities
Transform XML files.	Apply XSLT
Perform an XPath query on an XML file.	Query XML
Transform existing published data or variable items into new content.	Map Published Data
Compare two text values or two numerical values and determine whether or not they are equal.	Compare Values
Create or add information to an HTML file.	Write Web Page
Read lines in a structured text log file.	Read Text Log
Write a row into a database table.	Write to Database
Query a database and return the resulting rows as published data.	Query Database
Invoke a runbook when a counter has reached a specific value.	Monitor Counter

Tasks	Utilities Activities
Retrieve the value of a counter and return it as a published data item.	Get Counter Value
Increment and decrement a counter and reset it to its default value.	Modify Counter
Run a web service with XML parameters you specify.	Invoke Web Services
Transform existing date and time formats into customized formats.	Format Date/Time
Generate random strings of text.	Generate Random Text
Map a network path.	Map Network Path
Disconnect a network path.	Disconnect Network Path
Connect to a dial-up or VPN network.	Connect/Disconnect Dial-up
Get the status of a dial-up or VPN network.	Get Dial-up Status

Apply XSLT

The Apply XSLT activity enables you to transform the content of an XML file according to the rules in an XSLT file that you specify. You can use the Apply XSLT activity to transform the content of an XML file to an HTML file.

Configuring the Apply XSLT Activity

Before you configure the Apply XSLT activity, you need to determine the following:

- The name of the XML file that will be converted.
- The name that you want to assign to the XML file that results from the transformation.
- The name of the XSLT file that you will use to transform the XML file.

Use the following information to configure the Apply XSLT activity.

Details Tab

Settings	Configuration Instructions
Input XML file	Type the path and file name of the XML file that

Settings	Configuration Instructions
	you want to transform, or click the ellipsis button () and browse for it.
Output XML file	Type the path, filename, and file name extension for the file that will hold the results of the transformation. Alternatively, click the ellipsis button () and browse for the folder where you will save the file. From the Windows Open dialog box, enter the file name and file name extension in the File name box.
XSLT file	Type the path and name of the XSLT file that you want to use to transform the input XML file, or click the ellipsis button () and browse for it.

Published Data

The following table lists the published data items.

Item	Description
Input XML	The path and file name of the XML file that will be transformed.
Output XML	The path and file name of the XML file that will contain the result of the transformation.
XSLT file	The path and file name of the XSLT file used to transform the input XML file.

Query XML

The Query XML activity is used to perform an XPath query on an XML file. You can use this activity to search for a string in an XML file.

Configuring the Query XML Activity

Before you configure the Query XML activity, you need to determine the following:

- The XML file name or Block of XML that you want to search.
- The query you will use to perform the search.

Use the following information to configure the Query XML activity.

Details Tab

Settings	Configuration Instructions
XML File	Select either this option or the XML Text option. Type the path or URL of the XML file that you want to search in, or click the ellipsis button () and browse for it.
XML Text	Select either this option or the XML File option. Type the name of the element in the XML text that you want to search in.
XPath Query	Type the XPath query for your search.

Published Data

The following table lists the published data items.

Item	Description
Escaped Query Result	The result of the query.
Escaped XML Attributes	The attributes found in the element tag of the query result.
The input XML file	The name of the XML file that you are searching in. This item is blank if you used the Block of XML option.
The input XML text	The XML text that you searched in. This item is blank if you used the XML File option.
The XPath query.	The XPath query that was used in the search.
Node count	The number of results published from the query.

Map Published Data

The Map Published Data activity transforms the existing Published Data items or variable values into new values according to the rules that you specify. You can use this activity to convert numeric values to word values, simplify multiple versions of software program names into one name, or perform other string conversion activities within a runbook.

When you place this activity in a runbook, you must place it after the activities that create the Published Data items that you want to transform, and you must place it before activities that will use the new transformed items. If you use the Map Published Data activity to transform variable items, you can place it at the beginning of the runbook.

Configure the Map Published Data Activity

Before you configure the Map Published Data activity, you need to determine the following:

- The names of the Published Data or variables that you want to transform.
- The method you want to use to transform the Published Data or variable.

Use the following procedure to configure the Map Published Data activity.

To configure the Map Published Data activity

- 1. From the **Activity pane**, drag a **Map Published Data** activity to the runbook.
- 2. Double-click the Map Published Data activity icon to open the Properties dialog box.
- 3. Configure the settings on the **Mapping rules** tab. Configuration instructions are listed in the following table.

Mapping rules

Settings	Configuration Instructions
Add	Click Add to open the Add Mapping dialog box.
Output Published Data	Type the name that you want to assign to the new Published Data item that you are creating.
Source data	Insert Published Data or variable items to map to the new Published Data item. To insert items, right-click the edit box and select Subscribe , select Published Data or Subscribe , and then select Variable . You can insert as many items as you want. You can also type text to transform to a new Published Data item.
Pattern	Type the existing pattern that you want to transform.
Мар То	Type the new text that replaces the text of those items that match Pattern. Click OK to return to the Map Published Data Properties dialog box.

Settings	Configuration Instructions
Add, Edit, Remove	If you want to add more rules, click Add and repeat the Pattern and Map To configuration instructions. On the Mapping rules tab, you will see a list of all transformations you created.
	To remove items from the rules list, click Remove . To edit an item in the rules list, click Edit .

Examples

The following examples describe how to use the Map Published Data activity.

• Single Published Data or variable item

The <u>Read Line</u> activity creates a Published Data item called File and path name. If the path in this item is expressed as a drive letter, you can create a mapping to convert it to a UNC path.

To map a drive letter to a UNC path

- 1. In the **Source data** field, insert the File and path name Published Data item from the Read Line activity.
- 2. In the **Pattern** field, type the drive letter and a colon, such as **Y**:
- 3. In the **Map to** field, type the UNC path that will replace Y:, such as \\servername\folder.
- 4. Click Add, then OK.
- Convert output of one system to be compatible with another system's formatting
 You use two software programs that express severity levels with the following methods:
 - Numerically: 0, 1, 2, and so on.
 - Descriptions: High, Medium, or Low

To convert the numbers expressed by one software program to the words used by another

- 1. In the **Source data** field, insert the Published Data item for the severity level from the software program that expresses severity levels in numbers.
- 2. In the Pattern field, type 1.
- 3. In the **Map to** field, type **High**.
- 4. Click Add.
- 5. Repeat for each severity level, such as 2, 3, and 4, match the appropriate word to each numeric value.

6. Click **OK**.

To transfer severity levels to the software program that expresses them in words, insert the Published Data item that you created in the field. Items from the originating software program with a severity level of 1 are placed into the receiving software program with a severity level of High.

Wildcards

You can replace strings of words with wildcards combined with words. For example, a Published Data item can describe Windows Server 2008 R2 inconsistently, as either Win2K8R2, or W2K8R2.

There are two wildcards available:

- * use the asterisk to search for any number of characters after your alpha-numeric search character. For example, a* will produce aa, aaa, aaaa, aaabbb, and so on.
- ? use the question mark to find a specified number of characters after your alpha-numeric search character. For example, a?? will produce aaa, abb, abc, aac, but not aaaa or aaabbb, and so on.

To change variations in the data to a single value

- 1. In the **Source data** field, insert the Published Data item that represents the inconsistent names.
- 2. In the **Pattern** field, type **W*K8**.
- 3. In the Map to field, type Windows Server 2008
- 4. Click Add, and then click OK.

Multiple Published Data or Variable Items

Operating system names are usually composed of multiple parts, such as manufacturer, platform, version year, and release. In Orchestrator, each part of a name can be represented by an individual Published Data item. You can combine multiple Published Data items into one new item, such as Windows Server.

To map multiple values to a single value

- 1. In the **Source data** field, insert each Published Data item that you want to combine into the new item, separated by spaces.
- 2. In the Pattern field, type Windows Server*.
- 3. In the **Map to** field, type **Windows Server**.
- 4. Click **Add**, then **OK**. All references are changed to Windows Server.

Published Data

This activity only generates Published Data that you specify.

Compare Values

The Compare Values activity compares two text values or two numerical values and then determines whether or not they are equal. This activity can also be used to test error messages or numbers against known issues and automatically route the runbook to the appropriate activity.

Configuring the Compare Values Activity

Before you configure the Compare Values activity, you need to determine what type of values you want to compare.

Use the following information to configure the Compare Values activity.

General Tab

Settings	Configuration Instructions
Name	Type a descriptive name for the activity.
Description	Type a detailed description of the actions of the activity.
Туре	Select the Type from the drop-down list that matches the server you want to monitor. The options include the following: Compare Strings Compare Numeric Values Configuration instructions for each Details tab Type are listed in the following tables.

Details Tab Compare Strings

Settings	Configuration Instructions
Test	Type the first text, select how you want the first to be compared to the second text, and then type the second text. From the drop-down menu, when selecting the matches the pattern or does not match pattern comparisons, use the wildcards? and * to specify the pattern.
Case sensitive test	Select to cause the comparison to be case

Settings	Configuration Instructions
	sensitive.

Details Tab Compare Numeric Values

Settings	Configuration Instructions
Test that	Type the first number, select how you want the first to be compared to the second number, and then type the second number.

Published Data

The following table lists the published data items.

Item	Description
String to compare	The first string that was entered for the comparison. This published data is only available when Compare Strings is selected on the General tab.
String to compare to	The second string that was entered for the comparison. This published data is only available when Compare Strings is selected on the General tab.
Case sensitive comparison	Determines whether the comparison was case sensitive. This value can be either true or false.
Value to compare	The first value that was entered for the comparison. This published data is only available when Compare Numeric Values is selected on the General tab.
Value to compare to	The second value that was entered for the comparison. This published data is only available when Compare Numeric Values is selected on the General tab.
Comparison result	The result of the comparison. This value will be true if the two strings or numeric values match and false otherwise.

Write Web Page

The Write Web Page activity will create or add information to an HTML file. When creating new web pages with the Write Web Page activity, you can use templates that can contain any of the formatting that you require. When using the Write Web Page activity to create web pages, all occurrences of the tags <DOC-TITLE> and <DOC-TEXT> will be replaced by the title and text that you specify, respectively.

When using the Write Web Page activity to append to an existing web page, you will specify an HTML tag that will mark the position where the page will be appended. The Write Web Page activity will append the text above the HTML tag that you have specified.

The Write Web Page activity can be used to output the entire audit log of a runbook that was run to correct a problem on the network. You can also use the Write Web Page activity to keep a constant update of what maintenance runbooks are running and their status.

Configuring the Write Web Page Activity

Use the following information to configure the Write Web Page activity.

General Tab

Settings	Configuration Instructions
Name	Type a descriptive name for the activity.
Description	Type a detailed description of the actions of the activity.
Туре	Select the Type from the drop-down list that matches the server you want to monitor. The options include the following: • Create Web Page
	Append to Web Page
	Configuration instructions for each Details tab Type are listed in the following tables.

Details Tab Create Web Page

Settings	Configuration Instructions
Title	Type the title of the web page. If you are using a template, this value will replace <doc-title> anywhere it appears in the template</doc-title>
	file. If you are not using a template, this value will be used within the <title> tag in the</th></tr></tbody></table></title>

Settings	Configuration Instructions
	header of the HTML file.
Text	Type the text of the web page. If you are using a template, this value will replace <doc-text> anywhere it appears in the template file. If you are not using a template, this value will be used within the <body> tag in the header of the HTML file.</body></doc-text>
Template	Type the path and file name of the template file that you are using. You can also use the ellipsis () button to browse for the file.
Folder	Type the path of the folder where you want the web page to be created.
File name	Select to specify the file name of the web page that will be created.
Create a file with a unique name	Select to automatically generate a unique name for the web page when it is created. This file will have the extension .html.

Details Tab Append to Web Page

Settings	Configuration Instructions
Tag	Type the tag that will mark the point where the page will be appended. It is recommended to use the default tag <op-tag-append-web>.</op-tag-append-web>
Text	Type the text that will be appended to the web page.
Web page	Type the path and file name of the web page file that you are appending. You can also use the ellipsis () button to browse for the file.

Published Data

The following table lists the published data items.

Item	Description
Full path and name of Web Page	The full path of the HTML file that was created

Item	Description
	or appended.
Source text of the web page created	The text contained within the HTML file.
Title to add to the created web page	The title that was added to the web page. This published data is only available when Create Web Page is selected on the General tab.
Text to add to the web page	The text that was added to the web page. This value is determined by the Text field in both the Create Web Page and Append to Web Page modes.

Read Text Log

The Read Text Log activity will read lines in a structured text log file. If you have log files that change names, you can configure the Read Text Log activity to read from the newest file in a folder that matches a file name pattern. The Read Text Log activity can be used to check a log for errors and then take corrective action on the server that is creating the log or send an email to an administrator to escalate the issue.



Important

For the Read Text Log activity to work correctly, every line in the text log file must begin with a timestamp.

Configuring the Read Text Log Activity

Before you configure the Read Text Log activity, you need to determine the following:

- The log file name you are reading.
- The timestamps format in the log.

Use the following information to configure the Read Text Log activity.

Details Tab

Settings	Configuration Instructions
File	Select and type the name of the log file that will be read. You can also use the ellipsis () button to browse for the file.
The most recent file in this folder	Select and type the folder where the most recent file that matches the pattern that you specify will be read. You can also use the

Settings	Configuration Instructions
	ellipsis () button to browse for the folder.
Matching this pattern	Type the file name pattern that will be used to find the log file that will be read. You can use the * and ? wildcards to specify the pattern. These wildcards behave in the same way as the Windows Command Prompt.
Read	Select and specify the dates that the lines you are reading are from:
	lines between the dates: Select and specify the begin date and end date that make up the range that will be read. The dates that you specify must match the Timestamp format.
	lines more recent than: Select and specify the oldest date of the logs that will be read. The date that you specify must match the entered Timestamp format.
	new lines : Select to read all the logs that have not previously been read by the Read Text Log activity.
Timestamp format	Specify the format of the timestamp of the logs. For more information on how to specify the timestamp format, see the following Timestamp Format Codes table.
Read the last lines	Enter the number of lines.

Timestamp Format Codes

Code	Description
%y	Year in two digits. For example, in this format '2005' would be represented as '05'.
%Y	Year in four digits.
%m	Month in two digits. For example, in this format 'September' would be represented as '09'.
%d	Day in two digits.
%Н	Hour in two digits in the 24 hour format. For example, in this format '1 pm' would be

Code	Description
	represented as '13'
%M	Minutes in two digits.
%S	Seconds in two digits.
%s	Milliseconds in three digits.

Here are some examples of dates and their corresponding timestamp format.

Date	Format
03/26/2010 14:07:46	%m/%d/%Y %H:%M:%S
[03/26/2010] [14:07:46]	[%m/%d/%Y] [%H:%M:%S
15-11-10 02:09:45:056	%d-%m-%y %H:%M:%S:%s

Published Data

The following table lists the published data items.

Item	Description
Full path and name of the log file	The full path of the log file that is being read.
Number of lines matching the condition	The number of lines that were read.
For each line read	
Full line matching the filter, including timestamp	The entire line as it appears in the log file.
Timestamp of matching line	The timestamp of the line that was read.
Message of matching line	The log message of the line that was read.

Write to Database

The Write to Database activity writes a row into a database table. This activity interacts with the following databases:

- Access
- ODBC
- Oracle
- SQL Server

The Write to Database activity can be used to replicate important Windows Event Log Events to a database table that is able to be queried and maintained.

Configuring the Write to Database Activity

Before you configure the Write to Database activity, you need to determine the following:

- The database you are connecting to.
- The table and fields you are updating.

Use the following information to configure the Write to Database activity.

Details Tab

Settings	Configuration Instructions
Table name	Type the name of the database table that you are adding the row to.
Data	The list displays all the fields in the table that will be set. To add a field, click Add and enter the Field name and Value . To remove a field, select it and click Remove . To edit a field, double-click the field name.

Connection Tab

Settings	Configuration Instructions
Database type	Select the Database type from the drop-down list. The options include the following:
	Access
	ODBC
	Oracle
	SQL Server
	Configuration instructions for each Connection
	tab Database type are listed in the following tables.

Access Connections Tab

Settings	Configuration Instructions
File	Type the name of the Access database file that you want to access.
Workgroup file	Type the name of the Access workgroup file

Settings	Configuration Instructions
	that is associated with this database.
User name	Type the user name for the workgroup file.
Password	Type the password for the workgroup file.
DB password	Type the password for the Access database.

ODBC Connections Tab

Settings	Configuration Instructions
DSN	Enter the data source name.
User name	Enter the user name for this database.
Password	Enter the password for this database.

Oracle Connections Tab

Settings	Configuration Instructions
Service Name	Enter the service name.
User name	Enter the user name for this database.
Password	Enter the password for this database.

SQL Server Connections Tab

Settings	Configuration Instructions
Authentication	Select either Windows Authentication or SQL Server Authentication.
Server	Enter the name of the SQL Server that you want to access.
Initial catalog	Enter the name of the initial catalog. If you selected the SQL Server Authentication option, type the user name and password used to access the SQL Server in the User name and Password boxes.

Timeout Tab

Settings	Configuration Instructions
Timeout	Enter the amount of time that the Query
	Database activity will wait for the database
	operation to complete.
	Set this value to 0 to wait indefinitely.

Security Credentials Tab

Settings	Configuration Instructions
Use the security of the account assigned to the service	Select this option if you want to run the Query Database activity using the same account that the runbook server uses.
This account	Use this option to specify a different account. Enter the User name and Password .
	If you specify an invalid user name or password, the account assigned to the runbook server will be used to run the activity.

Published Data

The following table lists the published data items.

Item	Description
Initial Catalog	The initial catalog that was used when connecting to the database. This published data will only be available when SQL Server is selected on the Connection tab.
Database server	The name of the database server. This published data will only be available when SQL Server is selected on the Connection tab.
Table name	The name of the table that was written to.
Database user	The name of the user used to connect to the database server.
ODBC DSN	The name of the ODBC DSN. This published

Item	Description
	data will only be available when ODBC is selected on the Connection tab.
Oracle Service Name	The service name. This published data will only be available when Oracle is selected on the Connection tab.
Access file	The Access database file that was modified. This published data will only be available when Access is selected on the Connection tab.
Access workgroup information file	The Access workgroup file that is associated with the Access database file. This published data will only be available when Access is selected on the Connection tab.

Query Database

The Query Database activity queries a database and returns the resulting rows as published data. This activity supports the following database types:

- Access
- ODBC
- Oracle
- SQL Server

The Query Database activity can be used to query a database for the detailed description of an error code that has appeared on one of the systems in the data center and then that description is sent to an administrator in an email message.

Configuring the Query Database Activity

Before you configure the Query Database activity you will need to determine the following:

- The database that you are connecting to
- The SQL query that you are running

Use the following information to configure the Query Database activity.

Details

Settings	Configuration Instructions
Query	Type the SQL query in the Query field



Warning

The Query Database activity does not support queries that return data as XML, such as queries that use the FOR XML clause in SQL Server.

Connection

Settings	Configuration Instructions
Database type	Select the Database type from the drop-down list. The options include the following: • Access • ODBC • Oracle • SQL Server



Important

When Orchestrator is installed on a non-English operating system, and you set the Connection for Database type to SQL Server, the Server input value cannot be localhost. You must use the actual computer name.

Configuration instructions for each Connection tab Database type are listed in the following tables.

Access Connections

Settings	Configuration Instructions
File	Type the name of the Access database file that you want to access.
Workgroup file	Type the name of the Access workgroup file that is associated with this database.
User name	Type the user name for the workgroup file.
Password	Type the password for the workgroup file.
DB password	Type the password for the Access database.

ODBC Connections

Settings	Configuration Instructions
DSN	Enter the data source name.

Settings	Configuration Instructions
User name	Enter the user name for this database.
Password	Enter the password for this database.

Oracle Connections

Settings	Configuration Instructions
Service Name	Enter the service name.
User name	Enter the user name for this database.
Password	Enter the password for this database.

SQL Server Connections

Settings	Configuration Instructions
Authentication	Select either Windows Authentication or SQL Server Authentication.
Server	Enter the name of the SQL Server that you want to access.
Initial catalog	Enter the name of the initial catalog. If you selected the SQL Server Authentication option, type the user name and password used to access the SQL Server in the User name and Password boxes.

Timeout

Settings	Configuration Instructions
Timeout	Enter the amount of time that the Query
	Database activity will wait for the database
	operation to complete.
	Set this value to 0 to wait indefinitely.

Security Credentials

Settings	Configuration Instructions
Use the security of the account assigned to	Select this option if you want to run the Query
the service	Database activity using the same account that

Settings	Configuration Instructions
	the runbook server uses.
This account	Use this option to specify a different account. Enter the User name and Password. Note If you specify an invalid user name or password, the account assigned to the runbook server will be used to run the activity.

Published Data

The following table lists the published data items.

Item	Description
Numeric return value of the query	When a query that returns a numeric value is used, this will be the value. For example, "Select COUNT(*) where FirstName=John"
Database query	The database query that was sent to the database.
Initial Catalog	The initial catalog that was used when connecting to the database. This published data will only be available when SQL Server is selected on the Connection tab.
Database server	The name of the database server. This published data will only be available when SQL Server is selected on the Connection tab.
Database user	The name of the user used to connect to the database server.
ODBC DSN	The name of the ODBC DSN. This published data will only be available when ODBC is selected on the Connection tab.
Oracle Service Name	The service name. This published data will only be available when Oracle is selected on the Connection tab.
Access file	The Access database file that was queried. This published data will only be available when

Item	Description
	Access is selected on the Connection tab.
Access workgroup information file	The Access workgroup file that is associated with the Access database file. This published xata will only be available when Access is selected on the Connection tab.
For each row published	
Full line as a string with fields separated by ;	The entire the row that was published with each field in the row separated by a semi-colon (;). Use the Field data manipulation function to obtain the values of a field within the row

Monitor Counter

The Monitor Counter activity invokes a runbook when a counter has reached a value that you specify. Each Monitor Counter activity monitors one counter.

Use the Monitor Counter activity to monitor a counter that counts the number of times that a runbook has attempted to start a service. When that number reaches the number that you configure in the Monitor Counter activity, the Monitor Counter activity can invoke a Send Email activity to notify an administrator to investigate the problem.

Configuring the Monitor Counter Activity

Before you configure the Monitor Counter activity, you need to determine the following:

The **Counter** you will be monitoring.



Warning

Before you can use this activity, you must configure a Counter.

The value that will invoke the runbook

Use the following information to configure the Monitor Counter activity.

Published Data

The following table lists the published data items.

Item	Description
Counter Value	The value of the counter being monitored

Get Counter Value

The Get Counter Value activity retrieves the value of a counter and returns it as a published data item. Wherever you need to use the value of a counter, use the published data that is published by the Get Counter Value to retrieve that value.

Configuring the Get Counter Value Activity

Before you configure the Get Counter Value activity, you need to determine which counter you will retrieve.



Warning

Before you can use this activity, you must configure a Counter. To modify a counter, use the Modify Counter activity

Use the following information to configure the Get Counter Value activity.

Details Tab

Settings	Configuration Instructions
Counter	Click the ellipsis () button to select the Counter that you are retrieving.

Published Data

The following table lists the published data items.

Item	Description
Counter Value	The value of the counter.

Modify Counter

The Modify Counter increments and decrements a counter, as well as resets it to its default value. It also sets it to a value you specify. Wherever you need to update the value of a counter, use the Modify Counter activity to update its value.

The current value of a counter is specific for every runbook that uses that counter. The first time a counter is used, the default value that has been specified in the counters configuration will be used. You can only modify the value of counters in a runbook using the Modify Counter activity.

Configuring the Modify Counter Activity

Before you configure the Modify Counter activity, you need to determine the following:

- The counter you are updating.
- The type of update that will be made.

Use the following information to configure the Modify Counter activity.

Details Tab

Settings	Configuration Instructions
Counter	Click the ellipsis () button to select the Counter that you are retrieving.
Action	Select how you want the value of the counter to be changed:
	Increment: add the Step value to the value of the counter.
	Decrement : subtract the Step value from the value of the counter.
	Set: set the value of the counter to the Step value.
	Reset: reset the value of the counter to the default value.
Value	The value used by the Increment , Decrement , or Set action.

Published Data

The following table lists the published data items.

Item	Description
Counter Value	The value of the counter

Invoke Web Services

The Invoke Web Services activity runs a web service with XML parameters you specify.

Configuring the Invoke Web Services Activity

Before you configure the Invoke Web Services activity, you need to determine the following:

- WSDL file of the web service
- Web service method name
- Input SOAP message body format
- Output SOAP message body format

Use the following information to configure the Invoke Web Services activity.

Details

Settings	Configuration Instructions
WSDL	Type the path of the WSDL file or use the ellipsis () button to browse for the file.
Method	Type the name of the method that you are invoking on the web service, or click the ellipsis button () and browse for it. Make sure that you match the casing of the method.
XML Request Payload	Type the parameters that you are sending to the web service method. Make sure that the format matches what is described in the WSDL document.
Format Hint	Click this button to receive hints on formatting the XML job payload. Replace the placeholder values with your own.

Advanced

Settings	Configuration Instructions
Save	To save the responses, select the Save check box and specify the folder where the responses will be saved.
URL	To specify the URL location of the web service, select the URL check box and type the URL location.
Value	Select the SOAP protocol that the web service uses. The Value options include the following: SOAP 1.1 SOAP 1.2

Security

Settings	Configuration Instructions
Enable	Select the Enable check box to enable HTTP authentication, and fill in the fields.
User name	Type the user name to access the secured web service.
Password	Type the password to access the secured web service.

HTTPS certificate options

Orchestrator allows you to configure HTTPS certificate options in cases where certificate validation fails.

Use the following steps to configure HTTPS certificate options.

To configure HTTPS certificate options

- 1. In the Runbook Designer, click the **Options** menu, and select **Invoke Web Services** to open the **Invoke Web Services** dialog box.
- 2. Configure the settings on the **Details** tab. Configuration instructions are listed in the following table.

Details

Settings	Configuration Instructions
HTTPS Options	Select one of the following HTTPS Options: • Accept all certificates • Accept certificates from trusted hosts Configuration instructions for each of the
	HTTPS Options are listed in the following tables.

Accept all certificates Details

Settings	Configuration Instructions
Accept all certificates	Accepts certificates from all hosts.
	After you select this HTTPS option, click
	Finish.

Accept certificates from trusted hosts Details

Settings	Configuration Instructions
Accept certificates from trusted hosts	Specifies the hosts you want to accept the certificates from.
	Click Add to open the Trusted Host dialog box.
	2. Type the trusted host name in the Value box, and click OK . The host is then added to the list.
	To edit hosts, click Edit .
	To remove hosts, click Remove .

Published Data

The following table lists the published data items.

Item	Description
WSDL Path	The WSDL path.
Method Name	The name of the web method.
XML Job Payload	The text of the XML job payload.
XML Response Payload	The text of the XML response payload.
Response File	The path and filename of the response.
Web Service URL	The URL of the web service.
Web Service protocol	The protocol that the web service uses.

Publishing web services

The Invoke Web Service object builds an assembly at C:\ProgramData\Microsoft System Center 2012\Orchestrator\Activities\WebServices2or

C:\Users\USERNAME\AppData\Local\Microsoft System Center

2012\Orchestrator\Activities\WebServices2. The assembly is identified by the web service location. For example, http://localhost/TestService/DylanService.asmx?WSDL.

If you publish additional services, or update an existing service, you must clean the cache, except for the **wspkey.snk** file. After cleaning the cache, the web service changes are correctly published.

Format Date/Time

The Format Date/Time activity enables you to transform existing date and time formats into custom formats that you create.

Configuring the Format Date/Time Activity

Before you configure the Format Date/Time activity, you need to determine the following:

- The existing date and time format you want to transform.
- The new date and time format you want it to become.

Use the following information to configure the Format Date/Time activity.

Details Tab

Settings	Configuration Instructions
Date/Time	Type the time that you want to convert.
Format	Type the format of the time that you want to convert. See the Date/Time Format Codes table for format codes and examples.
Format	Type the format that you want to convert the input time to.
Output Adjustments	Type a number in any of the following fields to adjust the output time from the input time. For example, if the input time is coming from a server that is three hours ahead of your local time, type -3 in the Hours field to set the output time (your local time) to three hours behind the input time. If the input time is three hours behind your local time, type 3 to set the output time to three hours ahead of the input time.
	You can adjust the output time using the following time units:
	Days
	Months
	Years
	Hours
	Minutes
	Seconds

To specify a date/time format, you must enter the codes that represent each part of the date and time.

Date/Time Format Codes

Code	Description
у	The last digit of the year. For example, 2005 would be represented as 5.
уу	The last two digits of the year number. For example, in this format 2005 would be represented as 05.
уууу	The year number in four digits. For example, in this format 2005 would be represented as 2005.
М	Month as a number from 1 to 12. If the month number is a single-digit number, it is displayed without a leading zero.
ММ	Month in two digits. If the month number is a single-digit number, it is displayed with a leading zero.
MMM	The name of the month in three letters. For example, August would be represented as Aug.
MMMM	The name of the month spelled in full. This format is supported only for output time. Note This format is only supported for the output format.
d	Day as a number from 1 to 31. If the day number is a single-digit number, it is displayed without a leading zero.
dd	Day in two digits. If the day number is a single-digit number, it is displayed with a leading zero.
ddd	The abbreviated name of the day of the week in three letters. For example, Saturday is abbreviated as "Sat".
dddd	The full name of the day of the week. For example, Saturday is displayed in full.

Code	Description
	Note This format is only supported for the output format.
h	Hour as a number from 1 to 12 when using the 12-hour clock. If the hour number is a single-digit number, it is displayed without a leading zero.
hh	Hour in two digits using the 12-hour clock. If the hour number is a single-digit number, it is displayed with a leading zero.
Н	Hour as a number from 0 to 23 when using the 24-hour clock. For example, in this format 1 pm would be represented as 13. If the hour number is a single-digit number, it is displayed without a leading zero.
НН	Hour in two digits using the 24-hour clock. For example, in this format 1 pm would be represented as 13. If the hour number is a single-digit number, it is displayed with a leading zero
m	Minutes as a number from 0 to 59. If the minute number is a single-digit number, it is displayed without a leading zero.
mm	Minutes in two digits. If the minute number is a single-digit number, it is displayed with a leading zero.
s	Seconds as a number from 0 to 59. If the second number is a single-digit number, it is displayed without a leading zero.
ss	Seconds in two digits. If the second number is a single-digit number, it is displayed with a leading zero.
tt	A.M. or P.M. as two letters: A.M. or P.M. as defined on your system.

Here are some examples of dates and times and their corresponding format:

Format	Date
MM/dd/yyyy hh:mm:ss tt	08/05/2006 03:05:15 PM
M/d/yy h:m:s tt	8/5/06 3:5:15 PM
ddd MMM dd yyyy	Sat Aug 05 2006
dddd, MMMM dd yyyy	Saturday, August 05 2006

Published Data

The following table lists the published data items.

Item	Description
Format Result	The result of the format in the specified form.
Format Result without adjustments	The formatted result, but without any adjustments made to it.
Input Time	The input time.
Input Format	The format of the date and time that was entered as the input time.
Output Format	The format of the date and time that was entered as the output time.

Generate Random Text

The Generate Random Text activity generates random strings of text.

Configuring the Generate Random Text Activity

Before you configure the Generate Random Text activity, you need to determine the random text string attributes you want to generate.

Use the following information to configure the Generate Random Text activity.

Details Tab

Settings	Configuration Instructions
Text Length	Type the number of characters that you want

Settings	Configuration Instructions
	the string to include, for example, 45.
Text Contents	Select the options for the items that you want the Generate Random Text activity to include in the random text string. In the Minimum Quantity field for each option that you select, type the minimum number of these characters that you want to include in the string. The total of all Minimum Quantity fields must not be more than the number you typed in the Text Length field.
	Lower-Case Characters
	Upper-Case Characters
	Numbers
	Symbols

Published Data

The following table lists the published data items.

Item	Description
Random text	The string of random text that this activity creates.
Random text length	The length of the text that was generated.

Map Network Path

The Map Network Path activity enables you to map a network path using a UNC path.

Configuring the Map Network Path Activity

Before you configure the Map Network Path activity, you need to determine the following:

- The UNC path you want to map.
- The user account and password you need to log into that path; if required.

Use the following information to configure the Map Network Path activity.

Details Tab

Settings	Configuration Instructions
Network path	Type the network path that you want to connect to in UNC format (\\servername\\foldername), or click the ellipsis button () and browse for it. Verify that the network path that you want to map does not already exist.
User account	Type the user account that you need to access the network path.
Password	Type the password that you need to access the network path.

Published Data

The following table lists the published data items.

Item	Description
Network path	The network path that you are mapping.
User account	The user account that you used to access the network path.

Disconnect Network Path

The Disconnect Network Path activity allows you to disconnect a network path. You can disconnect network paths you mapped using the Map Network Path activity or using another method.

Configuring the Disconnect Network Path Activity

Before you configure the Disconnect Network Path activity, you need to determine the network path you want to disconnect.



You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

Use the following information to configure the Disconnect Network Path activity.

Details Tab

Settings	Configuration Instructions
Network path	Type the name of the network path that you want to disconnect, or click the ellipsis button () and browse for it.

Published Data

The following table lists the published data items.

Item	Description
Network path	The network path you are disconnecting.

Get Dial-up Status

The Get Dial-up Status activity retrieves the status of a dial-up or VPN network connection on the Runbook server. For more information on creating a network connection in Windows Server 2008, see Establish Network Connections

Configuring the Get Dial-up Status Activity

Use the following information to configure the Get Dial-up Status activity.



Note

You cannot set individual security credentials for this activity. It will run under the service account configured for the Runbook Service on the Runbook server where the instance of the activity is running. This account must have the authority to access the resources and perform the actions required by this activity.

Connection Tab

Settings	Configuration Instructions
Dial-up or VPN entry	Type the name of the entry as entered in the address book, or click the ellipsis () button and select the entry from the Remote Access Phone Book.

The following table lists the published data items.

Item	Description
Dial-up or VPN name	The name assigned to the dial-up connection
Line status	Indicates whether the network connection is connected or disconnected

Connect/Disconnect Dial-up

The Connect/Disconnect Dial-up activity connects or disconnects a dial-up connection or VPN. The connection must be configured on the Runbook server before the activity can use it. For more information on creating a network connection in Windows Server 2008, see <u>Establish</u> Network Connections

Configuring the Connect/Disconnect Dial-up Activity

Use the following information to configure the Connect/Disconnect Dial-up activity.

Connection Tab

Settings	Configuration Instructions
Dial-up or VPN entry	Type the name of the entry as entered in the address book, or click the ellipsis () button and select the entry from the Remote Access Phone Book.
Connect/Disconnect	Select whether to connect to or disconnect from the dial-up connection or VPN
Attempts	Enter the number of times the activity should attempt to connect to the remote network before quitting.
Delay	Enter the amount of time, in seconds, that the activity should wait between retry attempts.

Authentication Tab

Settings	Configuration Instructions
Domain	Enter the name of the domain for the username.
Username	Enter the username to logon to the remote network.
Password	Enter the password for the username.

Published Data

The following table lists the published data items.

Item	Description
Dial-up or VPN name	The name assigned of the network connection
Number of retries attempted	Indicates how many times the activity attempted to establish the connection before succeeding or failing.
Domain name credential	The domain name used by the activity when establishing a connection
User name credential	The user name used by the activity when establishing a connection

Text File Management

The following table provides a brief description of tasks you can accomplish when using each Text File Management activity.



Caution

If permissions on the Orchestrator installation path are changed and the activity's Security Credentials has a custom user account that does not include Read/Execute permissions to ExecutionData.dll on the Runbook server, the activity will fail.

Tasks	Text File Management Activities
Append a line of text into a text file.	Append Line
Delete lines from a text file.	Delete Line

Tasks	Text File Management Activities
Find lines in a text file.	Find Text
Get multiple lines from a text file.	Get Lines
Insert lines into a text file on a line number you specify.	Insert Line
Read lines from a text file.	Read Line
Search for and replaces text in a file.	Search and Replace Text

Append Line

The Append Line activity appends a line of text into a text file. Use the Append Line activity to append lines to a log file to create audits trails of runbooks.

This activity replaces functionality in the Manage Text File legacy activity from Opalis 6.3.

Configuring the Append Line Activity

Before you configure the Append Line activity, you need to determine the following:

- The file name you want to append to.
- The type of file encoding that the file you are appending to uses.
- · Text you append.

Use the following information to configure the Append Line activity.

Details Tab

Settings	Configuration Instructions
File	Type the path and name of the file that you want to append the text to, or click the ellipsis button () and browse for it.
File encoding	Click the ellipsis button () and select the format that the file is encoded in from the File encoding drop-down list. Verify that you select the correct encoding format. If the file uses a different encoding format, the activity fails.
Text	Type the text that you want to append to the file

Settings	Configuration Instructions
	that you specified.

The following table lists the published data items.

Item	Description
File path	The path and file name of the text file to which the line is appended.
File encoding	The file encoding format that you selected in the File encoding field.
Line text	The text of the line that was appended to the text file.
Line number	The line number where the text was appended.

Delete Line

The Delete Line activity deletes lines from a text file. Use the Delete Line activity to delete outdated lines of text from a text file.

This activity replaces functionality in the Manage Text File legacy activity from Opalis 6.3.

Configuring the Delete Line Activity

Before you configure the Delete Line activity you need to determine the following:

- The name of the file that you want to delete the line from.
- The file encoding type that the file you want to delete the line from uses.
- The line numbers of the lines that you want to delete.

Use the following information to configure the Delete Line activity.

Details Tab

Settings	Configuration Instructions
File	Type the path and name of the file that you want to delete the text from, or click the ellipsis button () and browse for it.

Settings	Configuration Instructions
File encoding	Click the ellipsis button () and select the format that the file is encoded in from the File encoding drop-down list. Verify that you select the correct encoding format. If the file uses a different encoding format, the activity fails.
Line numbers	Type the line numbers of the text that you want to delete from the file that you specified.
	To specify a range of lines, use a hyphen: 1-3. This deletes lines 1 to 3.
	To specify specific lines, use a comma: 5,7,9. This deletes lines 5, 7, and 9.
	Combine the range and specific lines: 1-3,5,7,9. This deletes lines 1 to 3, and lines 5, 7, and 9.
	To specify from a specific line to the last line of the file, type the line number, hyphen, and END: 4-END. This deletes lines 4 to the last line of the file.
	To specify from a specific line to a line relative to the last line of the file, type the line number, hyphen, the less-than sign, and the line number relative to the end line: 4-END<3. If the file has 20 lines, this deletes lines 4 to 17 from the file. <3 represents the third line from the end.
	To specify the last number of lines, type LASTLINES, colon, and the last number of lines that you want to delete: LASTLINES:10. This deletes the last 10 lines of the file.
	Combine different types of operations: 1-5, 8, 10-END<20, LASTLINES:10. This deletes lines 1 to 5, line 8, line 10 to the 20th line from the end, and the last 10 lines. Do not overlap lines or line ranges when combining operations. For example, 5-END, LASTLINES:10 fails because the 5-END operation already deletes to the end, so the LASTLINES:10 operation cannot succeed because the lines are already deleted, and the activity fails.
	Important Do not specify lines numbers that do

Settings	Configuration Instructions
	not exist in the file, and do not specify a line number more than once, or the activity fails.

The following table lists the published data items.

Item	Description
File name	The file name of the text file from which the line is deleted.
File encoding	The file encoding format that you selected in the File encoding field.
Line number	The line number of each line that was deleted. A Published Data item is created for each line that was deleted.
Line numbers	The line number range that the user typed in the field.
Deleted text	The text that was deleted from the file.
Number of deletions	The number of deletions that occurred.

Find Text

The Find Text activity finds lines in a text file. Use the Find Text activity to find according to a search string that you specify.

This activity replaces functionality in the Manage Text File legacy activity from Opalis 6.3.

Configuring the Find Text Activity

Before you configure the Find Text activity, you need to determine the following:

- The name of the file that you want to search in.
- The encoding type of the file you want to search in uses.
- The text that you want to search for.

Use the following information to configure the Find Text activity.

Details Tab

Settings	Configuration Instructions
File	Type the path and name of the file that you want to find the text in, or click the ellipsis button () and browse for it.
File encoding	Click the ellipsis button () and select the format that the file is encoded in from the File encoding drop-down list. Verify that you select the correct encoding format: if the file uses a different encoding format, the activity fails.
Search text	Type the text that you are searching for in the file.
Case sensitive	Select this option to search only for lines where the case of the words matches the text from the Search text field exactly.
Use regular expressions	Select this option to use regular expressions in your search. For more information on regular expressions, see Using Regular Expressions .
Result	Select one of the following options for your results:
	Only the first line that matches the text will be returned
	All lines that match the text will be returned

Published Data

The following table lists the published data items.

Item	Description
Case sensitive	Indicates whether the Case sensitive check box was selected.
File encoding	The file encoding format that you selected in the File encoding field.
File name	The name of the file that was searched for text.
Return first line or all lines	Indicates whether the Only the first line that matches the text will be published or All lines that match the text will be published option was

Item	Description
	selected.
Line number of match	The line number where matching text was found.
Match end	The character offset position that the match ends on.
Match start	The character offset position that the match starts on.
Matched text	The text that matched the search string.
Number of lines matched	The number of lines where matching text was found.
Number of matches	The number of matching items that were found.
Original line	The entire line that contains the matching item.
Search text	The search string that was used for the search.
Use Regex	Indicates whether the Use regular expressions check box was selected.

Get Lines

The Get Lines activity gets multiple lines from a text file according to criteria that you specify. You can use the Get Lines activity to get specific lines from any location in a text file.

This activity replaces functionality in the Manage Text File legacy activity from Opalis 6.3.

Configuring the Get Lines Activity

Before you configure the Get Lines activity, you need to determine the following:

- The name of the file you want to get lines from.
- The encoding type that the file you want to get the lines from uses.
- The criteria you use to filter the lines.

Use the following information to configure the Get Lines activity.

Details Tab

Settings	Configuration Instructions
File	Type the path and name of the file that you
	want to get the text from, or click the ellipsis

Settings	Configuration Instructions
	button () and browse for it.
File encoding	Click the ellipsis button () and select the format that the file is encoded in from the File encoding drop-down list. Verify that you select the correct encoding format. If the file uses a different encoding format, the activity fails.
Lines	Click Add to open the Add Line dialog box and create filters for the lines that you want to get from the file: Name: Search for lines by their name. Range: Search for lines by their range.

The following table lists the published data items.

Item	Description
File path	The filename and path of the file that the lines were taken from.
File encoding	The file encoding format that you selected in the File encoding field.
#Name# line text	For each item that you add in the lines list of the dialog, a new published data item is created. This item displays the line text of each item in the Lines list. #Name# represents the name that you assigned in the Name field.
#Name# line numbers	For each item that you add in the Lines list of the dialog, a new published data item is created. This item displays the line numbers where text was found from each item in the Lines list. #Name# represents the name that you assigned in the Name field.
Total Number of Lines in the Ranges Specified	The total number of lines that were found in the ranges that were specified.

Insert Line

The Insert Line activity inserts lines into a text file on a line number that you specify.

This activity replaces functionality in the Manage Text File legacy activity from Opalis 6.3.

Configuring the Insert Line Activity

Before you configure the Insert Line File activity, you need to determine the following:

- The name of the file you want to insert text into.
- The file encoding type of the file you want to insert text into.
- The line number location where you want to insert the text.

Use the following information to configure the Insert Line activity.

Details Tab

Settings	Configuration Instructions
File	Type the path and name of the file that you want to insert the text into, or click the ellipsis button () and browse for it.
File encoding	Click the ellipsis button () and select the format that the file is encoded in from the File encoding drop-down list. Verify that you select the correct encoding format. If the file uses a different encoding format, the activity fails.
Text	Type the text that you want to insert into the file.
Line number	Type the line number where the text will be inserted.

Published Data

The following table lists the published data items.

Item	Description
File name	The name of the file that the text was inserted into.
File encoding	The file encoding format that you selected in the File encoding field.

Item	Description
Line text	The text of the line that was inserted.
Line number	The line number that was inserted, if only one line was inserted.

Read Line

The Read Line activity reads lines from a text file. You can use the Read Line activity to read lines from a text file and pass them to another activity using published data.

This activity replaces functionality in the Manage Text File legacy activity from Opalis 6.3.

Configuring the Read Line Activity

Use the following information to configure the Read Line activity.

Details Tab

Settings	Configuration Instructions
File	Type the path and name of the file that you want to read the text from, or click the ellipsis button () and browse for it.
File encoding	Click the ellipsis button () and select the format that the file is encoded in from the File encoding drop-down list. Verify that you select the correct encoding format. If the file uses a different encoding format, the activity fails.
Line numbers	 Type the line numbers of the text that you want to read from the file that you specified. To specify a range of lines, use a hyphen: 1-3. This reads lines 1 to 3. To specify specific lines, use a comma: 5,7,9. This reads lines 5, 7, and 9. Combine the range and specific lines: 1-3,5,7,9. This reads lines 1 to 3, and lines 5, 7, and 9. To specify from a specific line to the last line of the file, type the line number, hyphen, and END: 4-END. This reads lines 4 to the last line of the file.

Settings	Configuration Instructions
	To specify from a specific line to a line relative to the last line of the file, type the line number, hyphen, the less-than sign, and the line number relative to the end line: 4-END<3. If the file has 20 lines, this reads lines 4 to 17 from the file. <3 represents the third line from the end.
	 To specify the last number of lines, type LASTLINES, colon, and the last number of lines that you want to delete: LASTLINES:10. This reads the last 10 lines of the file.
	Combine different types of operations: 1-5, 8, 10-END<20, LASTLINES:10. This reads lines 1 to 5, line 8, line 10 to the 20th line from the end, and the last 10 lines. Do not overlap lines or line ranges when combining operations. For example, 5-END, LASTLINES:10 fails because the 5-END operation already reads to the end, so the LASTLINES:10 operation cannot succeed because the lines are already read, and the activity fails.
	Important Do not specify lines numbers that do not exist in the file, and do not specify a line number more than once, or the activity will fail.

The following table lists the published data items.

Item	Description
File name	The file name of the text file that was read.
File encoding	The file encoding format that you selected in the File encoding field.
Line text	The text of the line that was read.

Item	Description
Line number	The line number of the text that was read. A published data item is created for each line that was read.
Line numbers	The line number range that the user typed in the field.

Search and Replace Text

The Search and Replace Text activity searches for and replaces text that you specify in a text file. This activity replaces functionality in the Manage Text File legacy activity from Opalis 6.3.

Configuring the Search and Replace Text Activity

Before you configure the Search and Replace Text activity, you need to determine the following:

- The file name you want to search in.
- The encoding that the file you want to search in uses.
- The text you want to search for.
- The replacement text

Use the following information to configure the Search and Replace Text activity.

Details Tab

Settings	Configuration Instructions
File	Type the path and name of the file that you want to read the text from, or click the ellipsis button () and browse for it.
File encoding	Click the ellipsis button () to open the File Encoding dialog box and select the format that the file is encoded in from the File Encoding drop-down list. Verify that you select the correct encoding format: if the file uses a different encoding format, the activity fails.
Search text	Type the text that you are searching for in the file.
Case sensitive	Select this option to search only for lines where the case of the words matches the text from the

Settings	Configuration Instructions
	Search text field exactly.
Use regular expressions	Select this option to use regular expressions in your search. For more information, see Using Regular Expressions .
Replacement text	Type the text that you want to replace the search text with.

The following table lists the published data items.

Item	Description
Case sensitive	Indicates whether the Case sensitive check box was checked or not.
File encoding	The file encoding format that you selected in the File encoding field.
File name	The name of the file that was searched for text.
Line number of match	The line number where matching text was found.
Modified line	The entire line of text as it was written after the replace operations occurred.
Number of lines matched	The number of lines where matching text was found.
Number of matches	The number of matching items that were found.
Original line	The entire line of text as it was written before the replace operation occurred.
Replace text	The text that was used to replace the search text.
Search text	The search string that was used for the search.
Use Regex	Indicates whether the Use regular expressions check box was checked or not.

Runbook Control

The following table provides a brief description of tasks you can accomplish with each activity.

Tasks	Activity
Create a starting point in your runbook.	Initialize Data
Return data from your runbook to another runbook or to an external system.	Return Data
Publish data from any branch.	<u>Junction</u>
Run a runbook.	Invoke Runbook

See Also

Standard Activities

Invoke Runbook

The Invoke Runbook activity launches a runbook that you have specified. You can transfer data to runbooks by configuring an <u>Initialize Data</u> activity in the invoked runbook. You can return data from the invoked runbook by configuring a <u>Return Data</u> activity.

You can use the Invoke Runbook activity to invoke generic runbooks that only perform specific actions that do not depend on how the runbook is invoked. For example, you can create a runbook that calls separate runbooks to perform a backup maintenance procedure that in turn calls a runbook to shut down services, another runbook to back up data, and then a final runbook to restart the services.



Important

If you modify the folder name or location of a runbook, you must also re-configure any Invoke Runbook activity that references the modified runbook.

Configuring the Invoke Runbook activity

Before you configure the Invoke Runbook activity, you need to know which runbook you are invoking.

Use the following information to configure the Invoke Runbook activity.

Details

Settings	Configuration Instructions
Runbook	Click the ellipsis () button to browse for the runbook that you want to invoke.
	Important Do not invoke a runbook that starts with a Monitor activity.
Invoke by path	Select to force the Invoke Runbook activity to invoke the runbook by the specific path and name. When selected, any runbook with the same name in the same folder location is invoked. When unselected, the runbook that is invoked can be moved around the environment and the Invoke Runbook activity automatically maps itself to the new location.
Wait for completion	Select to force the Invoke Runbook activity to keep the invoked runbook running until it is completed.
	Important Do not select Wait for completion if any return data in the invoked runbook is also return data in the invoking runbook.
Parameters	If you have selected a runbook that contains an Initialize Data activity, the list of parameters required to invoke that activity will be displayed. Enter a value for each parameter.
Runbook Servers	Type the list of runbook servers that will run this runbook. Separate each name with a semicolon (;). The order in which the runbook servers are listed will be the order used for failover and load balancing of the runbook. The runbook server names must correspond to the names that are displayed within the runbook server's tree in the Orchestrator Deployment Manager. Leave this field blank to use the runbook or global defaults for the runbook

Settings	Configuration Instructions
	server assignment.

The following table lists the published data items from the Invoke Runbook activity.

Item	Description
Child runbook Job ID	The job ID of the invoked runbook.
Child runbook status	The status published by the child runbook.

The Invoke Runbook activity returns any data that the invoked runbook has defined in the Returned Data tab of the runbook properties. The values of these properties must be populated using Return Data activity in that workflow. If the current runbook needs to return data from the invoked runbook, then it must have its own Return Data activity that includes these values.

Credentials

If you use the Invoke Runbook activity and you use <u>Security Credentials</u>, the account you use must be a member of the Orchestrator System group to run successfully.

See Also

Initialize Data
Return Data
Security Credentials

Initialize Data

The Initialize Data activity is a starting point for runbooks that require parameters from an Invoke Runbook activity. The Initialize Data activity is invoked by an Invoke Runbook activity. You can use the Initialize Data activity to launch generic runbooks that only perform specific actions. For example, use the Initialize Data activity to specify the files to back up in a runbook that performs backup operations. To return data to the invoking runbook, end the runbook's workflow with a Return Data activity.

Configuring the Initialize Data activity

Before you configure the Initialize Data activity, you need to know the parameters that you want to use within your runbook.

Use the following information to configure the Initialize Data activity.

Published Data

Each parameter that you have configured is available as published data to the other activities in the runbook while the runbook is running. To pass data back to the invoking runbook, use the Return Data activity.

See Also

Invoke Runbook Return Data

Junction

The Junction activity allows you to wait for multiple branches in a runbook to complete before continuing past the junction. This activity can also publish data again from any branch so that downstream activities past the Junction activity can consume the data. Data from different branches than the one you selected will not be available.

You can choose to propagate no data from any of the branches previous to the Junction activity. When you select an activity, the junction runs once, regardless of the data provided in previous activities. For example, a Monitor File activity waits for files to be added to a folder. When the files are added, two branches in the runbook will copy the file to a new location and at the same time, read the lines of the files and add them to master file. The Junction activity waits for all these to complete and then propagates the data from the Copy File branch and the Delete File activity will delete the original files.

Configuring the Junction activity

Before you configure the Junction activity, you need to determine which branch will continue on the runbook you are invoking.

Use the following information to configure the Junction activity.

Details Tab

Settings	Configuration Instructions
Return data from	Click the ellipsis () button and select the
	activity whose data you want to publish again to

Settings	Configuration Instructions
	the activities that follow the junction. From the Select an Activity dialog box, choose <none></none> to propagate no data to the activities following
	the junction.

The following table lists the data items published by this activity.

Item	Description
Selected branch	The activity that was selected to have its data published.

See Also

Standard Activities

Return Data

The Return Data activity allows you to return data from the current runbook to a runbook that invoked the runbook. You configure the runbook data by configuring the data parameters in the **Runbook Properties** dialog box.

Configuring the Return Data activity

Use the following information to configure the Return Data activity.

Published Data

The available published data items depend on the defined data elements.

See Also

Invoke Runbook
Initialize Data

Service Reporting in System Center 2012 R2

Service Reporting in System Center 2012 R2 enables administrators at IT hosting providers to view tenant consumption of virtual machines, resources (computation, network, and storage), and operating system inventory in their infrastructure.

Service reporting topics

Getting Started with Service Reporting

Provides information about what is new in Service Reporting, an overview of Service Reporting features, and high-level tasks that you can accomplish by using Service Reporting.

Support for Service Reporting

Contains release notes that describe issues that affect Service Reporting, and provides system requirements that you should be aware of before you install Service Reporting.

Planning for Service Reporting

Provides planning information that you need to know before you install Service Reporting.

Deploying Service Reporting

Provides information about how to install Service Reporting and how to configure it afterward.

Operating Service Reporting

Provides information about data that is gathered from source systems and how to configure those source systems, the day-to-day tasks that you accomplish to view and use reports, and how to back up and restore Service Reporting databases.

Getting Started with Service Reporting

Getting started topics

What's New in System Center 2012 R2 - Service Reporting
 Provides information about new features in Service Reporting.

Overview of Service Reporting

Provides a summary of Service Reporting features, explains how IT hosting organizations can use it, and describes the benefits of using it.

Other resources for this component

- Service Reporting in System Center 2012 R2
- Support for Service Reporting
- Planning for Service Reporting
- Deploying Service Reporting

Operating Service Reporting

What's New in System Center 2012 R2 - Service Reporting

Service Reporting is new in System Center 2012 R2, and its capabilities are outlined here. For details about known issues, read the <u>Release Notes for System Center 2012 R2 - Service Reporting</u>.

By using Service Reporting, hosting providers can retrieve computation resource usage and operating system inventory for tenants. Hosting providers can arrange with billing solution partners to use information from Service Reporting to account for expenses based on usage.

Setup

The setup program for Service Reporting installs the component and creates the Service Reporting data warehouse.

Data sources

Service Reporting obtains computation resource usage and inventory data from System Center 2012 – Operations Manager and from Windows Azure Pack for Windows Server. After you run the setup program, you edit and run Windows PowerShell® scripts to configure Service Reporting to retrieve data from source systems.

Usage and inventory reports

Information about computation resource usage and operating system inventory is available in a Microsoft Excel report that you can view and update. The report's data connection to the data cubes in the Service Reporting data warehouse provides the means to query for data in real time. Because the report uses data cubes, you can modify the report to display any fact, dimension, and measure that you want.

Microsoft Update

During Service Reporting setup, you can choose to enable Microsoft Update if you want to automatically update Service Reporting when an update is available.

See Also

Getting Started with Service Reporting
Service Reporting in System Center 2012 R2

Overview of Service Reporting

Administrators for hardware and server operating systems and hosting providers are often asked to produce detailed reports about the consumption of virtual machine resources and to do billing for their tenants. Collecting the data to create the reports typically requires using many disparate tools, and determining the allocation and consumption of tenant resources is often time-consuming. Internally developed custom tools often do not satisfy all the needs of a hosting provider.

When collected and reported data is inconsistent, a hosting provider's revenue can be underreported and tenants might be overbilled. Providing incorrect consumption information can damage the profitability of a business and harm relationships with tenants.

A hosting provider can use the reporting information in Service Reporting to deliver transparent and accurate usage data to managers and tenants.

Benefits

After you deploy the Windows Server 2012 operating system, Windows Azure Pack for Windows Server, and System Center 2012 R2, you have all the tools that you need to understand your tenants' consumption of virtual machine resources and Windows Azure Pack services. Then, you can generate reports that you can use to:

- Create detailed views, for each tenant, of the consumption of the computational, memory, storage, and networking resources for virtual machines and for Windows Azure Pack services.
- Add accuracy in the billing process and reduce underreporting.
- Reduce time and costs of managing and developing a usage tracking system.
- Help your business offer services to tenants that other businesses might not offer.

By using Service Reporting, you can easily view and run monthly reports to share with tenants. You can provide detailed information to managers. You can emphasize areas where incorrect or incomplete tracking data contributes to revenue loss, and call attention to areas that are most profitable.

Architecture

Service Reporting is installed as an optional component of System Center 2012 R2. You can install it by using the initial setup page for System Center 2012 R2 Orchestrator. After you install Service Reporting, you can use Windows PowerShell® scripts to configure it. The scripts establish a connection from the Service Reporting data warehouse to the Windows Azure Pack Usage endpoint and to System Center 2012 – Operations Manager. However, Virtual Machine Manager data from Operations Manager is available only when its agents monitor System Center 2012 R2 Virtual Machine Manager.

After Service Reporting is configured, users open the sample Microsoft Excel files and modify the connection information to point to the Microsoft SQL Server Analysis Services database that is part of the Service Reporting data warehouse.

The Service Reporting data warehouse gathers information from the following sources:

- Windows Azure Pack for usage data about its VM clouds
- Operations Manager agents for information about fabric capacity data, which is gathered from System Center 2012 R2 Virtual Machine Manager

However, the inventory component of Service Reporting does not depend on Windows Azure Pack, so Service Reporting can provide inventory reporting for the environment even if Windows Azure Pack is not installed.

SSIS jobs

The SQL Server Agent job in Service Reporting schedules SQL Server Integration Services (SSIS) package execution periodically. It gathers data from Windows Azure Pack and then processes the data. Then, it gathers data from Operations Manager and processes that data. Occasionally, you must use Windows PowerShell cmdlets to change credentials, such as passwords, that the SSIS jobs use.

Reports

The sample reports highlight the capability of online analytical processing (OLAP) data cubes and show usage and capacity data from virtual machines. These reports also show an inventory of operating systems that tenants use. Service Reporting collects data for the reports hourly and stores the data until database grooming occurs.

The sample reports are stored in a file share when Service Reporting is installed.

See Also

Getting Started with Service Reporting
Service Reporting in System Center 2012 R2

Support for Service Reporting

Support topics

- Release Notes for System Center 2012 R2 Service Reporting
 Contains information about issues that you should be aware of before you install Service Reporting.
- System Requirements for Service Reporting
 Describes the hardware and software requirements for Service Reporting.

Troubleshooting Service Reporting

Contains information that you can use to troubleshoot common problems with Service Reporting.

Other resources for this component

- Service Reporting in System Center 2012 R2
- Getting Started with Service Reporting
- Planning for Service Reporting
- Deploying Service Reporting
- Operating Service Reporting

Release Notes for System Center 2012 R2 - Service Reporting

These release notes contain information that you require to successfully install and use Service Reporting in System Center 2012 R2. They contain information that is not available in the product documentation.

Known issues

Setup requires a domain account for SQL Server Integration Services

Description: If you have configured the Log On As account for Microsoft SQL Server Integration Services by using a domain account, Service Reporting Setup cannot determine that the service is installed, and Setup cannot continue.

Workaround: Update the Log On As account for SQL Server Integration Services by using Local System temporarily, to allow Setup to proceed. Then, after Setup finishes, update the service by using a domain account.

Setup prerequisite checker does not check again

Description: During setup, the Service Reporting Installation wizard **Check prerequisites again** option does not work properly. For example, if you check prerequisites and Microsoft SQL Server 2012 Analysis Management Objects is not found, and then you minimize Setup, install Analysis Management Objects, and check prerequisites again, Setup erroneously does not detect it.

Workaround: To avoid this situation, exit Setup when it fails to detect prerequisites. Correct the deficiency, and then run Setup again.

See Also

Service Reporting in System Center 2012 R2 Support for Service Reporting

System Requirements for Service Reporting

This topic lists only the system requirements for Service Reporting in System Center 2012 R2. Because Service Reporting is a component of System Center 2012 R2, you must also be aware of the requirements for System Center 2012 R2, which are located at System Requirements for System Center 2012 R2.

Hardware

Service Reporting Setup requires 100 gigabytes (GB) of free disk space. After installation, the majority of disk space that Service Reporting uses is reserved for the data warehouse databases. Initially, the size of the databases is small. After data is collected over time, the databases grows but usually does not exceed 60 GB. However, the maximum possible total size of the databases is 100 GB.



Note

The data warehouse online analytical processing (OLAP) instance in Service Reporting can use an extensive amount of RAM.

Software

The setup program verifies that the software is installed on the server where you install Service Reporting. If any prerequisites are missing, Setup cannot finish successfully.

Operating systems

Service Reporting supports the following server operating systems:

- Windows Server® 2012 R2
 - Service Reporting supports Enterprise and Standard editions, although we recommend the Enterprise edition. Regardless of the edition that you use, you can use a Server Core installation.
- Windows Server 2012
 - Service Reporting supports all editions, although we recommend the Enterprise edition. Regardless of the edition that you use, you can use a Server Core installation.

Running Service Reporting on Windows Azure virtual machines

Service Reporting runs on Windows Azure just as it does on physical computer systems. It is recommended on Windows Azure for the primary purpose of Service Reporting, which is the collection and reporting of usage and inventory data gathered from System Center 2012 R2 Operations Manager and Windows Azure Pack for Windows Server endpoints.

Service Reporting was tested by Microsoft by installing and using it in a Windows Azure virtual machine. The testing concluded that Service Reporting was fully functional by it operating exactly the same as on physical hardware.

Stability and performance benchmarks inside a Windows Azure virtual machine were at a level where no special considerations were needed.

System Center 2012 R2 components and Windows Azure Pack requirements

Service Reporting can gather usage information from System Center 2012 R2 Operations Manager and from Windows Azure Pack for Windows Server. However, other components and systems can interact with Operations Manager and from Windows Azure Pack.

Service Reporting requires information from Operations Manager that is available only when agents monitor System Center 2012 R2 Virtual Machine Manager (VMM). For Service Reporting to gather and show VMM information from Windows Azure Pack, Service Reporting must have a connection to Operations Manager.

The following list summarizes the relationships that Service Reporting has with other components and Windows Azure Pack. It also prescribes the order in which you should install and configure System Center 2012 R2 components and Windows Azure Pack before you register data connections from either Operations Manager or Windows Azure Pack to Service Reporting.

- Virtual Machine Manager
 - VMM is the primary source of usage data for hosted servers. In a test environment, you should install and configure VMM before other System Center components.
- 2. Operations Manager

Operations Manager agents monitor and collect data from computers in your server infrastructure. Of primary interest to Service Reporting is that Operations Manager uses agents to gather usage data for VMM fabric and store it in the operational database where Service Reporting can retrieve it. Therefore, you should install and configure Operations Manager and deploy agents after you install VMM. You can read more about using agents to collect VMM data at Operations Manager Agent Installation Methods. After you install VMM and Operations Manager, you can read more about how to connect them at How to Connect VMM with Operations Manager.

Later, after you install Service Reporting, you create the connection to the Operations Manager operational database by using the information at How to Configure Service Reporting for Windows Azure Pack and System Center.

3. Service Provider Foundation

Service Provider Foundation is a component of System Center 2012 R2, and it provides an extensible OData web service that interacts with VMM. This enables service providers and hosting providers to design and implement multitenant self-service portals that integrate IT as a service (laaS) capabilities that are available in System Center 2012 R2. You can read more about installing Service Provider Foundation at How to Install Service Provider Foundation 2012 R2.



Note

Service Provider Foundation Setup is a part of System Center 2012 R2 Orchestrator.

4. Windows Azure Pack for Windows Server

You do not need Windows Azure Pack to view information about inventory reporting or virtual machine fabric from Operations Manager. However, to view usage data for virtual machines, you can use Windows Azure Pack as an optional, but important, source of usage data about hosted services that Service Reporting can report on. You can read more about installing and configuring Windows Azure Pack at Deploy Windows Azure Pack for Windows Server. Later, after you install Service Reporting, you create the connection to the Windows Azure Pack Management database by using the information at How to Configure Service Reporting

for Windows Azure Pack and System Center. After you install and configure all the System Center 2012 R2 components and Windows Azure

SQL Server

Service Reporting supports the following versions of Microsoft SQL Server software.

Pack requirements, you have all the systems in place for Service Reporting installation.



Note

Service Reporting supports the Standard edition for Microsoft SQL Server 2012 and SQL Server 2008 R2, and that edition provides acceptable performance behavior. However, some of the advanced features of SQL Server are constrained in the Standard edition.

- SQL Server 2012 Enterprise and Standard editions. Configuration requirements are as
 - You must configure SQL Server services by using a domain account to ensure that SQL Server Analysis Services, SQL Server (MSSQLSERVER), and SQL Server Agent (MSSQLSERVER) start by using the domain account credentials. You cannot use Local System for SQL Server services with Service Reporting because a remote server that is running SQL Server cannot access the InstallationDrive:\Program Files\Microsoft System Center 2012 R2\Service Reporting\SSISPackages folder share. The account must have read access to the Operations Manager database for retrieval of source data.
 - Service Reporting supports SQL Server 2012 AlwaysOn availability groups. For more information about SQL Server 2012 AlwaysOn failover clustering, see SQL Server 2012 AlwaysOn: Multisite Failover Cluster Instance.
 - Service Reporting requires the SQL Server Integration Services (SSIS) role on the server or servers where you want to install the Service Reporting data warehouse databases.

During Service Reporting installation, respective SSIS packages are created in the servers. These packages use functions that SSIS provides for processing in the data for the Service Reporting data warehouse.

- Service Reporting requires the SQL Server Analysis Services role on the server or servers where you want to install the Service Reporting OLAP databases. During Service Reporting installation, a SQL Server instance is created that hosts the SQL Server role.
- Service Reporting requires SQL Server 2012 Analysis Management Objects on the server where you are installing Service Reporting.
- SQL Server 2008 R2 Enterprise and Standard editions. SQL Server 2008 R2 has the same configuration requirements as SQL Server 2012.

Account requirements for SQL Server services

Service Reporting uses the following accounts for SQL Server services. If you ever modify the accounts for the services, ensure that you preserve or restore permission to the objects.

- SQL Server
 - The service must log on as a domain account.
- SQL Server Agent

The service must log on as a domain account, and it must have read access to the System Center 2012 R2 Operations Manager database. By default, this account is configured with full permission to the SSISPackages and WorkDir folders. If the account is modified, you must ensure that permissions to the folders are preserved.

- SQL Server Analysis Services
 - The service must log on as a domain account, and it must have access permission to the UsageDatawarehouseDB database of the Service Reporting data warehouse.
- SQL Server Integration Services
 - The service must log on as a domain account.

Data warehouse SSIS instance in Service Reporting

During Service Reporting installation, the instance of the Service Reporting data warehouse is created with the following configuration (shown with minimal hardware):

- 8 GB of RAM
- 2 CPUs
- Databases:
 - UsageETLRepositoryDB. Initially configured with 10 GB of disk space without a file group.
 - UsageStagingDB. Initially configured with 100 GB of disk space without a file group.
 - UsageDWRepositoryDB. Initially configured with 1 GB of disk space with eight file groups. However, SQL Server Standard Edition does not include file groups.

Data warehouse OLAP instance in Service Reporting

During Service Reporting installation, the SQL Server Analysis Services instance of the Service Reporting data warehouse is created with the following configuration (shown with minimal hardware):

- 16 GB of RAM
- 2 CPUs
- UsageAnalysisDB database, in a size that is determined by the data volume in the data warehouse



This SQL Server instance can use an extensive amount of RAM.

SQL Server language support in Service Reporting

The information in the following table represents the approved collations and the locale identifiers that Microsoft tested for Service Reporting. In the list of collations in this table, "CI" indicates case-insensitive, and "AS" indicates accent-sensitive.

Language	Windows collation
English, Dutch, German, Italian	Latin1_General_100_CI_AS
Chinese	Chinese_Simplified_Pinyin_100_CI_AS
Chinese Traditional	Chinese_Traditional_Stroke_Count_100_CI_AS
Czech	Czech_100_CI_AS
Danish	Danish_Norwegian_CI_AS
Finnish	Finnish_Swedish_100_CI_AS
French	French_100_CI_AS
Greek	Greek_100_CI_AS
Hungarian	Hungarian_100_CI_AS
Japanese	Japanese_XJIS_100_CI_AS
Korean	Korean_100_CI_AS
Norwegian	Norwegian_100_CI_AS
Polish	Polish_100_CI_AS
Portuguese (Brazil)	Latin1_General_CI_AS
Russian	Cyrillic_General_100_CI_AS
Spanish	Modern_Spanish_100_CI_AS

Language	Windows collation
Swedish	Finnish_Swedish_100_CI_AS
Turkish	Turkish_100_CI_AS

Windows PowerShell support

You must install Windows PowerShell® Extensions for SQL Server 2012 on the server where you install Service Reporting. The extensions are dependent on SQL Server 2012 Shared Management Objects, which in turn is dependent on Microsoft System CLR Types for Microsoft SQL Server 2012. All of these are available in the SQL Server 2012 Feature Pack that is available in SQL Server 2012 Setup.

Windows PowerShell 3.0 is required. If you are running Windows Server 2008 R2, you can install Windows Management Framework 3.0 from the <u>Microsoft Download Center</u>.

.NET Framework 4.5

Service Reporting requires Microsoft .NET Framework 4.5. You can get the full installation of .NET Framework 4.5 (dotNetFx40_Full_setup.exe) from the <u>Microsoft Download Center</u>.

Excel 2010

Service Reporting requires Microsoft Excel 2010 or Excel 2013 for viewing usage and inventory reports. You can install Excel on any computer that has network connectivity to the Service Reporting data warehouse. Excel Viewer is not supported because it does not support viewing data cubes.

After you install Service Reporting, Excel workbooks that contain the Service Reporting reports are installed in the following folder:

InstallationDrive:\Program Files\Microsoft System Center 2012 R2\Service Reporting\Reports

See Also

Service Reporting in System Center 2012 R2

Support for Service Reporting

Release Notes for System Center 2012 R2 - Service Reporting

Troubleshooting Service Reporting

These troubleshooting tips contain guidance that can resolve common issue when you configure or use Service Reporting in System Center 2012 R2.

No data appears in the Service Reporting reports

Description: After you have installed Service Reporting, no information appears in reports.

Action: Enable logging to see if you can identify the problem. Keep in mind that logging only occurs when Windows Azure Pack for Windows Server calls the usage endpoint. To enable and review Analytic and Debug logging, perform the following steps.

- 1. Log on to the server running Service Provider Foundation and run Eventvwr.msc.
- 2. Enable the Microsoft/Service Provider Foundation/Core/The Analytic Channel for the SPF Core log.
- 3. Filter the log for warning and critical error events to determine if the problem was logged.

Action: Verify that the app pool account is configured for the Service Provider Foundation Usage endpoint has connect and query access to both the Operations Manager data warehouse and the Service Provider Foundation database.

Action: Verify that account used to access the Service Provider Foundation Usage endpoint from Windows Azure Pack is a member of the SPF_Usage group on the server that hosts the Service Provider Foundation Usage endpoint.

Action: Ensure that you have registered the Service Provider Foundation Usage endpoint with Windows Azure Pack. Verify that the correct URL was specified properly.

See Also

Support for Service Reporting

Service Reporting in System Center 2012 R2

System Requirements for Service Reporting

Planning for Service Reporting

This section provides planning information that you should review before you install and configure Service Reporting.

Planning topics

Preparing Windows Azure Pack and System Center Components for Service Reporting
 Provides preparatory information about Windows Azure Pack for Windows Server and
 System Center 2012 R2 components for use with Service Reporting.

Other resources for this component

- Service Reporting in System Center 2012 R2
- Getting Started with Service Reporting

- Support for Service Reporting
- Deploying Service Reporting
- Operating Service Reporting

Preparing Windows Azure Pack and System Center Components for Service Reporting

The following software is used to collect and report inventory and metering data, which aggregates and summaries Information as a Service (laaS) data for IT hosting service providers.

- System Center 2012 R2 Virtual Machine Manager
- System Center 2012 R2 Operations Manager
- System Center 2012 R2 Service Provider Foundation
- Windows Azure Pack for Windows Server
- System Center 2012 R2 Service Reporting

The following process describes how data flows from source systems and results in reports for laaS resources hosted in your environment.

- 1. Data is collected in System Center from Virtual Machine Manager and Operations Manager.
 - Virtual Machine Manager collects data for all virtual machines and hosts that it manages.
 Such data includes CPU, memory, storage, network, started and stopped metrics. This data is stored for a short period.
 - Virtual Machine Manager sends its performance data to Operations Manager by using the VMM to OM connector.
 - c. Operations Manager saves the raw data in the Operations Manager data warehouse.
 - d. Every hour the Operations Manager data warehouse moves the raw data to the hour dataset for long-time storage.
- Data is collected by Windows Azure Pack and Service Provider Foundation.

In Windows Azure Pack for Windows Server, the Collector service collects data for its various Resource Provider services. Each Resource Provider has a list of data sets that it collects.

- a. The Usage Collector collects usage data from every service provider and saves it into the collector database, which is part of Windows Azure Pack.
- b. The Usage Collector for VM Clouds uses Service Provider Foundation to collect data for the VM Resource Service provider.
- c. The Data Collection module for VM Clouds in Windows Azure Pack calls Service Provider Foundation for needed data such as CPU utilization, memory utilization, and disk utilization.
- d. Service Provider Foundation calls Operations Manager data warehouse and returns the data to the Windows Azure Pack Collector service.
- e. Windows Azure Pack Usage Collector saves the data for VM clouds in the Usage Collector database, where it saves the data for one billing cycle, an estimated 40 days.

- 3. Data is analyzed by Service Reporting and stored long-term.
 - a. Service Reporting uses ETL (Extract, Transfer Load) to collect data.
 - b. The extract process contacts the Windows Azure Pack Usage REST API to extract data.
 - c. Windows Azure Pack REST API queries the Collector database and returns the data to the extract process.
 - d. The Service Reporting data warehouse stores the data indefinitely and it is transferred to a data cube and loaded for analysis.
 - e. You can analyze the data in Excel or SharePoint Performance Point.

The following diagram depicts this process.



Configuring VMM and Operations Manager for laaS usage and metering

Before you configure VMM and Operations Manager, ensure the following conditions are met.

- 1. Virtual Machine Manager is already installed.
- 2. Operations Manager is already installed.
- 3. At least one cloud is created in Virtual Machine Manager.
- 4. All service accounts for Operations Manager and Virtual Machine Manager are running under a domain account.

For Operations Manager to monitor and discover clouds, hosts and VMs, the VMM Management pack needs to be imported into Operations Manager. The VMM Management pack has a list of management packs that it depends on such as SQL, IIS and Windows Server. In order to make this easier, a PowerShell script is available to import the management packs. These management packs are located on the Virtual Machine Manager Server under the installation folder. For example, *InstallationDrive*\Program Files\Microsoft System Center 2012 R2\Virtual Machine Manager\ManagementPack.

▶ To import VMM and dependent management packs into Operations Manager

- 1. Copy the Dependencies folder from the Virtual Machine Manager server to the Operations Manager temp folder.
- 2. Start the Operations Manager console as Administrator.
- 3. Select the **Administration** workspace.
- 4. In the Administration Explorer view, select Management Packs.
- Select Import Management Packs.
- 6. Click Add and the select Import from disk.

- 7. Click No.
- 8. On the Operations Manager management server, navigate to the folder where you copied the Management Pack folder previously.
- 9. Select all files and then click Open.
- 10. Select Install.
- 11. Select the **Monitoring** workspace.
- 12. Verify that the following views appear in the monitoring views:
 - Microsoft System Center Virtual Machine Manager
 - Microsoft System Center Virtual Machine Manager PRO

To ensure that Operations Manager and Virtual Machine Manager can exchange information, both components need to have administrative access to each other. To verify that the service accounts have the correct access, perform the following procedure.

To share VMM and Operations Manager service accounts to VM and OM Admins security groups

- 1. Open **Services** on the Operations Manager and Virtual Machine Manager servers.
- 2. On the Virtual Machine Manager server, verify that the user for System Center Virtual Machine Manager is running under a domain account. For example, **CONTOSO\!vmm**.
- 3. On the Operations Manager server, verify that the user for System Center Data Access Service is running under a domain account. For example, **CONTOSO\!om_das**.
- 4. If Operations Manager administrators are managed with an Active Directory security group, then add the VMM Service Account to the OM Admins security group using Active Directory Users and Computers. For example, add the CONTOSO\!VMM user to the OM Admins Security Group.
- 5. If VMM Administrators are managed with an Active Directory security group, then add the VMM Service Account to the OM Admins security group.
- 6. Open the VMM console as an administrator.
- 7. Select Settings and then click System Center Settings.
- 8. Type the fully qualified domain name of the Operations Manager server.
- Use default Service Account that is suggested, and then click Next.
- 10. Specify the Service Account for Operations Manager that was added to the VMM security group previously and then click **Next**.
- 11. Click **Finish** to enable the integration.

To verify integration between Operations Manager and Virtual Machine Manager

- 1. Open the VMM console as an administrator.
- 2. Select **Settings** and then click **System Center Settings**.
- 3. Double-click Operations Manager Server.
- 4. Verify that the connection status is **OK**.
- 5. Select Management Pack and verify that management packs in the list are version

- 3.1.6011.0 or higher.
- 6. Open the Operations Manager console as an administrator.
- 7. Select Monitoring, click Monitoring, and then click Virtual Machine Manager Views.
- 8. When the diagram view is fully displayed, expand **Managed Resources** and then expand **All Clouds**.
- 9. You should see an expanded view of clouds.

To verify that data is being stored in the Operations Manager data warehouse

- 1. Open SQL Management Studio.
- Connect to the SQL instance that hosts the Operations Manager data warehouse database.
- 3. Select the OperationsManagerDW database.
- 4. In the **Object Explorer** pane, expand the database and expand the **Tables** folder.
- 5. Select the **Perf.PerfHourly_** table.
- 6. Right-click Perf.PerfHourly_ table then click Select Top 1000 rows.
- 7. In the results pane, verify that records are displayed.

Configuring SPF and Windows Azure Pack for laaS usage and metering

Before you configure Service Provider Foundation and Windows Azure Pack, ensure the following conditions are met.

- 1. Operations Manager and Virtual Machine Manager integration has been configured and is working.
- 2. Service Provider Foundation has been installed and configured to work with VMM.
- 3. Windows Azure Pack has been installed and configured.
- 4. At a minimum, at least one Windows Azure Pack plan has been configured.

For Operations Manager and Virtual Machine Manager to integrate, you perform the following procedures.

- 1. Register the SPF usage metering endpoint with the Operations Manager operational database
- 2. Resister the SPF Metering Endpoint in the Windows Azure Pack Administration Portal
- 3. Verify that data is stored in the Collection Database

To register the SPF usage metering endpoint with the Operations Manager operational database

- 1. Log on to the Service Provider Foundation server as the Usage App Pool account.
- 2. Verify that the App Pool account for the Usage web endpoint is running as a domain account.

- 3. Ensure that the Usage App Pool account is added to the SQL Login and granted execute permission to where the SCOM Data Warehouse database is installed.
- 4. Open PowerShell with administrative privilege.
- 5. Type Import-module spfadmin
- Save the following PowerShell script as RegistrespfwOMDW.ps1 on the Service Provider Foundation server.

```
# Provide server name to the OM DW instance
$OMDWSqlServer = <OMDWConnectionString>
# Provide server name to the OM instance
$OMServer = <OMManagementServerName>
# Provide the account for Windows Azure Pack to access the
SPF usage web endpoint
$UsageUser = <Domain\Username>
# Register the SCOM Data Warehouse instance to SPF usage
metering
$server = New-SCSPFServer -Name $OMServer -ServerType OMDW
$setting = New-SCSPFSetting -Name $OMDWSqlServer -
SettingString "Data Source=$OMDWSqlServer; Initial
Catalog=OperationsManagerDW; Integrated Security=True" -
SettingType DatabaseConnectionString -Server $server
# Ensure the caller account can access the SPF usage web
endpoint
Net.exe localgroup SPF Usage /add $UsageUser
```

- Update variables in the script that resemble <Variable> with those needed for your environment.
- 8. Run the script on the Service Provider Foundation server by typing ./RegistrespfwOMDW.ps1

To register the SPF Metering Endpoint in the Windows Azure Pack Administration Portal

- Log on to the Windows Azure Pack Administration Portal as an administrator.
- 2. Select VM Clouds.
- 3. Select Register Service Reporting Provider.
- Specify the URL of the Service Provider Foundation Usage Metering web endpoint that you configured previously. For example, https://<ServiceProviderFoundationServiceRoot>:8090/Usage/ Note the /usage/
 - portion of the URL.
- 5. Provide credentials to access the Service Provider Foundation Usage endpoint, which

- you configured previously.
- 6. Verify that the registration completes successfully.

To verify that data is stored in the Collection Database

- 1. Wait at least 1 hour after registering the Service Provider Foundation endpoints and then open SQL Server Management Studio.
- 2. Connect to the SQL instance that hosts the Windows Azure Pack Collection database.
- 3. In the Object Explorer, expand **Databases** and then select the **Microsoft.MgmtSvc.Usage** database.
- 4. Expand the database then expand **Tables**.
- 5. Right-click [usage].[Records] and then click Select Top 1000 rows.
- 6. Verify that records are showing in the Results view.

Deploy Service Reporting

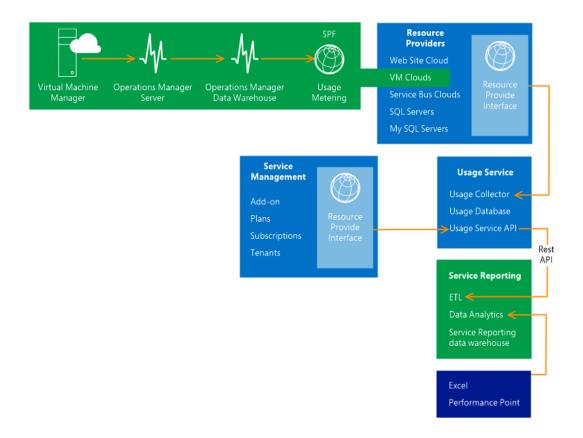
After you have prepared System Center 2012 R2 components and prepared Windows Azure Pack, you are ready to install and configure Service Reporting. The following procedures describe how to install and configure Service Reporting.

- 1. How to Install Service Reporting
- 2. How to Configure Service Reporting for Windows Azure Pack and System Center

Service Reporting Infrastructure Diagram

After you have Service Reporting deployed and configured it with other System Center 2012 R2 components and Windows Azure Pack, your infrastructure should resemble the following diagram. In the diagram, System Center components are shown in green where some data originates. Windows Azure Pack services are shown in blue where other data originates. Reporting components are shown in purple where usage and inventory data is ultimately displayed.

The Windows Azure Pack Usage Service and resource providers are shown as part of the diagram for completeness, however not every item in the diagram is used to collect or show usage or inventory information. Additionally, the REST API is present in the diagram to show the flow of information between computer systems and its functionality is more fully documented at Windows Azure Pack Usage Service.



See Also

Service Reporting in System Center 2012 R2
Planning for Service Reporting

Deploying Service Reporting

The information in this section describes how to install and configure Service Reporting.

Deploying topics

- How to Install Service Reporting
 Describes how to install Service Reporting by using a wizard or a command prompt.
- How to Uninstall Service Reporting
 Describes how to uninstall Service Reporting by using a wizard or a command prompt.
- How to Configure Service Reporting for Windows Azure Pack and System Center
 Describes how to configure Service Reporting after you install it so that it can gather data from source systems.

Upgrading Service Reporting

Provides information about upgrading Service Reporting.

Other resources for this component

- Service Reporting in System Center 2012 R2
- **Getting Started with Service Reporting**
- Support for Service Reporting
- Planning for Service Reporting
- **Operating Service Reporting**

How to Install Service Reporting

In System Center 2012 R2, you can install Service Reporting from the initial page of the System Center 2012 R2 Orchestrator setup program. Before you install, you should review the system requirements for Service Reporting, as described in System Requirements for Service Reporting.

On the SQL Servers that Service Reporting will use, the Microsoft SQL Server Agent service must be running and its startup type should be set to either Automatic or Automatic (Delayed Start).

The Log On As accounts for the following services must be domain accounts:

- SQL Server
- SQL Server Agent
- SQL Server Analysis Services
- SQL Server Integration Services



Note

The Service Reporting Setup wizard does not have an option to install or to configure Service Reporting by using credentials that you can specify. Consequently, Setup installs Service Reporting by using the credentials of the logged-on user. After Setup finishes, you can change the SQL Server account credentials that Service Reporting uses to extract Operations Manager data, as described in How to Configure Service Reporting for Windows Azure Pack and System Center.

If you want to install Service Reporting programmatically, you can do so by using a text-based response file from the command prompt. The command-line parameters that Service Reporting setup supports can be queried using the following command:

Setup.exe -?

You can use the -Slient parameter to install silently with a text file that contains required parameters and use the **-Uninstall** parameter to uninstall Service Reporting.



Note

The Service Reporting Setup wizard can help you prevent installation failure by running a prerequisite checker. Running Setup by using a command prompt does not run the prerequisite checker.

Parameter	Description	Example
-AcceptEULA	Acceptance of Microsoft Software License Terms (yes or no)	Yes
-SendCEIPReports	Participation in Customer Experience Improvement Program (yes or no)	Yes
-UseMicrosoftUpdate	Participation in Microsoft Update (yes or no)	Yes
-InstallFolder	Folder to install to	C:\Program Files\Microsoft System Center 2012 R2\Service Reporting
-DatabaseServer	Name of the server that is running SQL Server where the databases already exist, or where Setup will create them	ServerName
-DatabaseServerInstance	Name of the SQL Server database instance to install to	ServerName\InstanceName
-RepositoryDatabaseName	Name of the usage repository database that already exists, or that Setup will create	UsageRepositoryDB
-StagingDatabaseName	Name of the usage staging database that already exists, or that Setup will create	UsageStagingDB
-DWDatabaseName	Name of the usage data warehouse database that already	UsageWarehouseDB

Parameter	Description	Example
	exists, or that Setup will create	
-AnalysisDatabaseServer	Name of the server that is running SQL Server where the analysis database already exists, or where Setup will create it	ServerName
-AnalysisDatabaseServerInstance	Name of the SQL Server analysis database instance to install to	ServerName\InstanceName
-AnalysisDatabaseName	Name of the analysis database that already exists, or that Setup will create	UsageAnalysisDB

To install Service Reporting by using the Service Reporting Setup wizard

- 1. Start the Orchestrator setup program by using administrative credentials, and then click Install Service Reporting.
- 2. In the Service Reporting Setup wizard, click Install.
- 3. On the Getting Started page, read the license terms, select the I have read, understood and agree to these license terms check box, and then click Next.
- 4. On the Prerequisites page, Setup verifies that all Service Reporting hardware and software requirements are met. After verification, click Next.
- 5. On the Installation Location page, Setup displays the default installation folder. If you want to install in a different folder, click Change Folder. Setup also verifies that 1 gigabyte (GB) of disk space is available. Click Next.
- 6. On the Configure Service Reporting Data Warehouse page, Setup displays information about the existing Service Reporting data warehouse, if one is found. Or, Setup displays default information about the data warehouse that it will create. On this page, you can change the database server to install to, choose an existing SQL Server instance, and modify the names of the databases that Service Reporting will create. Click Next.



Note

Ensure that you use a domain account for the SQL Server and SQL Server Agent services, and update the RunAs account information.

- 7. On the Configure Analysis Server page, Setup displays information about the existing server that is running SQL Server Analysis Services, if one is found. Or, Setup displays default information about the SQL Server Analysis Services database that it will create. On this page, you can change the analysis database to install to, choose an existing SQL Server instance, and modify the name of the analysis database that Service Reporting will create. Click Next.
- 8. On the **Customer Experience Improvement Program** page, choose whether you want to participate in the program, and then choose whether you want to use Microsoft Update to update Microsoft software that is installed on the server, including Service Reporting. Then, click **Next**.
- On the Installation Summary page, Setup displays the choices that you have made about the installation folder, details about the data warehouse, and details about analysis services that it will install. Click Install.
- 10. On the Install page, Setup displays the progress of the Service Reporting components that it installs. When Setup finishes, the Finished page appears and shows links to Service Reporting documentation. If you have previously chosen to use Microsoft Update, you can check for updates by clicking Initiate machine wide Automatic Update. Click Close to complete Setup.

To install Service Reporting by using a command prompt

- 1. Open an elevated command prompt on the server where you want to install Service Reporting.
- 2. By using Setup.exe from the Service Reporting release media, type **Setup.exe** and use all required parameters, or optionally use the **–Silent** parameter.

See Also

Service Reporting in System Center 2012 R2

Deploying Service Reporting

How to Uninstall Service Reporting

How to Configure Service Reporting for Windows Azure Pack and System Center

Upgrading Service Reporting

How to Uninstall Service Reporting

In System Center 2012 R2, you can uninstall Service Reporting to completely remove it from your server.

To uninstall Service Reporting by using the Service Reporting Setup wizard

1. Open Control Panel, click **Programs and Features**, select **System Center 2012 R2 Service Reporting**, and then click **Uninstall**.

- In the System Center 2012 R2 Service Reporting Setup wizard, click Uninstall.
- On the Summary page, click Uninstall.
- 4. On the Finished page, click Close.

To uninstall Service Reporting by using a command prompt

- 1. Open an elevated command prompt on the server where you want to uninstall Service Reporting.
- 2. By using Setup.exe from the Service Reporting release media, type Setup.exe -silent -

See Also

Service Reporting in System Center 2012 R2 Deploying Service Reporting

How to Configure Service Reporting for Windows Azure Pack and System Center

To configure Service Reporting you manually configure it to gather data from Operations Manager and Windows Azure Pack for Windows Server and to provide credentials that Service Reporting uses to connect to the data sources. You accomplish this by editing connection information and then running a Windows PowerShell® script, as explained in the procedures in this topic.

After initial configuration, you can later update user credentials for a changed password by running the script again and using the updated password.

If your Service Reporting data warehouse is hosted on a clustered or remote server that is running Microsoft SQL Server software, you must configure your Windows Azure Pack data source for the environment for each clustered node or remote server that is running SQL Server.

By default, Service Reporting is installed to *InstallationDrive*:\Program Files\Microsoft System Center 2012 R2\Service Reporting.



Note

In the following procedure, you edit an XML file in the InstallationDrive:\Program Files folder. Windows File Protection helps protect this folder tree. Normally, you cannot manually edit and save files in the folder tree. You can, however, run an XML editor of your choice by using administrative credentials and then edit and save files in the folder tree. If you run your XML editor without using administrative credentials, you must save the edited file to a temporary location and then replace the original file by copying it to its original location.

To edit Service Reporting connection information for your Operations Manager and Windows Azure Pack data sources

- 1. Start Windows Explorer and browse to *InstallationDrive*:\Program Files\Microsoft System Center 2012 R2\Service Reporting\Maintenance.
- 2. Edit the MaintenanceConfig.xml file and update the section of the file that resembles
 <OperationsManagerSQLServers>Server1; Server2\SQLInstanceName/OperationsManagerSQL
 Servers> to update the values for Server1 and SQLInstanceName by using the name of the server that is running SQL Server and the name of the SQL Server instance where your Operations Manager operational database is stored. If you have only one server that is running SQL Server in your environment, you can remove any references to other servers that are running SQL Server but are not used.
- 3. Edit the section of the file that resembles
 - <OperationsManagerDB>OMDB1; OMDB2</OperationsManagerDB> and update the value for OMDB1 by using the name of the Operations Manager operational database that is used in your environment. If you have only one Operations Manager operational database in your environment, you can remove the semicolon separator and any references to other databases.
- 4. If you do not have servers that are running Windows Azure Pack in your environment, skip this step. If you do have servers that are running Windows Azure Pack in your environment, edit the section of the file that resembles
 <WAPUsageServiceURI>https://www.WAPCloud.com:111/usage?
 <WAPUsageServiceURI> and replace the value of the URI with the URI that is used in your environment. Update the port number, if necessary.
- 5. Save and close the file.

To add updated connection information and specify user credentials for Service Reporting

- On the server running the Windows Azure Pack Management Portal, open Windows PowerShell as an administrator and type Unprotect-MgmtSvcConfiguration UsageService.
- 2. Navigate to *InstallationDrive*:\inetpub\MgmtSvc-Usage\ and open web.config, and then verify that the following entry exists in the file:

```
<add key="Username" value="UsageClient" />
```

If the entry does not exist, add it to the file.

- 3. Type Set-MgmtSvcSetting -Namespace UsageService -Name Password -Value 'password' -Encode.
- 4. On the server running Service Reporting, start Windows PowerShell as an administrator.
- 5. Browse to the *InstallationDrive*:\Program Files\Microsoft System Center 2012 R2\Service Reporting\Maintenance folder. For example, type cd "c:\Program Files\Microsoft System Center 2012 R2\Service Reporting\Maintenance" and then press Enter.
- 6. If you are not using Windows Azure Pack, skip this step. If you are using Windows Azure Pack, type .\PostDeploymentConfig.ps1 -User UserName -Password Password

and replace the *UserName* and *Password* values with the credentials that are used to connect to Windows Azure Pack. Then, press Enter.

If the script is successful, it displays progress for Windows Azure Pack and Operations Manager registration, along with information about the data warehouse module. Exit Windows PowerShell.

If you are not using Windows Azure Pack, type
 .\RegisterSystemCenterComponents.ps1 and then press Enter.

If the script is successful, it displays progress for Operations Manager registration, along with information about the data warehouse module. Exit Windows PowerShell.

To configure your Windows Azure Pack data source for each clustered SQL Server node or remote server that is running SQL Server

- At each clustered SQL Server node or remote server that is running SQL Server where the Service Reporting data warehouse databases are installed, start Windows PowerShell as an administrator.
- Type \\RemoteComputerName\ServiceReporting\Maintenance\\
 PostDeploymentConfig.ps1 -User UserName -Password Password and replace the UserName and Password values with the credentials that are used to connect to Windows Azure Pack. Then, press Enter.

See Also

<u>Deploying Service Reporting</u>
Service Reporting in System Center 2012 R2

Upgrading Service Reporting

You can upgrade the preview release of System Center 2012 R2 Service Reporting to the final version by using the information in this topic. The version of the Windows Server operating system that you installed on the server determines the upgrade method that you must use. If you uninstall Service Reporting after you upgrade it, you must manually delete some files that Setup does not delete.

Upgrading Service Reporting on a Windowsbased server that has a GUI

To upgrade Service Reporting on a Windows-based server that has a GUI, run Setup.exe and complete the upgrade wizard. After Setup finishes, you must update and then run the PostDeploymentConfig script to ensure that Service Reporting can communicate with its data source systems. Use the following procedure to upgrade Service Reporting.

To upgrade Service Reporting on a Windows-based server that has a GUI

- 1. Use administrative credentials to log on to the Windows-based server that is running Service Reporting.
- 2. Open the final version of the release media that contains the Service Reporting software.
- 3. Run Setup.exe and in the in the Service Reporting Setup wizard, click Upgrade.
- 4. On the **Getting Started** page, read the license terms, select **I have read, understood** and agree to these license terms, and then click Next.
- 5. On the **Configuration** page, a summary of the upgrade is displayed. To continue, click **Upgrade**.
- 6. On the Install page, Setup displays the progress of the Service Reporting components that it installs. When Setup finishes, the Finished page appears and shows links to Service Reporting documentation. If you have previously chosen to use Microsoft Update, you can check for updates by clicking Initiate machine wide Automatic Update. Click Close to complete Setup.
- 7. Edit the MaintenanceConfig.xml file, as described at <u>How to Configure Service Reporting</u> for Windows Azure Pack and System Center.
- 8. Run the PostDeploymentConfig.ps1 Windows PowerShell script, as described at <u>How to Configure Service Reporting for Windows Azure Pack and System Center.</u>

Upgrading Service Reporting on a server that is running a Server Core installation of Windows

The upgrade method for Service Reporting on a server that is running a Server Core installation of Windows is to uninstall Service Reporting and then reinstall it by using a command prompt. After Setup finishes, you must update the MaintenanceConfig.xml file and then run the PostDeploymentConfig.ps1 Windows PowerShell® script to ensure that Service Reporting can communicate with its data source systems. Use the following procedure to upgrade Service Reporting.

To upgrade Service Reporting on a server that is running a Server Core installation

- 1. Use administrative credentials to log on to the Windows-based server that is running Service Reporting.
- Open an elevated command prompt on the server where you want to uninstall Service Reporting, and then browse to the final version of the release media that contains the Service Reporting software.
- 3. Type Setup.exe -silent -uninstall to uninstall Service Reporting.
- By using Setup.exe from the final Service Reporting release media, type Setup.exe silent <ResponseFileName>.
 - You can view a list of all response file parameters at How to Install Service Reporting.
- 5. Edit the MaintenanceConfig.xml file, as described at How to Configure Service Reporting

- for Windows Azure Pack and System Center.
- 6. Run the PostDeploymentConfig.ps1 Windows PowerShell script, as described at <u>How to Configure Service Reporting for Windows Azure Pack and System Center</u>.

Manually deleting files that remain after uninstallation of Service Reporting

If you want to uninstall an upgraded version of Service Reporting, you must manually delete some files that Setup does not delete. Use the following procedure to delete the files.

To manually delete files

- 1. On the server where you uninstalled Service Reporting, browse to the Program Files\Microsoft System Center 2012 R2\ folder.
- 2. Delete the Service Reporting folder.

See Also

Service Reporting in System Center 2012 R2
Deploying Service Reporting

Operating Service Reporting

This section describes the structure of Service Reporting functionality, how to use reports about software inventory and usage data, and how to back up and restore Service Reporting databases.

Operating topics

- Monitoring Service Reporting by Using Operations Manager
 - Provides detailed information about the configuration that you create so that System Center 2012 R2 Operations Manager can monitor the IT infrastructure that Service Reporting uses.
- Understanding Inventory and Usage Data from Windows Azure Pack and System Center
 Provides detailed information about data that Service Reporting aggregates from source systems.
- <u>Using Service Reporting Usage Data and Inventory Reports</u>
 - Provides information about the reports that are available in Service Reporting and explains how to use them.
- How to Back Up and Restore Service Reporting Databases
 Describes how to back up and restore Service Reporting databases in case of failure.

Other resources for this component

- Service Reporting in System Center 2012 R2
- Getting Started with Service Reporting
- Support for Service Reporting
- Planning for Service Reporting
- Deploying Service Reporting
- Operating Service Reporting

Monitoring Service Reporting by Using Operations Manager

This topic describes the configuration that you create so that System Center 2012 R2 Operations Manager can monitor the IT infrastructure that Service Reporting uses.

Addressing prerequisites

Install the following management pack files on the System Center 2012 R2 Operations Manager server, if they are not already installed. These management pack files are in the System Center Management Pack for SQL Server.

- Microsoft.SQLServer.Library.mp
- Microsoft.SQLServer.2012.Discovery.mp, if Service Reporting uses Microsoft SQL Server 2012
- Microsoft.SQLServer.2008.Discovery, if Service Reporting uses Microsoft SQL Server 2008 R2

Configure the following accounts with Run As profiles by using an account that has read access to the Service Reporting SQL Server databases:

- SQL Server Discovery Account
- SQL Server Monitoring Account

Downloading and installing the management pack

Download the <u>System Center 2012 R2 Service Reporting management pack</u>, and then install and configure it by performing the following steps.

To install and configure the Service Reporting management pack

 On the Operations Manager server, run Service Reporting Management Pack.msi to install it. This step installs management pack files to the server. The default folder is InstallationDrive:Program Files (x86)\System Center Management Packs\Service Reporting\.

- 2. Use the Operations Manager console to import the following management pack files that you installed from the management pack:
 - Microsoft.SystemCenter.ServiceReporting.Discovery.mp
 - Microsoft.SystemCenter.ServiceReporting.Library.mp
 - Microsoft.SystemCenter.ServiceReporting.Monitoring.mp
 - Microsoft.SystemCenter.ServiceReporting.Views.mp
- 3. If you need localized views, import the localized management pack files from the Localized Management Packs subfolder of the installation folder.

Monitoring by using the view

After you import the Service Reporting management pack files, the Service Reporting view is available in the Operations Manager console. Alerts appear in the view when the following conditions occur:

- The last successful job takes more than 75 percent of the time of the Service Reporting job interval.
- The current job is running for more than 200 percent of the time of the Service Reporting job interval.

See Also

Service Reporting in System Center 2012 R2
Operating Service Reporting

Understanding Inventory and Usage Data from Windows Azure Pack and System Center

This section provides detailed information about how Service Reporting aggregates data from source systems.

Inventory and usage data topics

- About Service Reporting Data Aggregation and Data Cubes
 Describes how Service Reporting collects data and how long it stores data.
- About VMM Monitored Data from Operations Manager
 Describes the System Center 2012 R2 Virtual Machine Manager data that Service Reporting collects from System Center 2012 Operations Manager.
- About Windows Azure Pack Monitored Data

Describes the data that Windows Azure Pack for Windows Server services collects.

Other resources for this component

- Service Reporting in System Center 2012 R2
- Getting Started with Service Reporting
- Support for Service Reporting
- Planning for Service Reporting
- Deploying Service Reporting
- Operating Service Reporting

About Service Reporting Data Aggregation and Data Cubes

Service Reporting data is collected hourly by a Microsoft SQL Server Agent job that executes a SQL Server Integration Services package to process the Service Reporting data warehouse. If the job is installed and registered to Service Reporting, the job first retrieves data from Windows Azure Pack for Windows Server, and then processes the data. After that, the job gathers data from System Center 2012 R2 Operations Manager and processes that data. As data is processed, it is stored in the Service Reporting data warehouse. The processed data is aggregated from the various sources and optimized for viewing in reports that have various units of time, including daily and monthly periods. Finally, the data is stored in the online analytical processing (OLAP) cube for viewing.

The following table shows how long the data warehouse stores the data that is shown in reports.

Unit of time for data	Storage duration
Hourly	90 days
Daily	Three years
Monthly	Indefinitely

See Also

Service Reporting in System Center 2012 R2

Operating Service Reporting

Understanding Inventory and Usage Data from Windows Azure Pack and System Center

About VMM Monitored Data from Operations Manager

Service Reporting collects data from System Center 2012 R2 Operations Manager about the virtual machines that System Center 2012 R2 Virtual Machine Manager (VMM) hosts for your environment. The following tables describe the data that Service Reporting collects.



All the data that the following tables show for managed entities is collected and stored in the Service Reporting data warehouse. However, only the display name appears in the attribute list of the corresponding entity dimensions in the online analytical processing (OLAP) cube.

Host-group data

Service Reporting collects the following data about host groups.

Data	Description
VMM server	Computer names of the VMM servers in the host group
Display name	Display name of the host group
Computer name	Computer names of the individual computers that VMM hosts within the host group

Hyper-V host

Service Reporting collects the following data about Hyper-V hosts.

Data	Description
VMM server	Computer name of the VMM server
Description	Description of the host
Host group	Names of hosts in the group. If all hosts are in host groups, the value is All Hosts
Host name	Computer name of the host
Total RAM	Total amount of RAM, in bytes, in the virtual machine host

Data	Description
Number of VMs deployed	Number of virtual machines in the host group
Number of VMs running	Number of virtual machines that were running when data was last refreshed
State	State of the host
Virtualization platform	Type of virtual host platform that the virtual machines are running on
Number of processors	Total number of processors that the host uses
Number of cores	Total number of CPU cores that the host uses
Operating system	Name of the server operating system that the host uses
Operating system version	Version number of the host operating system
Display name	Name of the server host

Private cloud

Service Reporting collects the following data about hosted private clouds.

Data	Description
Maximum VM count	Maximum number of virtual machines in the hosted private cloud
Maximum virtual CPU count	Maximum number of virtual CPUs in the hosted private cloud
Custom quota	Custom quota that the hosted private cloud uses
Maximum storage	Maximum storage that the hosted private cloud uses
Maximum memory	Maximum memory that the hosted private cloud uses
VMM server	Server name of the VMM server that is hosting the private cloud
Cluster names	Names of the clusters where the hosted private cloud is stored

Data	Description
Description	Description of the hosted private cloud
Display name	Name of the hosted private cloud

User role

Service Reporting collects the following data about user roles.

Data	Description
VMM server	Computer name of the VMM server where the role is used
Profile	Profile of the user role
Display name	Names of the users who are in the user role

User-role quota

Service Reporting collects the following data about user-role quotas.

Data	Description
Use VM count default	Specifies whether the user-role quota uses the default number of virtual machines
Use memory MB default	Specifies whether the user-role quota uses the default amount of memory, in megabytes
Use storage GB default	Specifies whether the user-role quota uses the default storage space value, in gigabytes
VM count	Indicates the number of virtual machines that the user-role quota allows
CPU count	Indicates the number of CPUs that the user-role quota allows
Memory MB	Indicates the amount of memory, in megabytes, that the user-role quota allows
Storage GB	Indicates the storage space, in gigabytes, that the user-role quota allows
Custom quota	Indicates the custom quota that the user-role

Data	Description
	quota defines

Virtual disk drive

Service Reporting collects the following data about virtual disk drives.

Data	Description
Drive type	Type of the virtual disk drive (dynamic or static)
Path	Path of the virtual disk drive; can be a Universal Naming Convention (UNC) path that resembles \machinename\path\ or <drive>:\path\</drive>
Maximum size	Maximum size of the virtual disk drive, in gigabytes
Current size	Current size of the virtual disk drive, in kilobytes, megabytes, or gigabytes
Classification	Classification of the virtual disk drive
Display name	Name of the disk volume of the virtual disk drive

Virtual machine

Service Reporting collects the following data about virtual machines.

Data	Description
CPU count	CPU count of the virtual machine
State	Current state of the virtual machine
Cost center	Cost center that is assigned to the virtual machine
Total size of virtual disks	Total size of the virtual disks in the virtual machine, in bytes
Dynamic memory	Dynamic memory of the virtual machine
Cloud	Cloud that the virtual machine is a part of
Description	Domain and user of the virtual machine, in

Data	Description
	domain\username format
Owner	Owner of the virtual machine
VMM server	Computer name of the VMM server that hosts the virtual machine
Creation date	Date that the virtual machine was created
Number of NICs	Number of network adapters that are assigned to the virtual machine
Guest FQDN	Fully qualified domain name of the guest virtual machine
Deployment location	Folder path of the file for the virtual machine
Display name	Display name of the guest virtual machine
Total RAM	Amount of RAM that is available for the virtual machine, in megabytes
Number of virtual disk drives	Number of virtual disk drives in the virtual machine
Host name	Computer name of the host server
VGS	Version number of the operating system that is running on the guest virtual machine
Maximum memory	Maximum memory that is available to the virtual machine
Virtual processor count	Number of virtual processors in the virtual machine
Service template	Service template name that the virtual machine uses
Operating system	Name of the operating system that is running on the virtual machine
Operating system version	Version number of the operating system that is running on the virtual machine
Is highly available	Specifies whether the virtual machine is highly available
Total virtual disk size MB	Total virtual disk size of the virtual machine, in megabytes

Data	Description
Maximum memory MB	Maximum memory available to the virtual machine, in megabytes

VMM management server

Service Reporting collects the following data about the VMM management server.

Data	Description
CEIP	Specifies whether the VMM management server collects and sends information for the Customer Experience Improvement Program (CEIP)
Database server	Computer name of the server that hosts the VMM database
Version	Version number of the VMM management server
Analysis server	Name of the Microsoft SQL Server Analysis Services server that the VMM server uses
Display name	Computer name of the VMM server

Virtual machine network

Service Reporting collects the following data about virtual machine networks.

Data	Description
Isolation type	Isolation type that the virtual machine network uses
VMM server	Computer name of the VMM server that is part of the virtual machine network
Description	Description of the virtual machine network
Display name	Name of the virtual machine network

Virtual network adapter

Service Reporting collects the following data about virtual network adapters.

Data	Description
MAC address	Media access control (MAC) address of the virtual network adapter
VLAN enabled	Specifies whether virtual local area network (VLAN) is enabled for the virtual network adapter
MAC address type	Type of MAC address that the virtual network adapter uses
IP v6 address type	IPv6 address type that the virtual network adapter uses
IP v4 address type	IPv4 address type that the virtual network adapter uses
Display name	Display name of the virtual network adapter

Host volume

Service Reporting collects the following data about host volumes.

Data	Description
Mount points	Mount points of the host volume
Capacity	Capacity of the host volume
Volume label	Volume label of the host volume
Is clustered	Specifies whether the host volume is clustered
Is cluster shared volume	Specifies whether the host volume is part of a cluster shared volume
Free space	Free space on the host volume
Is available for placement	Placement readiness of the host volume
Classification	Classification of the host volume
Display name	Name of the host volume

Virtual machine host disk

Service Reporting collects the following data about virtual machine host disks.

Data	Description
Capacity	Capacity of the virtual machine host disk
Is clustered	Specifies whether the virtual machine host disk is clustered
Is SAN Attached	Specifies whether the virtual machine host disk is attached to a storage area network (SAN)
Is pass-through capable	Specifies whether the virtual machine host disk is pass-through capable
Disk ID	Disk ID of the virtual machine host disk
Location	Location of the virtual machine host disk
SM LUN ID	Storage Manager logical unit number (LUN) of the virtual machine host disk
Available capacity	Available capacity of the virtual machine host disk
Display name	Display name of the virtual machine host disk

See Also

Service Reporting in System Center 2012 R2

Operating Service Reporting

Understanding Inventory and Usage Data from Windows Azure Pack and System Center

About Windows Azure Pack Monitored Data

Service Reporting collects data from Windows Azure Pack for Windows Server about the virtual machines in your environment that are hosted by Virtual Machine Manager. The following table describes the data items that Service Reporting collects.

Service Reporting collects the following data about Windows Azure Pack usage.

Data	Description
Start Time	Date and time that the virtual machine started

Data	Description
	running
End Time	Date and time that the virtual machine stopped running
Provider Name	Name of the hosting provider that is hosting the virtual machine
Service Type	Type of service that is hosted; typically, Virtual Machine
Subscription ID	GUID of the subscription for the virtual machine
Metered Service	Utilization of the virtual machine
Subscriber	User of the virtual machine
VM ID	GUID of the System Center 2012 R2 Virtual Machine Manager server
VMM Server	Computer name of the Virtual Machine Manager server that is hosting the virtual machine
VM Name	Computer name of the virtual machine
CPU Allocation Count – Average	Average number of CPUs that are allocated to the virtual machine
CPU Allocation Count – Maximum	Maximum number of CPUs that can be allocated to the virtual machine
CPU Allocation Count - Median	Median number of CPUs that are allocated to the virtual machine
CPU Allocation Count – Minimum	Minimum number of CPUs that can be allocated to the virtual machine
CPU Allocation Count – Sum	Sum of all CPUs that are allocated to the virtual machine
Runtime Seconds – Average	Average number of seconds that the virtual machine has been running
Runtime Seconds – Maximum	Maximum number of seconds that the virtual machine has been running
Runtime Seconds – Median	Median number of seconds that the virtual machine has been running
Runtime Seconds – Sum	Total number of seconds that the virtual

Data	Description
	machine has been running

See Also

Service Reporting in System Center 2012 R2

Operating Service Reporting

Understanding Inventory and Usage Data from Windows Azure Pack and System Center

About Service Reporting Data Aggregation and Data Cubes

About VMM Monitored Data from Operations Manager

Using Service Reporting Usage Data and Inventory Reports

This section provides information about the reports that are available in Service Reporting and explains how to use them.

Using reports topics

- <u>List of Usage Data and Inventory Reports in Service Reporting</u>
 Lists the reports that are available in Service Reporting.
- How to Configure the Connection Information in a Report

Describes how to configure the connection information in a report to your Service Reporting data warehouse.

• How to View and Analyze Usage and Inventory Data in Reports

Describes how to view and analyze information in reports.

Other resources for this component

- Service Reporting in System Center 2012 R2
- Getting Started with Service Reporting
- Support for Service Reporting
- Planning for Service Reporting
- Deploying Service Reporting
- Operating Service Reporting

List of Usage Data and Inventory Reports in Service Reporting

Service Reporting provides the following reports:

- Usage Report.xls
- Server Inventory Report.xls

By default, these reports are installed to *InstallationDrive*:\Program Files\Microsoft System Center 2012 R2\Service Reporting\Reports.

See Also

Service Reporting in System Center 2012 R2

Operating Service Reporting

Using Service Reporting Usage Data and Inventory Reports

How to Configure the Connection Information in a Report

The first time that you open a report in Microsoft Excel, you must configure the workbook data connection so that it can retrieve information from online analytical processing (OLAP) data cubes in the Service Reporting data warehouse.

To configure a workbook connection

- 1. Open a report in Excel.
- 2. Click the Data tab, and then click Connections.
- In the Workbook Connections dialog box, for each connection, view its properties and replace <LocalHost> with the name of your Service Reporting data warehouse management server.
- 4. Replace <dbname> with the name of your data warehouse analysis database.
- 5. Close the Workbook Connections dialog box and save the file.

See Also

Service Reporting in System Center 2012 R2

Operating Service Reporting

Using Service Reporting Usage Data and Inventory Reports

How to View and Analyze Usage and **Inventory Data in Reports**

You can use the following procedure to view and analyze an online analytical processing (OLAP) data cube from Service Reporting by using Microsoft Excel. You can also save your workbooks locally and share them with others. By using the PivotTable field list, you can drag fields from the cube into the workbook. For more information about using Excel slicers, see Creating and Using **Excel Slicers.**

To use the following procedure, you must have Microsoft Excel 2010 or Excel 2013 installed on your computer.



The first time that you analyze a cube by using Excel, the cube can take a few minutes to load.

To view and analyze usage and inventory data

- 1. Open File Explorer and browse to the folder where the reports are stored. By default, the folder is InstallationDrive:\Program Files\Microsoft System Center 2012 R2\Service Reporting\Reports.
- 2. Open one of the report files. For example, open Usage Report.xls.
 - For the usage report, information appears on the default tab and shows periodic consumption (including hourly, daily, and monthly) for groups and instances of virtual machines. Various columns of usage data also appear.
 - For the Server Inventory Report, operating system information appears for virtual machines that tenants are using. The displayed information includes the number of processors and the number of virtual machines.
- 3. Optionally, drag fields from the PivotTable field list and create slicers and charts for the usage or license data to generate a more complex analysis.
- 4. Optionally, save the workbook to a shared folder or another shared location, such as a Microsoft SharePoint folder.

See Also

Service Reporting in System Center 2012 R2

Operating Service Reporting

Using Service Reporting Usage Data and Inventory Reports

How to Back Up and Restore Service Reporting Databases

The databases in a System Center 2012 R2 Service Reporting environment are as follows:

- UsageDatawarehouseDB
- UsageETLRepositoryDB
- UsageStagingDB
- UsageAnalysisDB

If a computer that hosts a database fails, all you need for recovery is the ability to restore the database to a computer that has the same name as the original computer. You should base your disaster recovery strategy for the Service Reporting databases on procedures for general Microsoft SQL Server disaster recovery. For more information, see Back Up and Restore of SQL Server Databases.

When you back up a database, security information such as SQL Server users and roles are backed up with the database. If user's domain credentials are used by SQL Server users and roles and they differ on the computer that you restore to, you must manually update the domain credentials by using SQL Server Management Studio for the database that is restored.

You can restore Service Reporting databases as an optional step when you install Service Reporting, or you can restore databases by overwriting existing databases. After you have restored databases and updated SQL Server users and roles (if necessary), you can register Service Reporting data source systems, including System Center 2012 R2 Operations Manager and Windows Azure Pack for Windows Server, to reestablish connections to the data source systems.

In the event of Windows Azure Pack for Windows Server Usage database restoration, you must ensure that Service Reporting usage data is synchronized with Windows Azure Pack. To ensure that data is synchronized between the two systems, ensure that you perform the "To synchronize restored Windows Azure Pack usage data with Service Reporting" procedure below.

To back up a Service Reporting database

- 1. Start SQL Server Management Studio, select the database to back up, right-click the database, point to **Tasks**, and then click **Back Up**.
- 2. In the **Back Up Database** dialog box, choose your database backup options, and then click **OK** to back up the database.
- 3. When the backup is complete, click **OK** to close the **Back Up Database** dialog box.

To restore a Service Reporting database

- If you are restoring a database as part of a new Service Reporting installation, perform
 the steps for installing Service Reporting at <u>How to Install Service Reporting</u>. Backed-up
 databases are found as part of Setup.
- If you are restoring a database and overwriting an existing database, perform the

following steps:

- a. Start SQL Server Management Studio, select the database to restore, right-click the database, point to **Tasks**, point to **Restore**, and then click **Database**.
- b. In the **Restore Database** dialog box, choose your database restoration options, and then click **OK** to restore the database.
- c. When the restoration is complete, click **OK** to close the **Restore Database** dialog box.

To register Service Reporting with data source systems

 To register Service Reporting with data source systems, perform the steps at <u>How to</u> Configure Service Reporting for Windows Azure Pack and System Center.

To synchronize restored Windows Azure Pack usage data with Service Reporting

- Stop the Windows Azure Pack Usage and UsageCollector services to prevent any new usage events from being added to the system during the failure recovery.
- 2. Restore the Windows Azure Pack data from a saved backup.
- 3. Make note of the last usage event ID by running the following SQL query against the Windows Azure Pack Usage database.

```
SELECT MAX([RecordId])
FROM [Microsoft.MgmtSvc.Usage].[usage].[Records]
```

4. Synchronize the recorded usage event ID by running the following SQL query against the Service Reporting UsageETLRepositoryDB database.

```
DECLARE @LastWAPEventId INT = <EventId>
DECLARE @CurrEventId INT

SELECT @CurrEventId = InputString
FROM [UsageETLRepositoryDB].[dbo].[ProcessModuleInput]
WHERE InputName = '@StartEventId'

IF @CurrEventId > @LastWAPEventId

BEGIN

    UPDATE [UsageETLRepositoryDB].[dbo].[ProcessModuleInput]
    SET InputString = @LastWAPEventId

WHERE InputName = '@StartEventId'

END
```

Replace <EventId> in the above query with the value that you made note of from step 3.

Start the Windows Azure Pack Usage and UsageCollector services.

See Also

Service Reporting in System Center 2012 R2

Operating Service Reporting

Service Management Automation

Service Management Automation is an IT process automation solution for Windows Azure Pack for Windows Server. It enables you to automate the creation, monitoring, and deployment of resources in your Windows Azure Pack environment. This document describes Service Management Automation planning and deployment.

Service Management Automation topics

- Overview of Service Management Automation
 - Provides a brief overview of the purpose and capabilities of Service Management Automation.
- Architecture of Service Management Automation
 - Describes the architecture of a basic Service Management Automation deployment.
- Deploy Service Management Automation
 - Provides a brief overview of the steps to deploy Service Management Automation.
- Administer Service Management Automation
 - Provides information on how to maintain and use Service Management Automation.
- Extending Service Management Automation with runbooks
 - Provides information on how understand and author runbooks in Service Management Automation.

Overview of Service Management Automation

Service Management Automation is a set of tools that is integrated as the Automation extension in Windows Azure Pack for Windows Server. IT pros and IT developers can use Automation to construct, run, and manage runbooks to integrate, orchestrate, and automate IT business processes. Automation runbooks run on the Windows PowerShell workflow engine.

What's in Automation?

Automation uses the following three underlying components that are connected to Windows Azure Pack through the Service Management Automation service endpoint:

Web service

- Connects to Windows Azure Pack
- Distributes runbook jobs to runbook workers
- Supports HTTPS
- Enables security group to control access

Runbook worker

- Executes runbook jobs
- Runs under a service account

PowerShell module

Enables Automation management by using Windows PowerShell cmdlets

Should I use Automation or System Center 2012 - Orchestrator?

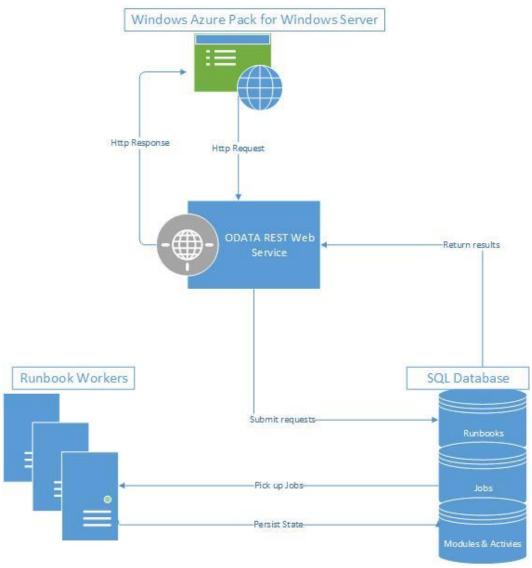
In System Center 2012, System Center 2012 SP1, and System Center 2012 R2, the Orchestrator component enables you to automate business processes and IT operations in your data center without scripting or programming. Orchestrator is a feature in System Center 2012. If you already have System Center 2012 installed, and you do not plan to install Windows Azure Pack, use Orchestrator.

Automation in Windows Azure Pack enables you to automate processes within the Windows Azure Pack. Because Automation runs Windows PowerShell workflows, you can also use Windows PowerShell cmdlets to run other System Center 2012 components, including Orchestrator. If you are planning to use the Windows Azure Pack, use Automation, and then you can continue to leverage your System Center 2012 installation (if one exists).

Architecture of Service Management Automation

The following diagram illustrates each of the Service Management Automation features and the communication between them.

Architecture for Automation in Windows Azure Pack



- The Automation web service communicates with Windows Azure Pack and authenticates
 users
- The SQL Server databases store and retrieve runbooks, runbook assets, activities, integration modules, and runbook job information.
- Runbook workers run the runbooks, and they can be used for load balancing.
- The management portal in Windows Azure Pack is where you author, debug, and start and stop runbooks.

Deploy Service Management Automation

Service Management Automation is a workflow management solution for Windows Azure Pack for Windows Server. It enables you to automate the creation, monitoring, and deployment of resources in your environment. This document describes Service Management Automation deployment.

Deployment topics

- System requirements for Service Management Automation
 - Provides an overview of the hardware, software, and security requirements to deploy Service Management Automation.
- How to install the Service Management Automation web service
 - Provides step-by-step instructions to install the Service Management Automation web service.
- How to install the Service Management Automation runbook worker
 - Provides step-by-step instructions to install a Service Management Automation runbook worker.
- How to install the Service Management Automation PowerShell module
 - Provides step-by-step instructions to install a Service Management Automation PowerShell module.
- Install Service Management Automation from a Command Prompt window
 - Provides complete documentation of the available command-line options for installing Service Management Automation.
- Post-installation tasks for Service Management Automation
 - Provides instructions to use after you install Service Management Automation.
- How to uninstall Service Management Automation
 - Provides step-by-step instructions to install for uninstalling Service Management Automation.

System requirements for Service Management Automation

This topic describes the minimum hardware and software configurations that are required for a full installation of Service Management Automation.

Hardware requirements

The following recommended configurations should be used.

Performance component	Recommendation
Virtual machines	Three, each with a runbook worker and web service installed
	Load-balanced incoming traffic
	Minimum of two cores and 4 GB of RAM for each virtual machine
	60 GB of available disk space
SQL Server	One computer with 8 GB of RAM and eight cores
	Note One month of data under heavy load (12 jobs per minute for a month) results in 20 GB of disk space usage. Job purging should be used to keep this usage from growing beyond this amount.

Software requirements

The following software must be installed for each role.

Role	Prerequisites
Runbook worker	Windows Server 2012 R2
	Windows PowerShell 4.0
Automation web service	Windows Server 2012 R2
	SQL Server 2012 (not Express edition)
	Internet Information Services (IIS) 7.5 (hosts the web service)
	IIS Basic Authentication
	IIS Windows Authentication
	IIS URL Authorization
	ASP.NET 4.5
	.NET Framework 3.5 (for the Setup program)
	.NET Framework 4.5
	WCF HTTP Activation
Windows PowerShell module	Windows PowerShell 4.0

Before installing the web service, use the following procedure to install .NET Framework 4.5 and HTTP Activation on Windows Server 2012 R2:

To install .NET Framework 4.5 and HTTP Activation

- 1. On the Windows Start screen, click the Server Manager tile.
- 2. On the Manage menu in the Server Manager console, click Add Roles and Features.
- 3. Follow the wizard until you reach the **Features** page.
- 4. Expand .NET Framework 4.5 Features.
- 5. Select .NET Framework 4.5 if it isn't already selected.
- 6. Expand WCF Services.
- 7. Select HTTP Activation if it isn't already selected.
- 8. Click **Next**, and follow the prompts to finish the installation.

Running Service Management Automation on Windows Azure virtual machines

Service Management Automation runs on Windows Azure just as it does on physical computer systems.

Service Management Automation was tested by Microsoft by installing and using it in a Windows Azure virtual machine. The testing concluded that Service Management Automation was fully functional and operated exactly the same as it does on physical hardware. Stability and performance benchmarks inside a Windows Azure virtual machine were at a level where no special considerations were needed.

Security Requirements

The following ports must be opened for each role.

Role	Requirement
Runbook worker	None
Automation web service	Default value: 9090. Configurable at install time port defaults to 9090. The installation program for Service Management Automation automatically opens the web service port on the local firewall.
Windows PowerShell module	None

The following certificates are required for each component.

Role	Requirement
Runbook worker	None
Automation web service	A certificate that can be used for Secure Sockets Layer (SSL) encryption over HTTPS. The installation program for Service Management Automation can be used to generate a self-signed certificate.
Windows PowerShell module	None

How to install the Service Management Automation web service

The Service Management Automation service endpoint enables you to automate IT administration and business processes by using Windows PowerShell workflow-based runbooks in Windows Azure Pack for Windows Server.

Use the following information to install and configure the Automation web service in Windows Azure Pack. The Service Management Automation PowerShell module is a required prerequisite of the Service Management Automation web service, so you must install the Service Management Automation PowerShell module before you deploy the Service Management Automation web service.

You can also install the Service Management Automation components by using an unattended installation. For more information, see <u>Installing Service Management Automation from a Command Prompt.</u>

Install the Service Management Automation web service

The Service Management Automation web service endpoint provides the connection between Service Management Automation and Windows Azure Pack. The Service Management Automation web service can be installed from the System Center 2012 R2 Orchestrator installation software.

Install the web service on any machine that can communicate with Windows Azure Pack and an instance of SQL Server.

To install the Service Automation web service

1. In the folder where you downloaded the System Center 2012 R2 Orchestrator installation software, click Setup to start the Setup wizard.

- 2. Under Service Management, click Web Service, and then click Install.
- 3. Complete the product registration information, and then click **Next**.
- 4. Review and accept the license terms, and then click **Next**.
- 5. Select **Service Management Automation Web Service**, and then click **Next**. This will launch the prerequisite check.
- 6. Review the results of the check. If all items are installed, click Next.



Note

If you see an X next to any of the prerequisite software, you must install the item, and then run the prerequisite check again. You cannot complete installation of the service endpoint until you pass the prerequisite check.

7. Provide the following information for the database the endpoint to use, and then click **Next**.

Server	Enter the name of the database server. By default, this is localhost.
	The format is sqlserver\instance, where \instance is optional.
Port number	Enter the port number that you want to use for the database. The default is 1433.
Database name	Enter the name of the database. The default is SMA.
Authentication Credentials	Select the type of authentication that you want to use. You can use Windows authentication or SQL Server authentication.
	If you choose SQL Server authentication, enter the user name and password for the computer running SQL Server.

8. Provide the following information to configure the Internet Information Settings (IIS) for the web service, and then click **Next**.

Domain security group or users with access	Enter a security group or one or more users who can grant access to the web service.
Application pool name	SMA This name is not configurable.
Application pool credentials	Specify the credentials to use for the

application pool. These are the credentials
that the web service will run under.

- 9. Enter the port number for the web service to use. By default, this is 9090.
- 10. Choose the security certificate to use to encrypt communication between Windows Azure Pack and the Service Management Automation web service endpoint.

You can have the installer generate a self-signed certificate to use, or you can select an existing certificate in your local certificate store.

Click Next.

- Review the location for the web service files. You can accept the default or specify a different location. Click Next.
- Indicate whether you want to participate in the Customer Experience Improvement Program (CEIP) and whether you want to use Microsoft Update to keep your software upto-date. Click Next.
- Review the installation summary, and then click Install.
 After the installation is complete, install a runbook worker as described in How to install the Service Management Automation runbook worker.

How to install the Service Management Automation runbook worker

The Service Management Automation web service endpoint enables you to automate IT administration and business processes by using Windows PowerShell workflow-based runbooks in Windows Azure Pack for Windows Server.

Use the following information to install and configure the Automation runbook worker in Windows Azure Pack. Before installing or uninstalling an Automation runbook worker, ensure that you have stopped the Runbook Worker service (rbsvc) on the computer where the runbook worker is installed. For instructions on how to avoid any data loss when removing a runbook worker, including Windows PowerShell cmdlets and scripting help for this operation, see the <u>overview of runbook worker deployments</u>.

You can also install the Service Management Automation components by using an unattended installation. For more information, see Install Service Management Automation from a Command Prompt.

Install a runbook worker

The Service Management Automation runbook worker provides the functionality to run Automation runbooks. The Service Management Automation runbook worker can be installed

from the System Center 2012 R2 Orchestrator installation software. Install the runbook worker on a physical or virtual machine that has access to the same SQL Server instance that the Service Management Automation web service is using.

To install the runbook worker

- 1. In the folder where you downloaded the System Center 2012 R2 Orchestrator installation software, click Setup to start the Setup wizard.
- 2. Under Service Management, click Runbook Worker, and then click Install.
- 3. Follow the instructions in the Setup wizard.

After the installation is complete, use administrative credentials to configure Automation in the Windows Azure Pack management portal.

How to install the Service Management Automation PowerShell module

The Service Management Automation service endpoint enables you to automate IT administration and business processes by using Windows PowerShell workflow-based runbooks in Windows Azure Pack for Windows Server.

Use the following information to install and configure the Service Management Automation PowerShell module. The Service Management Automation PowerShell module is a required prerequisite for the Service Management Automation web service, so you must install the Service Management Automation PowerShell module before you deploy the Service Management Automation web service.

You can also install the Service Management Automation components by using an unattended installation. For more information, see <u>Installing Service Management Automation from a Command Prompt.</u>

Install the Service Management Automation PowerShell module

The Service Management Automation PowerShell module provides the Windows PowerShell cmdlets that are used to administer Service Management Automation. The Service Management Automation PowerShell module can be installed from the System Center 2012 R2 Orchestrator installation software.

To install the Service Management AutomationPowerShell module

- 1. In the folder where you downloaded the System Center 2012 R2 Orchestrator installation software, start the Setup wizard.
- 2. Under Service Management, click PowerShell administration, and then click Install.

3. Follow the instructions in the Setup wizard.

Install Service Management Automation from a Command Prompt window

You can install the features of Service Management Automation by using commands in the Command Prompt window to guide the Windows Installer program for an unattended install.

Windows Installer files

Your installation media contains Windows Installer files for each Service Management Automation of the following features:

- PowerShell module: PowershellModuleInstaller.msi
- Web service: WebServiceInstaller.msi
- Runbook worker: WorkerInstaller.msi



The installation options must be entered at a command prompt. An answer file is not supported.

PowerShell module installation options

The Service Management Automation PowerShell module is a required prerequisite of the Service Management Automation web service, so you must install the Service Management Automation PowerShell module before you deploy the Service Management Automation web service. The PowerShell module installer takes no parameters. For example, you could use the following command:

msiexec.exe /i PowershellModuleInstaller.msi

Web service installation options

The following variables can be specified at a command prompt to override default behaviors.

Installation item	Command line switch	Valid values
IIS application pool	APPOOLACCOUNT	String
IIS application	APPOOLPASSWORD	String

pool		
IIS application pool	ADMINGROUPMEMBERS	String (a comma-separated list of users to add to the IIS Administrators group)
SQL Server database	CREATEDATABASE	"Yes" or "No" (the default value is "No")
SQL Server database	DATABASEAUTHENTICATION	SQL, Windows (the default value is Windows). If DATABASEAUTHENTICATION = SQL, you must also specify SQLUSER and SQLPASSWORD
SQL Server database	SQLUSER	String
SQL Server database	SQLPASSWORD	String
SQL Server database	SQLSERVER	In the format "Server name, port number." (The default values are "localhost, 1433." Supply a port number of 0 to specify a dynamic port.)
SQL Server database	SQLINSTANCE	String (optional server instance name)
SQL Server database	SQLDATABASE	String (the default database name value is SMA)
IIS web service	SITENAME	String (the default value is "SMA")
IIS web service	WEBSERVICEPORT	Integer (the default value is "9090")
IIS web service	INSTALLFOLDER	String (the default value is c:\inetpub\Service Management Automation)
IIS web service	USESSL	"Yes" or "No" (the default value is "Yes")
IIS web service	SPECIFYCERTIFICATE	"Yes" or "No" (the default value is "No"). A certificate is automatically created if you specify "No." If you select "Yes," also provide CERTIFICATESERIAL.
IIS web service	CERTIFICATESERIAL	Serial number of an existing certificate in concatenated hexadecimal format and with no spaces between digits, for example:

		"45C324C02318F48D4A9C4FC832B2CDCC"
Event tracing (ETW)	ETWMANIFEST	"Yes" or "No" (the default value is "Yes")
Software Quality Metrics (SQM) for Customer Experience Improvement Program (CEIP) reporting	SENDCEIPREPORTS	"Yes" or "No" (the default value is "No")
Automatic Microsoft Update	MSUPDATE	"Yes" (opt-in) or "No" (no change; this is the default value)
Product key	PRODUCTKEY	String

If logging is desired, use the Msiexec.exe command and specify the log path. For example, you could use the following command (be sure to use the name of your SQL Server instance).

```
msiexec.exe /i WebServiceInstaller.msi /L*v C:\Andreas\WebServiceInstaller.log
CREATEDATABASE="Yes" SQLSERVER="localhost" DATABASEAUTHENTICATION="Windows"
SQLDATABASE="SMA123"
```

Runbook worker installation options

A runbook worker cannot be installed on the same computer as another runbook worker. Also, you must install the runbook worker on a computer that has access to the same SQL Server instance that the Service Management Automation web service is using.

The following variables can be specified at a command prompt to override default behaviors.

Installation item	Command line switch	Valid values
Windows service	SERVICEACCOUNT	String
Windows service	SERVICEPASSWORD	String
SQL Server database	CREATEDATABASE	"Yes" or "No" (the default value is "No")
SQL Server database	DATABASEAUTHENTICATION	SQL Server or Windows (the default value is Windows)

SQL Server database	SQLUSER	String
SQL Server database	SQLPASSWORD	String
SQL Server database	SQLSERVER	In the format "Server name, port number" (The default values are "localhost, 1433." Supply a port number of 0 to specify a dynamic port.)
SQL Server database	SQLINSTANCE	String (optional server instance name)
SQL Server database	SQLDATABASE	String (the default database name value is SMA)
File install location	INSTALLFOLDER	String (the default value is C:\Program Files\Microsoft System Center 2012 R2\Service Management Automation)
Event tracing (ETW)	ETWMANIFEST	"Yes" or "No" (the default value is "Yes")
Software Quality Metrics (SQM) for Customer Experience Improvement Program (CEIP) reporting	SENDCEIPREPORTS	"Yes" or "No" (the default value is "No")
Automatic Microsoft Update	MSUPDATE	"Yes" (opt-in) or "No" (no change; this is the default value)
Product key	PRODUCTKEY	String

If logging is desired, use the Msiexec.exe command and specify the log path. For example, you could use the following command (be sure to use the name of your SQL Server instance).

msiexec.exe /i WorkerInstaller.msi /L*v C:\Andreas\WorkerInstaller.log CREATEDATABASE="Yes" SQLSERVER="localhost" DATABASEAUTHENTICATION="Windows" SQLDATABASE="SMA123"



If you install additional runbook workers, you must run the Windows PowerShell cmdlet **New-SmaRunbookWorkerDeployment** to properly configure the runbook worker.

- Stop the Runbook server service (RunbookService.exe) on each computer on which a runbook worker is installed.
- 2. Run the following Windows PowerShell command:
 - New-SmaRunbookWorkerDeployment -<ComputerName> -<WebServiceEndpoint>
- Restart the Runbook server service on each computer on which a runbook worker is installed.

See also

Install Service Management Automation

Post-installation tasks for Service Management Automation

After you install Service Management Automation, perform the following best practices.

Replace untrusted Self-Signed Certificates with trusted certificates

Each Service Management Automation component is installed on an Internet Information Services (IIS) website that, by default, is configured with a self-signed certificate. Because these self-signed certificates are not issued by any of the trusted root certification authorities that your browser loads on startup, your browser displays a security warning when you attempt to connect to any of the sites. We recommend that you replace the self-signed certificates with certificates that are issued by a trusted root certification authority to avoid this experience.

How to uninstall Service Management Automation

You can use the following procedures to uninstall a Service Management Automation web console, runbook worker, or PowerShell module. Before uninstalling an Automation runbook worker, ensure that you have stopped the Runbook Worker service (rbsvc) on the computer where the runbook worker is installed. For instructions on how to avoid any data loss when removing a runbook worker, including Windows PowerShell cmdlets and scripting help for this operation, see the <u>overview of runbook worker deployments</u>.

To uninstall a Service Management Automation web service

1. On the computer on which the web service is installed, click Start, and then click Control

Panel.

- 2. Under Programs, click Uninstall a program.
- 3. Under Name, double-click System Center 2012 R2 Service Management Automation Web Service.
- 4. Follow the prompts, and the **Uninstalling features** page appears and uninstallation progress is displayed.

To uninstall a Service Management Automation runbook worker

- 1. Ensure that you have prepared for removing the runbook worker as described in the overview of runbook worker deployments.
- 2. On the computer on which the runbook worker is installed, click **Start**, and then click **Control Panel**.
- 3. Under Programs, click Uninstall a program.
- 4. Under Name, double-click System Center 2012 R2 Service Management Automation Runbook Worker.
- 5. Follow the prompts, and the **Uninstalling features** page appears and uninstallation progress is displayed.

To uninstall a Service Management Automation PowerShell module

- 1. On the computer on which the PowerShell module is installed, click **Start**, and then click **Control Panel**.
- 2. Under Programs, click Uninstall a program.
- 3. Under Name, double-click System Center 2012 R2 Service Management Automation Powershell.
- 4. Follow the prompts, and the **Uninstalling features** page appears and uninstallation progress is displayed.

Administer Service Management Automation

Service Management Automation must be administered separately from Windows Azure Pack for Windows Server. These topics describe how to administer Service Management Automation.

Administration topics

- Scaling Service Management Automation up or down
 Describes how to approach and implement adding runbook workers and web services to or removing them from Service Management Automation.
- How to purge the Service Management Automation database

Describes how to approach and implement to install a Service Management Automation PowerShell module.

- Establish trust between Service Management Automation and Service Provider Foundation
 Provides an overview of how to establish trust relationships between other Windows Azure
 Pack components and Service Management Automation.
- Extending Service Management Automation with runbooks
 Describes the default runbooks that ship with Service Management Automation and how to add additional runbooks.

Establish trust between Service Management Automation and Service Provider Foundation

For Service Provider Foundation to successfully call the Service Management Automation web service, the Service Management Automation web service certificate must be trusted by the server on which Service Provider Foundation is installed. This topic applies whether you are using a self-signed certificate or a certification authority certificate for your Service Management Automation web service.

To trust the Service Management Automation certificate

- 1. Log on to the computer that is running Service Provider Foundation.
- 2. In a web browser, connect to the Service Management Automation web service endpoint. This procedure assumes that Internet Explorer is being used and that it is being run with elevated privileges.
- 3. Click Continue to this website (not recommended).
- 4. In the browser address bar, click **Certificate Error**, and then click **View Certificates** on the **Certificate Invalid** pop-up.
- 5. In the Certificate dialog box, click Install Certificate.
- 6. In the Certificate Import wizard, select the Local Machine option and click Next.
- 7. Select the Place all certificates in the following store option, and then click Browse.
- 8. In the Select Certificate Store dialog box, click **Trusted People**, and then click **OK**.
- 9. Click **Next**, and then review your choices and click **Finish**.
- 10. If the import is successful, click OK to close the **Certificate** dialog box.

Service Provider Foundation should now be able to successfully call the Service Management Automation web service.

For detailed guidance to understand, create, test, and publish runbooks, see <u>Authoring Runbooks</u> in Service Management Automation.

Scaling Service Management Automation up or down

Use the guidance in this section to scale out a machine tier in a service that is deployed in Service Management Automation. You can add runbook workers and web services to add additional capacity for runbook processing.

Initial recommendations

The recommended configuration is 3 virtual machines, each with an installed a runbook worker and web service. The incoming web traffic should be load balanced. The machines should each be at least each two cores and contain a minimum of 4 GB of RAM, along with 60 GB of storage. Only one PowerShell module should be installed.

SQL Server recommendations

For the SQL Server database, 8 GB of RAM and 8 cores are recommended.

1 month of data under heavy load (12 jobs per minute for a month) results in 20 GB of database space usage. By default, job purging should keep the space usage from growing much beyond this. For more on settings for database purging, see How to purge the Service Management Automation database.

Scale out Service Management Automation

If runbook jobs are running slowly, you might want to increase the number of runbook workers that are sharing workloads. New runbook worker/web service instances must be installed on their own virtual machines.

Before installing or uninstalling a Service Management Automation runbook worker, ensure that you have stopped the Runbook Worker service (rbsvc) on the computer where the runbook worker is installed. For instructions on how to avoid any data loss when removing a runbook worker, including Windows PowerShell cmdlets and scripting help for this operation, see the overview of runbook worker deployments.

How to purge the Service Management Automation database

In Service Management Automation, database purging is automatic, but you can adjust it to your needs.

To enable the automatic database purge, you must enable the SQL Server Agent (MSSQLSERVER) service for Automatic start. The service is not turned on by default, but it is frequently started by SQL Server database administrators for other tasks.

If the SQL Server Agent service is not running, the purge will not occur and eventually the customer will experience performance issues, first in the portal. and then with the back end.)

The job task that performs purge can be set up in the installer even if the customer is not running the SQL Server Agent service. But it will not do anything until the service is enabled.

The database purge job is automatic, but it can be regulated by the Service Management Automation administrator.

- By default, the database purge job runs every 15 minutes, and it runs only if there are records to purge.
- Records are purged only if they are older than the default duration of 30 days. This time is configurable by using the Set-SmaAdminConfiguration cmdlet and setting the – PurgeJobsOlderThanCountDays parameter.
- If the total job record count exceeds the MaxJobRecords parameter set by the same Set-SmaAdminConfiguration cmdlet, then more job records will be purged. The default value for this parameter is 120,000 records.

Extending Service Management Automation with runbooks

Runbooks are how Service Management Automation extends the functionality of Windows Azure Pack for Windows Server.

Extension topics

- Authoring Runbooks in Service Management Automation
 - Describes how to create runbooks to implement and extend your business processes in Service Management Automation.
- Service Management Automation sample runbooks
 - Describes the runbooks that are targeted toward hosting service processes in Service Management Automation.
- Service Management Automation system runbooks
 - Describes the runbooks that are used to govern internal processes in Service Management Automation.

Authoring Runbooks in Service Management Automation

Runbooks in Service Management Automation are Windows PowerShell workflows that run on Automation Worker servers. They provide the ability to automate administrative processes for managing and deploying cloud servers or any other function that a Windows PowerShell script can perform.

The additional services provided by Automation for working with Windows PowerShell Workflows include the following:

- Centralized storage and management of runbooks with the Automation database and Windows Azure Pack management portal.
- Scalable architecture for scheduling and running runbooks with multiple Worker servers.
- Global resources that are centrally managed and available to all runbooks.
- User interface for authoring and testing runbooks.
- Set of cmdlets for managing and starting runbooks.

Runbook Authoring Guide Topics

The following topics provide information on creating and working with Automation runbooks.

Runbook Concepts

Describes the concepts of Automation runbooks and Windows PowerShell Workflows.

Runbook and Module Operations

Procedures for working with runbooks and modules using both the Windows Azure Pack management portal and Windows PowerShell.

Global Resources

Resources such as connections and variables that are available to all runbooks.

Runbook Concepts

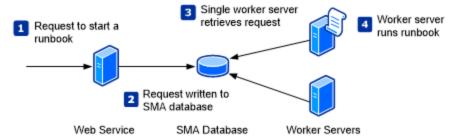
Runbooks in Automation are implemented as Windows PowerShell workflows. This section provides a brief overview of critical features of workflows that are common to Automation runbooks. Complete details on workflows are available in Getting Started with Windows PowerShell Workflow.

Runbook Execution

Requests to start a runbook are performed by the Service Management Automation web service using either the Service Management Portal or the Start-SmaRunbook Windows PowerShell cmdlet. The web service writes this request to the Automation database where it is retrieved by

one of the Automation Worker servers. You do not have control over which Worker server will service the request.

The runbook will run on the Worker server that services the request and remotely accesses any computers or other resources that it will work with. This requires the cmdlets in the runbook to be able to remotely access these resources. Alternatively, the runbook can include an InlineScript command in order to use PowerShell Remoting to run commands locally on a target computer. This concept is illustrated in the following diagram.



Permissions

In order for a runbook to perform its required actions, it must have permissions to access the resources that it works with. Runbooks always run in the context of the service account of the Automation Runbook Service. If this account does not have required permissions, then you can use either a **Credentials** or a **Connections** global resource in your runbook to run required commands using credentials with the required permissions. These credentials can either be used with a cmdlet that accepts credentials through a parameter or with InlineScript to run a block of code using alternate credentials.

Windows PowerShell Workflows

A workflow is a sequence of programmed, connected steps that perform long-running tasks or require the coordination of multiple steps across multiple devices or managed nodes. The benefits of a workflow over a normal script include the ability to simultaneously perform an action against multiple devices and the ability to automatically recover from failures. A Windows PowerShell Workflow is a Windows PowerShell script that leverages Windows Workflow Foundation. While the workflow is written with Windows PowerShell syntax and launched by Windows PowerShell, it is processed by Windows Workflow Foundation.

Basic Structure

A Windows PowerShell Workflow starts with the **Workflow** keyword followed by the body of the script enclosed in braces. The name of the workflow follows the **Workflow** keyword as shown in the following syntax. The name of the workflow matches the name of the Automation runbook.

```
Workflow Test-Runbook {
```

```
<Commands>
```

}

To add parameters to the workflow, use the **Param** keyword as shown in the following syntax. The Service Management Portal will prompt the user to provide values for these parameters when they start the runbook. This sample uses the optional Parameter attribute which specifies whether or not the parameter is mandatory.

```
Workflow Test-Runbook
{
    Param
    (
       [Parameter(Mandatory=<$True | $False>]
       [Type]$<ParameterName>,

      [Parameter(Mandatory=<$True | $False>]
       [Type]$<ParameterName>
    )
       <Commands>
}
```

Naming

The name of the workflow should conform to the Verb-Noun format that is standard with Windows PowerShell. You can refer to Approved Verbs for Windows PowerShell Commands for a list of approved verbs to use. The name of the workflow must match the name of the Automation runbook. If the runbook is being imported, then the filename must match the workflow name and must end in .ps1.

Activities

An activity is a specific task in a workflow. Just as a script is composed of one or more commands, a workflow is composed of one or more activities that are carried out in a sequence. Windows PowerShell Workflow automatically converts many of the Windows PowerShell cmdlets to activities when it runs a workflow. When you specify one of these cmdlets in your runbook, the corresponding activity is actually run by Windows Workflow Foundation. For those cmdlets without a corresponding activity, Windows PowerShell Workflow automatically runs the cmdlet within an InlineScript activity. There is a set of cmdlets that are excluded and cannot be used in a workflow unless you explicitly include them in an InlineScript block. For further details on these concepts, see Using Activities in Script Workflows.

Workflow activities share a set of common parameters to configure their operation. For details about the workflow common parameters, see <u>about WorkflowCommonParameters</u>.

Integration Modules

An Integration Module is a package that contains a Windows PowerShell Module and can be imported into Automation. Windows PowerShell Modules contain cmdlets and workflows that can be used in Automation runbooks. Products such as Operations Manager and Virtual Machine Manager have modules that include cmdlets specific to their operation. For a Windows PowerShell workflow to use a cmdlet that is defined in a module, the module must be installed on the computer where the script is running.

Integration Modules that are imported into Automation are automatically installed on all Worker servers so they are available to all runbooks. Since Automation is based on Windows PowerShell 4.0, it supports auto loading of modules meaning that cmdlets from installed modules can be used without importing them into the script with Import-Module.

Any Windows PowerShell module can be imported into Automation as long as all of its dependencies can be located in a single folder. If the module depends on registry settings or files not in the default path, then it can be imported, but it will most likely not work because Automation will not be able to locate its dependencies.

Modules with external dependencies can still be used in a runbook but must be manually installed on each Worker server or accessed on another host using an **InlineScript** script block. While the cmdlets in these modules can be used in runbooks, they will not be discovered by Automation to support such features as the Insert Activity wizard. In order to use this feature, you can create a Portable module using the **New-SmaPortableModule** cmdlet. This cmdlet creates a module that includes a stub for each of its cmdlets and can be imported into Automation. When a runbook uses one of those cmdlets, the stub redirects the call to the actual cmdlet in the external module. That module must be installed on each Worker server or the call will fail.

Parallel Execution

One advantage of Windows PowerShell Workflows is the ability to perform a set of commands in parallel instead of sequentially as with a typical script. This is particularly useful in runbooks since they may perform multiple actions that take a significant time to complete. For example, a runbook might provision a set of virtual machines. Rather than performing each provisioning process in sequence with one another, the actions could be performed simultaneously increasing overall efficiency. Only when all are complete would the runbook continue.

You can use the **Parallel** keyword to create a script block with multiple commands that will run concurrently. This uses the syntax shown below. In this case, Activity1 and Activity2 will start at the same time. Activity3 will start only after both Activity1 and Activity2 have completed.

```
Parallel
{
     <Activity1>
```

```
<Activity2>
}
<Activity3>
```

You can use the **ForEach -Parallel** construct to process commands for each item in a collection concurrently. The items in the collection are processed in parallel while the commands in the script block run sequentially. This uses the syntax shown below. In this case, Activity1 will start at the same time for all items in the collection. For each item, Activity2 will start after Activity1 is complete. Activity3 will start only after both Activity1 and Activity2 have completed for all items.

```
ForEach -Parallel ($<item> in $<collection>)
{
     <Activity1>
     <Activity2>
}
<Activity3>
```

The **Sequence** keyword is used to run commands in sequence within a **Parallel** script block. The **Sequence** script block runs in parallel with other commands, but the commands within the block run sequentially. This uses the syntax shown below. In this case, Activity1, Activity2, and Activity3 will start at the same time. Activity4 will start only after Activity3 has completed. Activity5 will start after all other activities have completed

```
Parallel
{
     <Activity1>
     <Activity2>

     Sequence
     {
           <Activity3>
           <Activity4>
      }
}
<Activity5>
```

Checkpoints

A checkpoint is a snapshot of the current state of the workflow that includes the current value for variables and any output generated to that point. It is saved to the Automation database so that

the workflow can resume even in the case of an outage on the Worker server. The checkpoint data is removed once the runbook job is complete.

You can set a checkpoint in a workflow with the **Checkpoint-Workflow** activity. When you include this activity in a runbook, a checkpoint is immediately taken. If the runbook is suspended by an error, when the job is resumed, it will resume from the point of the last checkpoint set. In the following sample code, an error occurs after Activity2 causing the runbook to suspend. When the job is resumed, it starts by running Activity2 since this was the last checkpoint set.

```
<Activity1>
Checkpoint-Workflow
<Activity2>
<Error>
<Activity3>
```

You should set checkpoints in a runbook after activities that may be prone to error and should not be repeated if the runbook is resumed. For example, your runbook may create a virtual machine. You could set a checkpoint both before and after the commands to create the virtual machine. If the creation fails, then the commands are repeated when the runbook is resumed. If the creation succeeds but the runbook later fails, then the virtual machine will not be created again when the runbook is resumed.

For more information about checkpoints, see Adding Checkpoints to a Script Workflow.

Suspending a Runbook

You can force a runbook to suspend itself with the **Suspend-Workflow** activity. This activity will set a checkpoint and cause the workflow to immediately suspend. Suspending a workflow is useful for runbooks that may require a manual step to performed before another set of activities are run.

For more information about suspending a workflow, see Making a Workflow Suspend Itself.

InlineScript

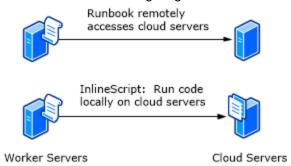
The InlineScript activity runs a block of commands in a separate, non-workflow session and returns its output to the workflow. While commands in a workflow are sent to Windows Workflow Foundation for processing, commands in an InlineScript block are processed by Windows PowerShell. The activity uses the standard workflow common parameters including PSComputerName and PSCredential which allow you to specify that the code block be run on another computer or using alternate credentials.

InlineScript uses the syntax shown below.

```
InlineScript
{
     <Script Block>
```

```
} <Common Parameters>
```

The most common use for **InlineScript** in a runbook is to run a block of code on another computer. This is required when cmdlets in your runbook are not installed on the Worker servers or if the action only has permissions to be performed locally on the target computer. This is illustrated in the following diagram.



} -PSComputer \$con.ComputerName -PSCredential \$cred

In order to run the code block on another computer, the **PSComputer** and **PSCredential** parameters are used with the **InlineScript** activity. A global resource such as a **Credential** or **Connection** is typically used in a runbook to provide values for these parameters. The following sample code runs a set of commands on a computer represented by a connection called MyConnection.

While **InlineScript** activities may be critical in certain runbooks, they should only be used when necessary for the following reasons:

- You cannot use checkpoints from within an **InlineScript** block. If a failure occurs within the block, it must be resumed from the beginning.
- InlineScript affects scalability of the of the runbook since it holds the Windows PowerShell session on the Worker server for the entire length of the InlineScript block.
- Activities such as Get-AutomationVariable and Get-AutomationPSCredential are not available in an InlineScript block. You can pass values into the script though with the \$Using scope modifier. See <u>about InlineScript</u> for further detail.

If you do need to use an **InlineScript**, you should minimize its scope. For example, if your runbook iterates over a collection while applying the same operation to each item, the loop should occur outside of the **InlineScript**. This will provide the following advantages:

- You can <u>Checkpoints</u> the workflow after each iteration. If the job is suspended or interrupted and resumed, the loop will be able to resume.
- 2. You can use ForEach -Parallel to handle collection items concurrently.

For further details on using InlineScript, see Running Windows PowerShell Commands in a Workflow.

Starting a Runbook from another Runbook

A runbook can call another runbook just as it would another activity, providing values for each of its parameters. For example, to call a runbook called Sample-MyChildRunbook with parameters called Param1 and Param2, the parent runbook would use the following line.

```
Sample-MyChildRunbook -Param1 "Param1 value" -Param2 "Param2 value"
```

To return data from a child runbook so that it can be used by a parent runbook, the child runbook needs to write its output to the standard output stream using cmdlets such as Write-Output. Any output from a cmdlet that is not directed into a variable will also be written to the standard output stream available to the parent. The parent runbook typically assigns this value to a variable where it can be used. All runbooks installed in Automation are available to be used from all other runbooks.



If a runbook includes a call to another runbook that is not yet published in Automation, the calling runbook will need to be opened in draft mode and published again after the runbook being called has been published. This is because Automation creates a reference to any runbooks being called by the current runbook when it is published. If those runbooks don't yet exist, then the references cannot be created. By publishing the parent runbook again, the references can then be created.

See Also

Authoring Runbooks in Service Management Automation **Runbook and Module Operations** Global Resources

Runbook and Module Operations

The steps for creating and working with Service Management Automation runbooks are different depending on whether you using the Windows Azure Pack management portal or Windows PowerShell. The basic steps for various common operations using both methods are provided in the following sections.

Creating a Runbook

Creating a Runbook with the Windows Azure Pack management portal

When you create a runbook with the management portal for administrators, you first create an empty runbook and then later edit it with the Automation editor to create the script. With Windows PowerShell, you import an existing script file to create the runbook.

To create a runbook with the Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the bottom of the window, click **New**.
- 3. Click Quick Create.
- 4. Type a name for the runbook in the **Runbook Name** box taking into account the <u>Naming</u> recommendations.
- 5. Optionally, type a description in the Description box and one or more tags separated by commas.
- 6. Click Create.
- 7. Follow one of the procedures in Editing a Runbook to edit the runbook's contents.

To create a runbook with Windows PowerShell

When you create a runbook with Windows PowerShell, you use the editor of your choice
to write the workflow script. You then use the lmport-SmaRunbookcmdlet to import the
script file and create the runbook. The name of the script file must match the name of the
workflow, and this name will be used for the runbook.

When you create the runbook, you can use the –Tag parameter to set a tag on the runbook. You cannot set the tag with Windows PowerShell after the runbook has been created.

The following sample commands show how to create a runbook. This example

```
$webServer = 'https://MyServer'
$port = 9090
$runbookPath = 'c:\runbooks\Sample-TestRunbook.ps1'
Import-SMARunbook -WebServiceEndpoint $webServer -Port $port -Path $runbookPath
```

Editing Runbook Properties

Properties of a runbook include its description and logging properties in addition to a tag that may be required for the runbook to be accessed by certain services. You can edit these properties with the management portal for administrators or with the Set-SMARunbookConfiguration cmdlet.

The Tag property cannot be modified for an existing runbook with Windows PowerShell but can only be set when the runbook is created with lmport-SMARunbook.

To edit runbook properties with the Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the top of the window, click Runbooks.
- 3. Locate the runbook to edit and click on its name.
- 4. At the top of the window, click Configure.
- 5. Set any properties that should be changed.
- 6. Click Save when your edits are complete.

To edit runbook properties with Windows PowerShell

The following sample commands show how to set the properties for a runbook. In this
example, the Description and Debug Logging properties are modified.

```
$webServer = 'https://MyServer'
$port = 9090
$runbookPath = 'c:\runbooks\Sample-TestRunbook.ps1'
$runbookName = 'Sample-TestRunbook'
Set-SMARunbookConfiguration -WebServiceEndpoint $webServer -
Port $port -Name $runbookName -Description "Sample runbook" -
LogDebug $true
```

Editing a Runbook

One a runbook has been created, you can edit the draft version of its workflow. You later Publishing a Runbook the Draft version so that it is available to be run in production.

The management portal for administrators includes an editor that you can use to view and edit runbooks. In addition to providing text editing capabilities, the editor provides the ability to automatically insert code for Global Settings, Activities, and Runbooks.

The Automation editor includes a feature to insert code for Activities, Settings and Runbooks into a runbook. Rather than typing in the code yourself, you can select from a list of available assets and have the appropriate code inserted into the runbook.

To Edit a Runbook with the Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the top of the window, click Runbooks.
- 3. Locate the runbook to edit and click on its name.
- 4. At the top of the window, click Author.

- 5. Click Draft.
- 6. Perform the required editing.
- 7. Click **Save** when your edits are complete.

Inserting Code into a Runbook

To Insert Code for a Runbook into a Runbook

- 1. Open the runbook in the SMA editor.
- 2. At the bottom of the screen, click **Insert** and then **Runbook**.
- 3. Select the runbook to insert from the center column and click the right arrow.
- 4. If the runbook has parameters, they will be listed for your information.
- 5. Click the check button.
- 6. Code to run the runbook will be inserted into the runbook.
- 7. If the runbook requires parameters, provide an appropriate value in place of the data type surrounded by braces <>.

To Insert a Global Setting into a Runbook

- 1. Open the runbook in the SMA editor.
- 2. At the bottom of the screen, click **Insert** and then **Setting**.
- 3. In the **Setting Action** column, select the type of code that you require
- 4. Select from the available assets in the center column.
- 5. Click the check button.

To Insert an Activity into a Runbook

- 1. Open the runbook in the SMA editor.
- 2. At the bottom of the screen, click Insert and then Activity.
- 3. In the **Integration Module** column, select the module that contains the activity.
- 4. In the **Activity** pane, select an activity.
- 5. In the **Description** column, note the description of the activity. Optionally, you can click **View detailed help** to launch help for the activity in the browser.
- 6. Click the right arrow.
- 7. If the activity has parameters, they will be listed for your information.
- 8. Click the check button.
- 9. Code to run the activity will be inserted into the runbook.
- 10. If the activity requires parameters, provide an appropriate value in place of the data type surrounded by braces <>.

Note that only activities from modules that are imported into SMA are available from the Insert feature. Any cmdlet from a module installed on the Worker servers can be used in a

runbook, but if they are not imported into SMA, then the editor has no knowledge of them. For further details, see the <u>Modules</u> section of this guide.

Editing a Runbook Using Windows PowerShell

To edit a runbook with Windows PowerShell, you edit the script using the editor of your choice and save it to a .ps1 file. You can use the Get-SmaRunbookDefinition to retrieve the contents of the runbook and then Edit-SmaRunbook cmdlet to replace the existing script with the modified one.

To Retrieve the Contents of a Runbook Using Windows PowerShell

The following sample commands show how to retrieve the script for a runbook. In this
example, the Draft version is retrieved. It is also possible to retrieve the Published version
of the runbook although this version cannot be changed.

```
$webServer = 'https://MyServer'
$port = 9090
$runbookPath = 'c:\runbooks\Sample-TestRunbook.ps1'
$runbookName = 'Sample-TestRunbook'
$content = Get-SMARunbookDefinition -WebServiceEndpoint
$webServer -Port $port -Name $runbookName -Type Draft
```

To Change the Contents of a Runbook Using Windows PowerShell

 The following sample commands show how to replace the existing contents of a runbook with the contents of a script file.

```
$webServer = 'https://MyServer'
$port = 9090
$runbookPath = 'c:\runbooks\Sample-TestRunbook'
$runbookName = 'Sample-TestRunbook'
Edit-SMARunbook -WebServiceEndpoint $webServer -Port $port -Name $runbookName -Path $runbookPath
```

Testing a Runbook

You can test the Draft version of a runbook before publishing it. This allows you to validate its operation before making it available in production by overwriting the existing Published version. When you test the runbook, the Draft version is run and any output sent to the **Output Pane** in the management portal for administrators.

When a runbook is tested, its output is written more quickly to the Automation database than a production run of the runbook since it is assumed that an administrator is interacting with the test

version. Also, Debug, Verbose, and Progress streams are disabled for test runs regardless of their settings in the runbook configuration. You can turn them on in the script by setting the appropriate Preference variable.

To Test a Runbook Using Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the top of the screen, select Runbooks.
- 3. Locate the runbook to edit and click on its name.
- 4. At the top of the screen, click **Author**.
- 5. Click Draft.
- 6. At the bottom of the screen, click Test.
- 7. Click **Yes** to the verification message.
- 8. If the runbook has parameters you will be presented with a dialog box to provide values for each.
- 9. Inspect the output in the Output Pane.

Publishing a Runbook

Each runbook has a Draft and a Published version. Only the Published version is available to be run, and only the Draft version can be edited. The Published version is unaffected by any changes to the Draft version. When the Draft version should be made available, then you publish it which overwrites the Published version with the Draft version.

To Publish a Runbook Using Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the top of the screen, select **Runbooks**.
- 3. Locate the runbook to edit and click on its name.
- 4. At the top of the screen, click Author.
- 5. Click Draft.
- 6. At the bottom of the screen, click **Publish**.
- 7. Click **Yes** to the verification message.

To Publish a Runbook Using Windows PowerShell

The following sample commands show how to publish a runbook.

```
$webServer = 'https://MyServer'

$port = 9090

$runbookPath = 'c:\runbooks\Sample-TestRunbook.ps1'

$runbookName = 'Sample-TestRunbook'
```

Enumerating Installed Modules

- To Get a List of Installed Modules using Windows Azure Pack management portal
 - 1. Select the **Automation** workspace.
 - 2. At the top of the window, click **Assets**.
 - 3. Inspect the assets in the list with a Type of **Module**.

To Get a List of Installed Modules using Windows PowerShell

The following sample commands retrieve all modules installed in SMA.

```
$webServer = 'https://MyWebServer'

$port = 9090

Get-SMAModule -WebServiceEndpoint $webServer -Port $port
```

Building an Integration Module

An <u>Integration Module</u> is a package that contains a <u>Windows PowerShell Module</u>. For information on writing a Windows PowerShell Module, see <u>Writing a Windows PowerShell Module</u>. An Integration Module can contain any of the valid Module Types specified in <u>Windows PowerShell Modules</u>. This includes Script Modules (.psm1), Binary Modules (.dll), and Manifest Modules.

The Integration Module package is a compressed file with the same name as the module and a .zip extension. It contains a single folder also with the name of the module. The Windows PowerShell module and any supporting files, including a manifest file (.psd1) if the module has one, must be contained in this folder.

If the module should contain a <u>Connection</u> type, it must also contain a file with the name <ModuleName>-Automation.json that specifies the connection type properties. This is a json file with the following format.

```
"ConnectionFields": [

{
    "IsEncrypted": false,
    "IsOptional": false,
    "Name": "ComputerName",
    "TypeName": "System.String"
},
```

```
"IsEncrypted": false,
    "IsOptional": true,
    "Name": "Username",
    "TypeName": "System.String"
},

{
    "IsEncrypted": true,
    "IsOptional": false,
    "Name": "Password",
    "TypeName": "System.String"
}],

"ConnectionTypeName": "DataProtectionManager",
    "IntegrationModuleName": "DataProtectionManager"]
```

Folder	Files
MyModule	MyModule.psd1
	MyModule-Automation.json

Importing a Module

A module is a compressed file with a .zip extension that contains a folder which includes one of the following file types:

- A module (psm1 file)
- A module manifest (psd1 file)

To Import a Module using Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the bottom of the window, click **Import Module**.
- 3. Click Browse for File.
- 4. Select the module file and click OK.
- 5. Click the checkmark button on the dialog box.

To Import a Module using Windows PowerShell

The following sample commands show how to import a module.

```
$webServer = 'https://MyWebServer'
$port = 9090
$modulePath = 'C:\Modules\MyModule.psm1'
Import-SMAModule -WebServiceEndpoint $webServer -Port $port -Path $modulePath
```

Enumerating Activities in a Module

To Get a List of Activities in a Module using Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the top of the window, click **Assets**.
- 3. Locate the module and select it.
- 4. Scroll to the bottom of the **Module Details** screen and inspect its activities.
- 5. Optionally, click the magnifying glass icon to filter for particular activities.

To Get a List of Activities in a Module using Windows PowerShell

1. The following sample commands show how to retrieve the activities in a particular module.

```
$webServer = 'https://MyWebServer'
$port = 9090
$moduleName = 'MyModule'
$module = Get-SMAModule -WebServiceEndpoint $webServer -Port
$port -Name $moduleName
$module.Activities
```

To Get a List of Activities in All Modules using Windows PowerShell

 The following sample commands show how to retrieve the activities in all modules installed in SMA.

```
$webServer = 'https://MyWebServer'

$port = 9090

$modules = Get-SMAModule -WebServiceEndpoint $webServer -Port
$port
```

\$modules | foreach {\$_.Activities} | sort Name, ModuleName |
ft Name, ModuleName, Description

See Also

Authoring Runbooks in Service Management Automation
Runbook Concepts
Global Resources

Global Resources

Global Resources are stored in the Service Management Automation database and available to all runbooks in an Automation environment. You create and configure them using either the Automation workspace in the Windows Azure Pack management portal or with the appropriate cmdlets in Windows PowerShell. From a runbook, you can retrieve and set values for global resources with activities in the **RunbookConstructs** module. The Windows PowerShell cmdlets are available to use in runbooks, but the activities are recommended as they are more efficient.

Credentials

An Automation Credential is either a username and password that can be used with Windows PowerShell commands or a certificate that is uploaded to the server. The properties for a credential are stored securely in the Automation database and can be accessed in the runbook with either the **Get-AutomationPSCredential** or **Get-AutomationCertificate** activity.

Windows PowerShell Cmdlets

The cmdlets in the following table are used to create and manage credentials with Windows PowerShell.

Cmdlets	Description
Get-SmaCertificate	Retrieves an Automation certificate.
Get-SmaCredential	Retrieves an Automation PowerShell credential.
Remove-SmaCertificate	Removes an Automation certificate.
Remove-SmaCredential	Removes an Automation PowerShell credental.
Set-SmaCertificate	Creates a new certificate or sets the properties for an existing certificate including uploading the certificate file and setting the password for

Cmdlets	Description
	a .pfx.
Set-SmaCredential	Creates a new Automation PowerShell credential or sets the properties for an existing credential.

Runbook Activities

The activities in the following table are used to access credentials in a runbook.

Activities	Description
Get-AutomationCertificate	Gets a certificate to use in a runbook.
Get-AutomationPSCredential	Gets a username/password to use in a runbook.



You should avoid using variables in the -Name parameter of Get-AutomationPSCredential and Get-AutomationCertificate since this can complicate discovering dependencies between runbooks and Automation variables.

Creating a new credential

To create a new PowerShell credential with Windows Azure Pack management portal

- Select the **Automation** workspace.
- 2. At the top of the window, click **Assets**.
- 3. At the bottom of the window, click **Add Setting**.
- 4. Click Add Credential.
- 5. In the Credential Type dropdown, select PowerShell Credential.
- 6. Type a name for the credential in the **Name** box.
- 7. Click the right arrow.
- 8. Type in values for each property.
- 9. Click the check mark to save the credential.

To create a new certificate with Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the top of the window, click **Assets**.
- 3. At the bottom of the window, click Add Setting.

- 4. Click Add Credential.
- 5. In the Credential Type dropdown, select Certificate.
- 6. Type a name for the certificate in the **Name** box.
- 7. Click the right arrow.
- 8. Click **Browse for File** and navigate to either a .cer or .pfx file.
- 9. If you selected a .pfx file, then provide its password.
- 10. Click the check mark to save the certificate.

To create a new PowerShell credential with Windows PowerShell

• The following sample commands show how to create a new credential.

```
$webServer = 'https://MyWebServer'
$port = 9090
$credName = 'MyCredential'
$user = 'contoso\MyUser'
$pwd = ConvertTo-SecureString -String 'P@$$w0rd' -AsPlainText
-Force
$cred = New-Object -TypeName
System.Management.Automation.PSCredential -ArgumentList
$user,$pwd

Set-SmaCredential -WebServiceEndpoint $webServer -port $port
-Name $credName -Value $cred
```

To create a new PowerShell certificate with Windows PowerShell

 The following sample commands show how to create a new certificate by importing a certificate file.

```
$webServer = 'https://MyWebServer'
$port = 9090
$certName = 'MyCertificate'
$path = 'c:\certs\MyCertificate.pfx'
$certPwd = ConvertTo-SecureString -String 'P@$$w0rd' -
AsPlainText -Force
Set-SmaCertificate -WebServiceEndpoint $webServer -port $port -Name $certName -Path $certPath -Password $certPwd
```

Using a PowerShell Credential in a Runbook

You retrieve a PowerShell Credential in a runbook with the **Get-AutomationPSCredential** activity. This returns a PSCredential object that you can use in the workflow.

To use a PowerShell credential in a runbook

The following sample commands show how to use a PowerShell credential in a runbook.
 In this example, the credential is used with an <u>InlineScript</u> activity to run a set of commands using alternate credentials.

Connections

An Automation Connection contains the information required to connect to a service or application from a runbook. This information is defined in the module for the application and typically includes such information as the username and password and the computer to connect to. Other information may also be required such as a certificate or a subscription Id. The properties for a connection are stored securely in the Automation database and can be accessed in the runbook with the **Get-AutomationConnection** activity.

Windows PowerShell Cmdlets

The cmdlets in the following table are used to create and manage credentials with Windows PowerShell.

Cmdlets	Description
Get-SmaConnection	Retrieves the values for each field in a particular connection.
Get-SmaConnectionField	Retrieves the field definitions for a particular connection type.
Get-SmaConnectionType	Retrieves the available connection types.
New-SmaConnection	Creates a new connection.
Remove-SmaConnection	Remove an existing connection.
Set-SmaConnectionFieldValue	Sets the value of a particular field for an existing connection.

Runbook Activities

The activities in the following table are used to access credentials in a runbook.

Activities	Description
Get-AutomationConnection	Gets a connection to use in a runbook.

Creating a New Connection

To create a new connection with Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the top of the window, click Assets.
- 3. At the bottom of the window, click **Add Setting**.
- 4. Click Add Connection.
- 5. In the **Connection Type** dropdown, select a connection type.
- 6. Type a name for the connection in the **Name** box.
- 7. Click the right arrow.
- 8. Type in a value for each property.
- 9. Click the check mark to save the connection.

To create a new connection with Windows PowerShell

 The following sample commands create a new Virtual Machine Manager connection with the name MyVMMConnection. Note that we use a hashtable to define the properties of the connection. This is because different types of connections require different sets of properties. A connection of another type would use a different set of field values.

For more information about hash tables, see about_Hash_Tables.

```
$webServer = 'https://MyWebServer'

$port = 9090

$connectionName = 'MyConnection'

$fieldValues =

@{"Username"="MyUser"; "Password"="password"; "ComputerName"="MyComputer"}

New-SMAConnection -WebServiceEndpoint $webServer -port $port
-Name $connectionName -ConnectionTypeName

"VirtualMachineManager" -ConnectionFieldValues $fieldValues
```

Using a connection in a runbook

Use the **Get-AutomationConnection** activity to use a connection in a runbook. This activity retrieves the values of the different fields in the connection and returns them as a hashtable which can then be used with the appropriate commands in the runbook.

For more information about hash tables, see about_Hash_Tables.

To use a connection in a runbook

 The following sample code shows how to use a connection to provide the computer name and credentials for an InlineScript block that runs commands on another computer.

Variables

Automation variables are values that are available to all runbooks. They can be created, modified, and retrieved from the Windows Azure Pack management portal, Windows PowerShell, or from within a runbook. Automation variables are useful for the following scenarios:

- Share a value between multiple runbooks.
- Share a value between multiple jobs from the same runbook.
- Manage a value from the Windows Azure Pack management portal or from the Windows PowerShell command line that is used by runbooks.

Automation Variables are persisted in the Automation database so that continue to be available if a Worker server is restarted. This also allows a value to be set by one runbook that is then used by another, or is used by the same runbook the next time that it is run.

When a variable is created, you must specify its data type from the following list. You can only assign a value of the correct type to a variable.

- String
- Integer
- Boolean
- Datetime

When a variable is created, you can specify that it be stored encrypted. When a variable is encrypted, it is stored securely in the SMA database, and its value cannot be retrieved from the

Get-SMAVariable cmdlet. The only way that the value can retrieved is from the Get-AutomationVariable activity in a runbook. You can store multiple values of the defined type to a single variable by creating a hashtable.

Windows PowerShell Cmdlets

The cmdlets in the following table are used to create and manage variables with Windows PowerShell.

Cmdlets	Description
Get-SmaVariable	Retrieves the value of an existing variable.
Set-SmaVariable	Creates a new variable or sets the value for an existing variable.

Runbook Activities

The activities in the following table are used to access credentials in a runbook.

Activities	Description
Get-AutomationVariable	Retrieves the value of an existing variable.
Set-AutomationVariable	Creates a new variable or sets the value for an existing variable.



You should avoid using variables in the -Name parameter of Get-AutomationVariable since this can complicate discovering dependencies between runbooks and Automation variables.

Creating a new Automation variable

To create a new variable with Windows Azure Pack management portal

- 1. Select the Automation workspace.
- 2. At the top of the window, click **Assets**.
- 3. At the bottom of the window, click Add Variable.
- 4. Click Add Variable.
- 5. In the **Type** dropdown, select a data type.
- 6. Type a name for the variable in the **Name** box.
- 7. Click the right arrow.

- 8. Type in a value for each property.
- 9. Click the check mark to save the variable.

To create a new variable with Windows PowerShell

The Set-SmaVariable cmdlet both creates a new variable and sets the value for an
existing variable. The following sample commands show how to create a variable of type
string.

```
$web = 'https://MySMAServer'
$port = 9090
Set-SMAVariable -WebServiceEndpoint $web -Port $port -Name
'MyVariable' -Value 'My String'
```

Using a variable in a runbook

Use the **Get-AutomationVariable** activity to use a variable in a runbook.

To use a variable in a runbook

 The following sample code shows how to set and retrieve a variable in a runbook. In this sample, it is assumed that variables of type integer named NumberOfIterations and NumberOfRunnings and a variable of type string named SampleMessage have already been created.

```
[int] $NumberOfIterations = Get-AutomationVariable -Name
'NumberOfIterations'
[int] $NumberOfRunnings = Get-AutomationVariable -Name
'NumberOfRunnings'
[string] $SampleMessage = Get-AutomationVariable -Name
'SampleMessage'
Write-Output "Runbook has been run $NumberOfRunnings times."
for ($i = 1; $i -le $NumberOfIterations; $i++) {
    Write-Output "$i`: $SampleMessage"
}
Set-AutomationVariable -Name NumberOfRunnings -Value
(NumberOfRunngs += 1)
```

Schedules

Automation Schedules are used to schedule runbooks to run automatically. This could be either a single date and time for the runbook to run once. Or it could be a recurring schedule to start the runbook multiple times. Schedules are typically not accessed from runbooks.

Windows PowerShell Cmdlets

The cmdlets in the following table are used to create and manage variables with Windows PowerShell.

Cmdlets	Description
Get-SmaSchedule	Retrieves a schedule.
Set-SmaSchedule	Creates a new schedule or sets the properties for an existing schedule.

Creating a new Schedule

To create a new schedule with Windows Azure Pack management portal

- 1. Select the **Automation** workspace.
- 2. At the top of the window, click Assets.
- 3. At the bottom of the window, click Add Setting.
- 4. Click Add Schedule.
- 5. Type a name for the variable in the **Name** box.
- 6. Click the right arrow.
- 7. Select One Time or Daily.
- 8. Select a Start Time.
- 9. For a Daily schedule, perform the following steps:
 - a. If the schedule should not run indefinitely, select **Schedule Expires On** and specify an expiration date and time.
 - b. In the **Recur Every** box, select the number of days the schedule should recur.
- 10. Click the check mark to save the variable.

To create a new schedule with Windows PowerShell

The **Set-SmaSchedule** cmdlet both creates a new schedule and sets the value for an existing variable. The following sample Windows PowerShell commands create a new schedule called My Daily Schedule that starts on the current day and continues for one year every day at noon:

```
$web = 'https://MySMAServer'

$port = 9090

$scheduleName = 'My Daily Schedule'

$startTime = (Get-Date).Date.AddHours(12)

$expiryTime = $startTime.AddYears(1)
```

Set-SMASchedule -WebServiceEndpoint \$web -Port \$port -Name \$scheduleName -ScheduleType OneTimeSchedule -StartTime \$startTime -ExpiryTime \$expiryTime -DayInterval 1

See Also

Authoring Runbooks in Service Management Automation
Runbook Concepts
Runbook and Module Operations

Service Management Automation system runbooks

The following runbooks ship with Service Management Automation as internal in-system runbooks. They intended to be used only by the Service Management Automation system, and they are not available to be used in the Windows Azure Pack for Windows Server.

System Runbooks

DiscoverAllLocalModules

- Runs immediately after a runbook worker is installed
- Discovers all native modules on the Windows Server system where the runbook worker has been installed, and extracts activities and activity metadata for these modules so that their activities can be used when authoring runbooks in Windows Azure Pack.

SetAutomationModuleActivityMetadata

- Runs immediately after a module is imported into Service Management Automation
- Extracts activities and activity metadata from a newly imported module so that its activities can be used when authoring runbooks in Windows Azure Pack.

Service Management Automation sample runbooks

The following runbooks ship with Service Management Automation as sample runbooks to illustrate techniques and best practices. They are available to be used in the Automation extension in Windows Azure Pack for Windows Server.

Sample runbooks

Runbook name	Description
Sample-Deleting-VMCloud-Subscription	Demonstrates a useful scenario for triggering a runbook when a user deletes a VM Clouds subscription.
Sample-Managing-Azure	Shows how to connect to a Windows Azure subscription and perform basic operations using the Windows Azure PowerShell module.
Sample-Managing-ConfigurationManager	Demonstrates the capability of Service Management Automation to connect into System Center Configuration Manager.
Sample-Managing-DataProtectionManager	Demonstrates how to connect to a Data Protection Manager (DPM) server and view information about the disks found on the DPM server.
Sample-Managing-MySQLServers	Demonstrates how to retrieve a security token which will be used to then retrieve a list of host servers.
Sample-Managing-OperationsManager	Demonstrates the capability of Service Management Automationto connect to System Center Operations Manager.
Sample-Managing-Orchestrator	Shows how to connect to System Center Orchestrator and start an Orchestrator runbook to leverage your existing infrastructure.
Sample-Managing-Plans	Demonstrates how to create a new plan and add the SQL Server service with a defined quota to the new plan.
Sample-Managing-ServiceBusClouds	Demonstrates how to connect to a Service Bus Cloud server and view information about the created namespaces.
Sample-Managing-SQLServers	Demonstrates how create a new server group and add a SQL hosting server.
Sample-Managing-UserAccounts	Demonstrates how to create a User in Windows Azure Pack for Windows Server, which will be created in Windows Azure Pack, and appear in the management portal for administrators Users extension. However, this user should also be integrated into the authentication provider (for example, AuthSite) for accessing

Runbook name	Description
	the management portal for tenants, which is not included in this sample.
Sample-Managing-VirtualMachineManager	Demonstrates how to connect to a Virtual Machine Manager (VMM) server and view information about the VMM server license.
Sample-Managing-VMClouds	Demonstrates how to access information about a Service Provider Foundation server's database connection and information about the VMM server objects managed by Service Provider Foundation.
Sample-Managing-WebSiteCloud	Demonstrates how to connect to a Web Site Clouds controller server and view information about the Web Site Clouds deployed servers.
Sample-Modify-VMCloud-Subscription	Demonstrates a useful scenario for triggering a runbook when tenant or administrator suspends or activates a VM Clouds subscription.
Sample-Using-Activities	Demonstrates the capability of Service Management Automation to use activities
Sample-Using-Checkpoints	Demonstrates the capability to use checkpoints in Service Management Automation.
Sample-Using-Connections	Demonstrates the capability of Service Management Automation to use connections to connect into remote systems.
Sample-Using-Credentials	Demonstrates the capability of Service Management Automation to use credentials, and outputs the user who the Service Management Automation runbook is running as. Then, it connects to the server 'ServerName' and outputs the user specified by 'SampleCredential' who is accessing the server.
Sample-Using-Modules	Demonstrates importing modules in runbooks, and outputs the number of already imported modules on the server 'ServerName'. Then, it imports the module specified by 'ModulePath' and outputs the new module count and information corresponding to the newly

Runbook name	Description
	imported module.
Sample-Using-RunbookParameters	Demonstrates how to use input parameters for runbooks and also specify whether parameters are required, provide default parameter values, and use parameter values later in the workflow.
Sample-Using-Runbooks	Demonstrates how to call a runbook from within another runbook.
Sample-Using-SuspendWorkflow	Demonstrates how to force a runbook to suspend. This could be useful if a manual step is required before a runbook should continue, such as receiving sign-off approval from a specific person. Once the manual step is completed, the suspended runbook would be resumed manually to continue the runbook.
Sample-Using-Variables	Demonstrates the capability of Service Management Automation to use variables.
Sample-Using-VMCloud-Automation	Demonstrates a useful scenario for triggering a runbook at the start of a Service Provider Foundation event.

Service Provider Foundation

Service Provider Foundation is provided with System Center 2012 - Orchestrator, a component of System Center 2012 R2 (and System Center 2012 SP1). Service Provider Foundation exposes an extensible OData web service that interacts with Virtual Machine Manager (VMM). This enables service providers and hosters to design and implement multi-tenant self-service portals that integrate laaS capabilities available on System Center 2012 R2.

The topics in this section pertain to both System Center 2012 R2 and System Center 2012 SP1 unless otherwise indicated.

For information about how to program client applications for Service Provider Foundation, see the <u>Service Provider Foundation Developer's Guide</u>.

Service Provider Foundation topics

Architecture Overview of Service Provider Foundation
 Provides a high-level overview of Service Provider Foundation.

What's New in Service Provider Foundation

Summarizes key new features and improvements in the current versions.

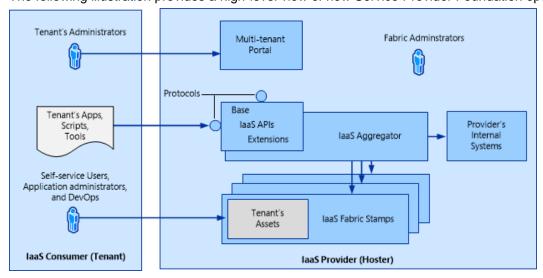
- Deploying Service Provider Foundation
 - Provides system requirements, installation, and other deployment guidance topics.
- Administering Service Provider Foundation
 - Describes how to use the capabilities and features of Service Provider Foundation.
- Cmdlets in System Center 2012 Service Provider Foundation
 - Provides a reference for Service Provider Foundation Windows PowerShell cmdlets.

Architecture Overview of Service Provider Foundation

Service providers can use Service Provider Foundation technology to offer infrastructure as a service (laaS) to their clients. If a service provider has a front-end portal for clients to interact with, Service Provider Foundation makes it possible for the clients to access the resources on their hosting provider's system without making changes to the portal.

Overview

The following illustration provides a high-level view of how Service Provider Foundation operates.



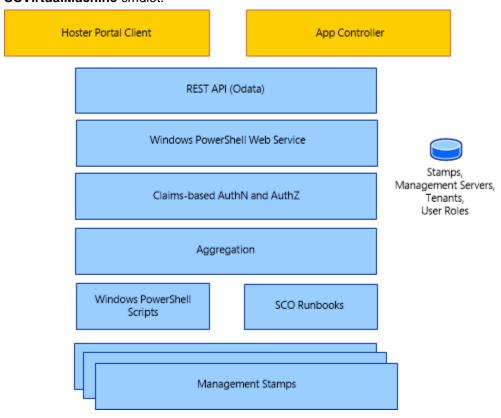
The tenant represents a hoster's customer, and the tenant has assets on the hoster's system. Each tenant has their own administrators, applications, scripts, and other tools.

The hoster provides tenants with the environment, which can include virtual machines. The hoster has an existing front-end portal, which all tenants can use. On the back end, the hoster has a

collection of resources, which is called the *fabric*. The hoster allocates those resources into discrete groups according to the hoster's needs. Each of these groups is known as a *stamp*. The hoster can then assign the tenant's resources to stamps in whatever manner is appropriate to the hoster. The resources may be divided across several stamps, according to the hoster's business model scheme. Service Provider Foundation makes it possible for the hoster to present a seamless user experience to the tenant by aggregating the data from each stamp and allowing the tenant to use the Service Provider Foundation application programming interfaces (APIs) to access that data.

A stamp in Service Provider Foundation is a logical scale unit designed for scalability that provides an association between a server and its System Center 2012 Service Pack 1 (SP1) components. As tenant demand increases, the hoster provides additional stamps to meet the demand. Note that Service Provider Foundation System Center 2012 SP1 supported only one type of stamp; that is a single server that has Virtual Machine Manager (VMM) installed.

Service Provider Foundation does not configure clouds; instead, it manages their resources. Virtual machines are set to clouds, for example, when they are created for VMM or when they are created by the **T:Microsoft.SystemCenter.VirtualMachineManager.Cmdlets.New-SCVirtualMachine** cmdlet.



The hoster can have a portal client, which faces the tenant, that provides access to the infrastructure that the hoster has granted. The portal uses an extensible representational state

transfer (REST) API to communicate with the web service by using the OData protocol. The claims-based authentication verifies the tenant's identity and associates it with the user role that the hoster assigns.

Service Provider Foundation uses a database to aggregate the tenant resources, which are managed with Windows PowerShell scripts and Orchestrator runbooks. This makes it possible for the hoster to distribute tenant resources among management stamps in whatever way it decides, while to the tenant the resources are easy to access and appear contiguous.

See Also

Deploying Service Provider Foundation

Administering Service Provider Foundation

Integrating Service Management Portal and API with System Center 2012 SP1

<u>Cloud Resource Management with System Center 2012 Service Pack 1 (SP1) – Orchestrator and Service Provider Foundation</u>

Cmdlets in System Center 2012 - Service Provider Foundation

Service Provider Foundation Developer's Guide

Deploying Service Provider Foundation

This section describes planning and deploying Service Provider Foundation.

In this section

- <u>Security Planning for Service Provider Foundation</u>
 Provides security guidance information for Service Provider Foundation.
- System Requirements for Service Provider Foundation for System Center 2012 SP1
 Lists the prerequisites to install the first version of Service Provider Foundation.
- System Requirements for Service Provider Foundation 2012 R2
 Lists the prerequisites to install the current version of Service Provider Foundation.
- Capacity Planning for Service Provider Foundation
 Provides hardware and database storage recommendations.
- How to Create an SSL Certificate for Testing Service Provider Foundation
 Describes how to create a testing certificate for installing Service Provider Foundation
 System Center 2012 SP1. You can skip this topic if you are installing the current version.
- How to Install Service Provider Foundation for System Center 2012 SP1
 Provides a detailed procedure for installing the first version of Service Provider Foundation.
- How to Install Service Provider Foundation 2012 R2

Provides a detailed procedure for installing the current version of Service Provider Foundation.

• Upgrading to Service Provider Foundation for System 2012 R2

Provides important information to consider before installing the current version of Service Provider Foundation.

- Setup Command-Line Options for Service Provider Foundation
 - Provides the commands for automating Service Provider Foundation installations.
- Post-Installation Tasks for Service Provider Foundation
 - Describes essential tasks after you install Service Provider Foundation.
- How to Uninstall Service Provider Foundation
 - Describes how to uninstall Service Provider Foundation from the Control Panel.
- Release Notes for Service Provider Foundation for System Center 2012 SP1
 - Provides the release notes for Service Provider Foundation System Center 2012 SP1.
- Release Notes for Service Provider Foundation 2012 R2
 - Provides the release notes for Service Provider Foundation System Center 2012 R2

See Also

Architecture Overview of Service Provider Foundation

Administering Service Provider Foundation

Cmdlets in System Center 2012 - Service Provider Foundation

Service Provider Foundation Developer's Guide

System Requirements for Service Provider Foundation for System Center 2012 SP1

The following tables describe the prerequisites for installing Service Provider Foundation for System Center 2012 Service Pack 1 (SP1), the first release of Service Provider Foundation. For requirements for the current version, see **System Requirements for Service Provider Foundation for System Center 2012 R2**.

For information on capacity planning for database storage, memory, and core processors, see Capacity Planning for Service Provider Foundation.

Required hardware and settings

Hardware	Minimum required and recommended values, per web service
RAM	2 gigabytes (GB) minimum, 4 GB preferred.

Hardware	Minimum required and recommended values, per web service
Available hard disk space	1 GB minimum, 3 GB preferred.

Required operating system and software

Software	Action required
Operating system - Windows Server 2012	Install on the server.
Secure Sockets Layer (SSL) server certificate for the Service Provider Foundation website	Obtain or create an SSL server certificate before installation. Applicable certificates will appear in the Service Provider Foundation setup wizard on the Specify a location for the SPF files page. Any preselected certificate may or may not be the most applicable certificate for your environment. You can obtain a certificate in the following ways: Purchase and import a certificate from a certification authority. Import an Active Directory Certificate. Create a self-signed certificate. For more information, see How to Create an SSL Certificate for Testing Service Provider Foundation.
Microsoft ASP.NET Model View Controller (MVC) 4	Install from the Microsoft Download Center.
 Microsoft SQL Server 2008 R2 or SQL Server 2012; supported editions include: SQL Server 2008 R2: Standard, Datacenter, and Enterprise. SQL Server 2012: Standard, Business Intelligence, and Enterprise. 	Install if needed. Although it is not required to install Service Provider Foundation, SQL Server is required on at least one server to contain the Service Provider Foundation database. The Setup program detects the existing installations of SQL Server and configures the use of any existing Service Provider Foundation database.
Windows Communication Foundation (WCF) Data Services 5.0 for Open Data Protocol (OData) V3	Install from the Microsoft Download Center.
System Center 2012 – Virtual Machine Manager (VMM) Console	Install from the Microsoft System Center Virtual Machine Manager 2012 Setup Wizard; see

Software	Action required
Although you do not need to install the full Virtual Machine Manager (VMM), you will need to make references to a server that has Virtual Machine Manager running and that can be accessed by supplied credentials.	How To Install the VMM Console. This is required so that Setup can install the VMM web service.
Management OData Internet Information Services (IIS) Extension	Add this feature in Windows Server 2012 Server Manager.
Windows Process Activation service. This features includes: Process Model Configuration application programming interfaces (APIs)	Add this feature in Windows Server 2012 Server Manager.
 Web Server (IIS). This server role includes: IIS Management Scripts and Tools Role Service IIS Security Basic Authentication IIS Application Deployment ASP.NET 4.5 IIS Security Windows Authentication Internet Server API (IASPI) extensions and filters ASP.NET 4.5 Role Service 	Add this role in Windows Server 2012 Server Manager if not already installed.
Windows PowerShell 3.0	None. Installed with Windows Server 2012.
Microsoft .NET Framework 3.5. This feature includes: • ASP.NET 3.5 • Common Language Runtime 2.0	None. Installed with Windows Server 2012.
.NET Framework 4.5. This feature includes: • ASP.NET 4.5 • WCF services HTTP Activation • Common Language Runtime 4.5	None. Installed with Windows Server 2012.

See Also

Capacity Planning for Service Provider Foundation

How to Install Service Provider Foundation for System Center 2012 SP1

Deploying Service Provider Foundation

Security Planning for Service Provider Foundation

This topic provides an overview of Service Provider Foundation security features and describes the security considerations for your deployment. You should create any required accounts and groups and determine if you have any additional security requirements before you start your Service Provider Foundation installation.

Security features

Service Provider Foundation provides a tightly coordinated implementation of Windows and Internet Information Services (IIS) security features. Note that credentials in a domain in the Active Directory must be used.

Service Provider Foundation relies on IIS to authenticate users. Starting with System Center 2012 R2, Service Provider Foundation accepts only the Secure Sockets Layer (SSL) requests protocol from its provider endpoints using the default port of 8090. Only HTTPS requests are accepted. Typically, the request should have the security context of the user who is logged on to the make the request.

When the setup wizard installs a web service, it creates a local security group on the computer that runs the web service. You can specify users or groups that have access to each web service. The wizard assigns those users or groups to a local security group. Service Provider Foundation checks that the user who sends the request belongs to the appropriate local security group.

In addition the wizard creates application domains pools in Internet Information Services (IIS) for each web service. You can specify the Network Service account or an account that also belongs to the security group.

The wizard creates the following security groups application pools as shown on the following table.

Security Group Name	Application Pool Name
SPF_Admin	Admin
SPF_Provider	Provider
SPF_VMM	VMM
SPF_Usage	Usage

After you install Service Provider Foundation, you must verify that the credentials for System Center 2012 – Virtual Machine Manager and the other service providers are configured correctly, as described in **Managing the Service Provider Foundation Web Services**.

See Also

Capacity Planning for Service Provider Foundation

How to Install Service Provider Foundation for System Center 2012 SP1

Setup Command-Line Options for Service Provider Foundation

Deploying Service Provider Foundation

Administering Service Provider Foundation

<u>Architecture Overview of Service Provider Foundation</u>

How to Install Service Provider Foundation for System Center 2012 SP1

You can install Service Provider Foundation on a single server or on multiple servers, with at least one server that has Microsoft SQL Server installed to contain the Service Provider Foundation database.

The Setup wizard configures an instance of Service Provider Foundation, along with the web services that you select for that computer. However, at this time, only the System Center 2012 – Virtual Machine Manager (VMM) web service is available for deployment. Installation of Service Provider Foundation onto a virtual machine is supported.

Before you install Service Provider Foundation, do the following:

- Make sure that each computer has sufficient RAM and hard disk space for all the web services that you intend to install. Also, be sure to have the prerequisite software installed.
 For more information, see <u>System Requirements for Service Provider Foundation for System</u> Center 2012 SP1.
- Make sure that you have a domain user account with administrative privileges on the computers on which you want to install Service Provider Foundation.
- Close any open programs, and make sure that the computer does not have a restart pending.

If there is a problem with the installation completing successfully, refer to the log files, named "Microsoft Service Provider*.log", in the %SYSTEMDRIVE%\%TEMP% folder.

You can also run a silent, unattended, installation. For more information, see <u>Setup Command-Line Options</u> for Service Provider Foundation.

To install Service Provider Foundation

 On the server where you want to install Service Provider Foundation, double-click SetupOrchestrator.exe on the installation media to start the System Center 2012 - Orchestrator Setup Wizard.



Note

We recommend that you run setup as Administrator. Doing so allows Customer Experience and Microsoft Update choices to be retained later in the setup.

- 2. On the main Setup page, click Service Provider Foundation.
- 3. On the Service Provider Foundation Setup page, click Install.
- 4. On the License Terms page, review the license agreement. If you agree with the terms, select the I have read, understood, and agree with the terms of the license agreement check box, and then click Next.
- 5. On the Select the web services to install page, select the check box for System Center Virtual Machine Manager 2012 Web Service, and then click Next.
- 6. On the **Prerequisites** page, wait for the wizard to complete the prerequisite verification, and then review the results. If any of the prerequisites are missing, install the missing prerequisites, and then click Check prerequisites again.
 - When all of the prerequisites are met, click Next.
- 7. On the Configure the database server page, in the server text box, enter the name of the server that hosts SQL Server, or accept the default localhost. In Port Number, type the port number that accesses the database, or accept the default of 1433, and then click Next.
- 8. On the Specify a location for the SPF files page, accept or change the location for the web service files by using the **Change Folder** button. Optionally, change **Website name**. In the Port Number section, enter the Internet Information Services (IIS) port number that you want to use, or accept the default of 8090.



Note

If you want to change the IIS port that you assign during the installation of Service Provider Foundation, you must uninstall or reinstall Service Provider Foundation.

The certificate store and name refers to the certificate that was used to configure the site bindings for the Service Provider Foundation website in Internet Services Information (IIS) Manager. The currently selected certificate may or may not be the most applicable certificate for your environment. For more information, see How to Create an SSL Certificate for Testing Service Provider Foundation.

Click Next.

9. On the Configure the Admin web service page, in the Domain security groups or users text box, type the domain and user name of each security group or user who will use this web service. Use the format domain\user name, and use a semicolon to separate multiple entries, for example, CONTOSO\JohnDoe; CONTOSO\TestGroup.

For application pool credentials, select the type of account that you want to use:

To use an internal system account, select **Network Service**. If you select Network Service, the machine account must be a System Center 2012 - Virtual Machine Manager (VMM) administrator, or it must have enough permissions to perform the Service Provider Foundation requests.

To use a different account, select Service Account, and then type the domain name, user name, and password of the account that you want the application pool to use.
 Make sure that the application pool account exists in the domain and that it has sufficient permissions to manage the server.

Click Next.

- 10. In the same manner, specify the settings for **Configure the Provider web service**, and then click **Next**.
- 11. In the same manner, specify the settings for **Configure the VMM web service**, and then click **Next**.
- 12. Choose the desired options on the **Help improve Microsoft System Center Service Provider Foundation** and **Microsoft Update** page, and then click **Next**.

Choices made on this page are not retained unless setup was run as Administrator.

- 13. On the **Installation summary** page, review your selections, and then do one of the following:
 - Click Previous to change any selections.
 - Click Install to install Service Provider Foundation.

After you click **Install**, the installation progress indicator appears.

14. Click **Close** when the message "Setup is complete" appears.

Repeat this procedure for each installation, such as for a web farm.

To upgrade from previous installations

- 1. Stop all web services and portal applications using Service Provider Foundation.
- 2. Uninstall Service Provider Foundation using **Control Panel**. For more information, see How to Uninstall Service Provider Foundation.

The Service Provider Foundation database remains in place, including any extensions and application programming interface (API) resources that were added.

Repeat this step for each installation, such as for a web farm.

3. Install the new version of Service Provider Foundation.

On the **Configure the database server** page, specify the name of name of the server that has the Service Provider Foundation database.

Make any other changes that may be required for the installation, such as on the **Configure the Admin web service** page.

Repeat this step for each installation, such as for a web farm.

To enable the use of Service Provider Foundation with a portal applications

 See <u>Configuring Portals for Service Provider Foundation</u> for instructions on configuring Service Provider Foundation with Windows Azure Pack for Windows Server and API and App Controller.

See Also

Setup Command-Line Options for Service Provider Foundation

How to Uninstall Service Provider Foundation

Post-Installation Tasks for Service Provider Foundation

Deploying Service Provider Foundation

Administering Service Provider Foundation

Architecture Overview of Service Provider Foundation

Setup Command-Line Options for Service Provider Foundation

You can use command-line options with the Service Provider Foundation setup to perform an unattended installation by using the **–Silent** option. All options that you want to specify must be in a response file (text file) whose path is specified after the **–Silent** option, except for **–Silent – Uninstall** which performs an unattended uninstallation. The options can be delineated by either one line for each option, or a single space separating each option.

This topic requires that you have located the **setup.exe** file for Service Provider Foundation in the installation media for Orchestrator.

Setup Command-Line Options

The **-Silent** option must be specified followed by the name of the response file, or followed by the **-Uninstall** option:

setup.exe -Silent <ResponseFile>

setup.exe -Silent -Uninstall

The options that you can include in the response file are as follows. The first three options are required. The fourth option is required only if you specify the **SpecifyCertificate** option as true.

- -SendCEIPReports < true | false>
- -UseMicrosoftUpdate <true|false>
- -SpecifyCertificate <true|false>
- -CertificateSerialNumber < CertificateSerialNumberNoSpaces>

[-CertificateStore < Personal|WebHosting>]

[-InstallFolder < InstallFolder>]

[-WebSitePortNumber < PortNumber>]

[-DatabaseServer < ServerName>]

[-DatabasePortNumber < PortNumber>]

[-ScvmmUserName < ScvmmUserName>]

- [-ScvmmPassword < ScvmmPassword>]
- [-ScvmmDomain < ScvmmDomain>]
- [-ScvmmNetworkServiceSelected <true|false>]
- [-VmmSecurityGroupUsers < VmmSecurityGroupUsers>]
- [-ScadminUserName < scadminUserName>]
- [-ScadminPassword < scadminPassword>]
- [-ScadminDomain < scadminDomain>]
- [-ScadminNetworkServiceSelected <true|false>]
- [-AdminSecurityGroupUsers < AdminSecurityGroupUsers>]
- [-ScproviderUserName < scproviderUserName>]
- [-ScproviderPassword < scproviderPassword>]
- [-ScproviderDomain < scproviderDomain>]
- [-ScproviderNetworkServiceSelected < true | false>]
- [-ProviderSecurityGroupUsers < ProviderSecurityGroupUsers>]
- [-ScusageUserName < scusageUserName>]
- [-ScusagePassword < scpusagePassword>]
- [-ScusageDomain < scusageDomain>]
- [-ScusageNetworkServiceSelected <true|false>]
- [-usageSecurityGroupUsers < usageSecurityGroupUsers>]

The following table describes the command-line options:

Option	Description
-Silent	Performs an unattended installation.
	You must include the name of response file after the –Silent option that contains the other options. The options can be delineated by either one line for each option, or a single space separating each option.
	The following options are required to be specified in the response file:
	SendCEIPReports
	UseMicrosoftUpdate
	SpecifyCertificate
	CertificateSerialNumber (but only if SpecifyCertificate is true)
-Silent -Uninstall	Performs an unattended uninstallation.

Option	Description
-SendCEIPReports	Send anonymous reports to the Customer Experience Improvement Program.
-UseMicrosoftUpdate	Use Microsoft Update to check for updates.
-SpecifyCertificate	Specify true to use an existing certificate or false to automatically generate a self-signed certificate. If you specify true, you must also specify a value for the CertificateSerialNumber option.
-CertificateSerialNumber	The serial number of the certificate used by IIS for HTTPS authentication. Must not contain any spaces. You can omit this option if you specified the SpecifyCertificate option as false.
[-DatabaseServer <servername>]</servername>	The name of the server that contains the Service Provider Foundation database. Use localhost if the database is on the same computer on which you are installing Service Provider Foundation.
[-CertificationStore]	The store location of the certificate.
[-InstallFolder < InstallFolder>]	The path to the directory to install the product. The path must not contain any spaces.
[-WebSitePortNumber < PortNumber>]	The port number for the web service.
[-DatabasePortNumber < PortNumber>]	The port number of the database. The default is 1433.
[-ScvmmUserName < ScvmmUserName>] [-ScadminUserName < scadminUserName>] [-ScproviderUserName < scproviderUserName>] [-ScusageUserName < scusageUserName>]	The username credential for the Internet Information Services (IIS) setting for the VMM, Admin, Provider, or Usage web service. Omit this option if you are specifying true for the Network Service option.
[-ScvmmPassword < ScvmmPassword>] [-ScadminPassword < scadminPassword>] [-ScproviderPassword < scproviderPassword>] [-ScusagePassword < scusagePassword>]	The password credential for the IIS setting for the VMM, Admin, Provider, or Usage web service. Omit this option if you are specifying true for the Network Service option.
[-ScvmmDomain < ScvmmDomain>]	The domain credential for the IIS setting for the

Option	Description
[-ScadminDomain <scadmindomain>] [-ScproviderDomain <scproviderdomain>] [-ScusageDomain <scusagedomain>]</scusagedomain></scproviderdomain></scadmindomain>	VMM, Admin, Provider, or Usage web service. Omit this option if you are specifying true for the Network Service option.
[-ScvmmNetworkServiceSelected	Specifies that Network Service is to be used for application pool credentials for the VMM, Admin, Provider, or Usage web service. Specify false if you are specifying the username, password, or domain credential options.
[-VmmSecurityGroupUsers <vmmsecuritygroupusers>] [-AdminSecurityGroupUsers <adminsecuritygroupusers>] [-ProviderSecurityGroupUsers <providersecuritygroupusers>] [-usageSecurityGroupUsers <usagesecuritygroupusers>]</usagesecuritygroupusers></providersecuritygroupusers></adminsecuritygroupusers></vmmsecuritygroupusers>	Specifies one or more security groups or users (separated with a semicolon) for the VMM, Admin, Provider, or Usage web service.

Troubleshooting

• If you get the following error:

Error 0x80070003: Failed to write state to file: C:\ProgramData\Package Cache\{97585be5-93f3-41eb-8b19-34f5fe52879d}\state.rsm

Create a directory named "{97585be5-93f3-41eb-8b19-34f5fe52879d}" in the C:\ProgramData\Package Cache\ directory and run the setup command again.

See Also

How to Install Service Provider Foundation 2012 R2

How to Uninstall Service Provider Foundation

Deploying Service Provider Foundation

Administering Service Provider Foundation

Post-Installation Tasks for Service Provider Foundation

After installing, review **Manage Web Services and Connections in Service Provider Foundation** for important information about web services, credentials, and connections for using Service Provider Foundation.

As the administrator for a hosting provider, there are a few key tasks that you need to perform after you install Service Provider Foundation. You need to populate the Service Provider Foundation database sufficiently to start managing tenants. There are three ways to get started:

Portal applications

If you installed Service Provider Foundation to use with Windows Azure Pack for Windows Server, you can register the Service Provider Foundation web service endpoint and start provisioning virtual machine clouds and create plans for tenants. For more information, see Register the Service Provider Foundation Endpoint for Virtual Machine Clouds.

If you installed Service Provider Foundation to use with App Controller, you can connect to hosting provider. For more information, see **How to Connect to a Hosting Provider in System Center 2012 SP1**.

For detailed information about using portals, see **Portals in Service Provider Foundation**.

Service Provider Foundation Cmdlets

These Windows PowerShell cmdlets are suited for performing administrative tasks efficiently. For more information see <u>Cmdlets in System Center 2012 - Service Provider Foundation</u>. For the most current help in the console, run the following command.

```
PS C:\> update-help -module spfadmin
```

Program applications that consume Service Provider Foundation web services
 See the <u>Service Provider Foundation Developer's Guide</u>.

Populating the database

A basic, general procedure for populating the SCSPF database using cmdlets is as follows:

```
PS C:\> # Create a server.

PS C:\> $server = New-SCSPFServer -Name "server23G.contoso.com" -ServerType VMM

PS C:\> # Create a stamp. A stamp is a logical container for a tenant's association with one or more servers.

PS C:\> $stamp = New-SCSPFStamp -Name "StampA" -Servers $server

PS C:\> # Create a tenant. A tenant is your paying customer or business unit.

PS C:\> $tenant = New-SCSPFTenant -Name "jonathan@treyresearch.net"
```

PS C:\> # Associate the stamp to the tenant. You can set the stamp to the tenant and also to a different server if needed.

PS C:\> Set-SCSPFStamp -Stamp \$stamp -Tenants \$tenant

See Also

Deploying Service Provider Foundation

Administering Service Provider Foundation

Architecture Overview of Service Provider Foundation

Integrating Service Management Portal and API with System Center 2012 SP1

How to Uninstall Service Provider Foundation

When you uninstall Service Provider Foundation, you remove all Service Provider Foundation features, including all web services that are associated with Service Provider Foundation.

You can also run a silent, unattended, uninstallation. For more information, see <u>Setup Command-</u> Line Options for Service Provider Foundation.

You must use a domain user account with administrative privileges on the computers on which you want to uninstall Service Provider Foundation.

If there is a problem with the uninstallation, consult the log files in the **%SYSTEMDRIVE%\%TEMP%** folder in which you want to uninstall Service Provider Foundation.

When you uninstall Service Provider Foundation, you can keep or remove the Service Provider Foundation database.

To uninstall Service Provider Foundation and all associated web services

- 1. On the computer on which Service Provider Foundation is installed, click **Start**, and then click **Control Panel**.
- 2. In Programs, click Uninstall a program.
- 3. Under Name, right-click System Center 2012 R2 Service Provider Foundation (or an earlier version), and then click Uninstall.
- 4. On the **Summary** page, review your selections and do one of the following:
 - Click Previous to change any selections.
 - Click Uninstall to uninstall Service Provider Foundation.

After you click **Uninstall**, the **Uninstalling web services** page appears and an uninstallation progress indicator appears.

5. After Service Provider Foundation is uninstalled, on The selected components were

removed successfully page, click Close.

See Also

How to Install Service Provider Foundation 2012 R2

Setup Command-Line Options for Service Provider Foundation

Deploying Service Provider Foundation

Administering Service Provider Foundation

Architecture Overview of Service Provider Foundation

Release Notes for Service Provider Foundation for System Center 2012 SP1

These release notes apply to Service Provider Foundation, a component of System Center 2012 - Orchestrator. They contain up-to-date information about known issues that you might experience.

Known Issues

Installation fails on a computer with only IPv6 enabled

Description: Installation of Service Provider Foundation fails on a computer that has only IPv6 enabled for the network adapters.

Workaround: Enable IPv4.

You cannot move a virtual machine across clouds

Description: Service Provider Foundation cannot accommodate moving a virtual machine from

one cloud to another.

Workaround: None. This is a known limitation.

A new virtual machine is unusable if it is created without a guest operating system

Description: You can create a new virtual machine from a template by using a blank virtual hard disk (VHD). However, a 13206 error appears in the summary page. The virtual machine will be unusable if it is created with a VHD that contains no guest operating system.

Workaround: Create the virtual machine with a guest operating system.

Creating a virtual disk drive from a large VHD results in timeout exceptions

Description: Sporadic timeout exceptions might occur when you are creating a virtual disk drive with a large VHD.

Workaround: Retry the operation.

Creation of a user role fails occasionally

Description: In attempting to create a new user role on a second Virtual Machine Manager (VMM) server, the administrator user role might be unable to create a new user role, because the attempt is treated as an accidental impersonation.

Workaround: Recycle the VMM application pool.

An endpoint was registered successfully, but it cannot be registered as a Virtual Machine Cloud Provider

Description: Service Provider Foundation has been successfully installed, the endpoint can be accessed, but the service will not register as a Virtual Machine Cloud Provider.

Workaround: Copy the required files to the Service Provider Foundation endpoint website.

Updating a tenant to a new issuer and key does not work

Description: If you create a tenant with a specified key and issuer name and then update that same tenant (such as by using the **T:Microsoft.SystemCenter.Foundation.Cmdlet.Set-SCSPFTenant** cmdlet) to a new key and issuer name, the new key for the tenant will not be updated in the database.

Workaround: Recreate the tenant as shown in the following example.

```
PS C:\> $tenant = Get-SCSPFTenant -Name "TenantToFix"

PS C:\> $issuerName = (Get-SCSPFTrustedIssuer -Tenant $tenant).Name
```

```
PS C:\> $stamps = Get-SCSPFStamp -Tenant $tenant

PS C:\> $userroles = Get-SCSPFTenantUserRole -Tenant $tenant | Select-Object Name

PS C:\> Remove-SCSPFTenant -Tenant $tenant

PS C:\> $tenant2 = New-SCSPFTenant -Name "FixedTenant" -IssuerName $issuerName -Key $key2 -Stamps $stamps

PS C:\> $userroles | foreach {New-SCSPFTenantUserRole -Tenant $tenant2 -Name $ .Name}
```

See Also

Deploying Service Provider Foundation

Administering Service Provider Foundation

Architecture Overview of Service Provider Foundation

Administering Service Provider Foundation

A tenant in Service Provider Foundation is a customer of a hoster, and the customer is maintained in the database together with its status, metadata, and with one or more of the following associations:

To a stamp.

A stamp in Service Provider Foundation is a logical scale unit designed for scalability that provides an association between a server and its System Center 2012 Service Pack 1 (SP1)components. As tenant demand increases, the hoster provides additional stamps to meet the demand. Currently, Service Provider Foundation supports only one type of stamp; that is a single server that has Virtual Machine Manager (VMM) installed.

• To a trusted issuer and a public key.

A public key to a certificate and the name of the trusted issuer can be specified for a tenant when the tenant is created.

To an offer.

Offers provide associations for a provider's plan to stamps and tenants.

• To tenant security user roles.

A Tenant Administrator Role and one or more Tenant Self-Service user roles can be associated with a tenant.

Administering topics

Recommended Administrator Capabilities in Service Provider Foundation
 Specifies recommended permissions for Service Provider Foundation administrators, database administrators, and application pool users.

Manage Web Services and Connections in Service Provider Foundation

Provides a comprehensive overview of the web services, credentials, and connectivity required to administer Service Provider Foundation.

Manage Certificates and User Roles in Service Provider Foundation

Provides an overview of how multi-tenant security is implemented in Service Provider Foundation. This section contains a walkthrough topic with procedures on creating and managing a tenant's certificate and defining tenant administrator and tenant self-service user roles. In addition, topics describe recommended administrator capabilities and an example of a token authentication.

Portals in Service Provider Foundation

Describes how client and portal applications can communicate with and obtain services from Service Provider Foundation. This section also contains procedures for configuring App Controller and Windows Azure Pack for Windows Server and API.

Usage Metering in Service Provider Foundation

Describes how Service Provider Foundation provides usage metering data of virtual machine usage by tenants.

Extensibility in Service Provider Foundation

Describes how to have an runbook in System Center 2012 - Orchestrator invoked by Service Provider Foundation.

Other resources for this component

- TechNet Library main page for <u>Service Provider Foundation</u>
- Deploying Service Provider Foundation
- Architecture Overview of Service Provider Foundation
- Cmdlets in System Center 2012 Service Provider Foundation
- Service Provider Foundation Developer's Guide

Manage Certificates and User Roles in Service Provider Foundation

Service Provider Foundation provides a claims-based authentication security model for a tenant's access to services and resources. It registers the certificate's public key and issuer name from an issued certificate, and maintains that information as trusted issuer objects.

To provide secure multi-tenant operations, requests are performed in the context of a user role that maps a claim token from a tenant to a Tenant Administrator User Role or to a Tenant Self-service User Role. These user roles must be defined in System Center 2012 – Virtual Machine Manager (VMM) including their scope, resources, and actions.

Hoster administrators can use the Service Provider Foundation OData services to create the required infrastructure. For more information, see the <u>Service Provider Foundation Developer's</u> Guide.

A typical on-boarding tenant scenario is as follows:

- 1. A prospective tenant investigates a hoster's services by evaluating the offered plans.
- 2. The prospective tenant subscribes to a plan (offer objects in Service Provider Foundation), which generates a new subscription in a portal application and creates a new tenant in the Service Provider Foundation database.
 - During this process, a tenant uploads the public key for their certificate file. This lets the host to register the tenant and configure user security roles in Virtual Machine Manager.
- The portal applications and hoster administrators configure a tenant's connections to the hoster's service by using the service OData protocol URLs and tokens verified with the tenant's certificate that contains the private key.

Hoster administrators can also use the IDs generated by Service Provider Foundation cmdlets that create tenant or tenant user roles as the ID values for the corresponding VMM cmdlets that create actual user roles. The Service Provider Foundation cmdlets do the following:

- Generate the ID for a Tenant Administrator User Role when a tenant is created by using the T:Microsoft.SystemCenter.Foundation.Cmdlet.New-SCSPFTenant cmdlet.
- Generate the ID for a Tenant Self-Service User Role when a tenant user role is created by using the T:Microsoft.SystemCenter.Foundation.Cmdlet.New-SCSPFTenantUserRole cmdlet.

Multi-tenancy is additionally aided by new feature capabilities that are available in Windows Server 2012 such as Network Virtualization.

Managing certificates and user roles topics

- Walkthrough: Creating a Certificate and User Roles for Service Provider Foundation
 This walkthrough topic provides procedures for creating and accessing certificates, obtaining keys, and creating security user roles.
- Example SAML Assertion for Authentication in Service Provider Foundation Shows an example of a authentication for a token to access tenant resources.

Other resources for this component

- TechNet Library main page for Service Provider Foundation
- Administering Service Provider Foundation
- Deploying Service Provider Foundation

Walkthrough: Creating a Certificate and User Roles for Service Provider Foundation

This walkthrough shows how to administer important tasks for managing both certificates and user roles in Service Provider Foundation. To start, we show how to generate a self-signed certificate if you are not already working with an issuer's signed certificate. Next, we show how to obtain the certificate's public key, and how to use that key to create the tenant in Service Provider Foundation and user roles in System Center 2012 – Virtual Machine Manager (VMM).

This walkthrough is organized into the following sections and procedures. The procedures are designed to be performed sequentially, although they contain the information that you need to run them individually as needed. These procedures are tasks for the hoster administrator to perform.

Section	Procedures
Create a certificate	To create a self-signed certificate for a tenant
Obtain and export keys	To export the public key To export the private key To obtain the public key in Windows PowerShell
Create the tenant and its user roles	To create a tenant with the certificate's public key To create a tenant administrator role in VMM To create a tenant self-service user role

Create a certificate

The following procedure describes how to create a certificate for a tenant by using **makecert.exe** (Certificate Creation Tool).

To create a self-signed certificate for a tenant

- 1. Open a command prompt as administrator.
- 2. Generate the certificate by running the following command:

```
makecert -r -pe -n "cn=contoso.com" -b 07/12/2012 -e
09/23/2014 -ss My -sr CurrentUser -sp "Microsoft RSA SChannel
Cryptographic Provider" -sy 12 -sky exchange
```

This command puts the certificate in the Current User Certificate Store.

To access the certificate that you created

- 1. On the Start screen, type certmgr.msc and then in the Apps results click certmgr.msc.
- 2. In the **certmgr** window, click **Certificates Current User**, open the **Personal** folder, and then open the **Certificates** folder to view the certificate that you just generated.

Obtain and export keys

The procedures in this section show how to export public and private keys from certificate files. You associate a public key with a tenant in Service Provider Foundation to later validate claims made, or made on behalf of, a tenant. This section includes a procedure that shows how to obtain the public key directly in your PowerShell session.

To export the public key

- 1. Open your **certificates** folder to view the certificate as described in the <u>To access the</u> certificate that you created procedure.
- 2. Right-click the certificate, click **All Tasks**, and then click **Export**.
- 3. After the Welcome page, on the **Export Private Key** page, choose **No**, **do not export the private key** and then click **Next**.
- On the Export File Format page, select Base-64 encoded X.509 (.CER) and then click Next
- 5. On the **File to Export** page, specify a path and filename for the certificate and then click **Next**.
- 6. On the Completing the Certificate Export Wizard page, click Finish.

To export the private key

- 1. Open your **certificates** folder to view the certificate as described in the <u>To access the certificate that you created</u> procedure.
- 2. Right-click the certificate, click **All Tasks**, and then click **Export**.
- 3. After the **Welcome** page, on the **Export Private Key** page choose **Yes, export the private key** and then click **Next**.
 - If the Yes option is disabled, that is because the **makecert** command to create the certificate did not include the *-pe* option.
- On the Export File Format page, select the Personal Information Exchange PKCS #12 (.PFX) option, check the Include all certificates in the certification path if possible check box and then click Next.
- 5. On the **Security** page, select the **Password:** option, provide and confirm a password, and then click **Next**.
- 6. On the **File to Export** page, specify a path and filename for the certificate and then click **Next**.
- 7. On the Completing the Certificate Export Wizard page, click Finish.

To obtain the public key in Windows PowerShell

You can obtain the public key directly from an exported public key certificate file (.CER) by using the .NET Framework cryptography classes. Run the following commands to obtain the key from the certificate's public key file that you exported in the <u>To export the public key</u> procedure.

```
PS C:\> $path = "C:\Temp\tenant4D.cer"

PS C:\> $cert = New-Object
System.Security.Cryptography.X509Certificates.X509Certificate
2($path)

PS C:\> $key = [Convert]::ToBase64String($cert.RawData)
```

The next procedure uses the \$key variable that you just created.

Create the tenant and its user roles

Service Provider Foundation does not create user roles or define their scope (such as clouds), resources, or actions. Instead, the New-SCSPFTenantUserRole cmdlet creates an association for a tenant with a user role name. When that association is created, it also generates an ID that can be used for the corresponding ID for creating the role in System Center 2012 – Virtual Machine Manager.

You can also create user roles by using the Admin OData protocol service that uses the <u>Service Provider Foundation Developer's Guide</u>.

To create a tenant with the certificate's public key

- 1. Run the System Center 2012 Service Provider Foundation Command Shell as Administrator.
- 2. Enter the following command to create the tenant. This command assumes that the \$key variable contains the public key as obtained from the To obtain the public key in Windows PowerShell procedure.

```
PS C:\> $tenant = New-SCSPFTenant -Name
"contoso.cloudspace.com" -IssuerName "contoso.cloudspace.com"
-Key $key
```

3. Verify that the public key for the tenant was imported successfully by running the following command and viewing the results:

```
PS C:\> Get-SCSPFTrustedIssuer
```

The next procedure uses the \$tenant variable that you just created.

To create a tenant administrator role in VMM

1. Enter the following command and agree to this elevation for the Windows PowerShell command shell:

```
PS C:\> Set-Executionpolicy remotesigned
```

2. Enter the following command to import the Virtual Machine Manager module:

```
PS C:\> Import-Module virtualmachinemanager
```

3. Use the Windows PowerShell

T:Microsoft.SystemCenter.VirtualMachineManager.Cmdlets.New-SCUserRole cmdlet to create the user role. This command assumes the \$tenant variable was created as described in the To create a tenant with the certificate's public key procedure.

```
PS C:\> $TARole = New-SCUserRole -Name contoso.cloudspace.com
-ID $tenant.Id -UserRoleProfile TenantAdmin
```



Note that if the user role was previously created by using the VMM Administration Console, its permissions would be overwritten by those specified by the **New-SCSUserRole** cmdlet.

- 4. Verify that the user role was created by verifying that it is listed in the **User Roles** in **Settings** workspace in the VMM Administration Console.
- 5. Define the following for the role by selecting the role and clicking **Properties** on the toolbar:
 - On the Scope tab, select one or more clouds.
 - On the Resources tab, add any resources such as templates.
 - On the Actions tab, select one or more actions.

Repeat this procedure for every server assigned to the tenant.

The next procedure uses the \$TARole variable that you just created.

To create a tenant self-service user role

 Enter the following command to create a self-service user in Service Provider Foundation for the tenant you created in the <u>To create a tenant with the certificate's public key</u> procedure.

```
PS C:\> $TenantSSU = New-SCSPFTenantUserRole -Name
ContosoCloudSpaceSSU -Tenant $tenant
```

2. Create the corresponding tenant user role in VMM by entering the following command:

```
PS C:\> $vmmSSU = New-SCUserRole -Name
ContosoCloudSpaceVMMSSU -UserRoleProfile SelfServiceUser -
ParentUserRole $TARole -ID $TenantSSU.ID
```

3. Verify that the user role was created by verifying that it is listed in the **User Roles** in **Settings** workspace in the VMM Administration Console. Notice that the parent of the

role is the tenant administrator.

Repeat this procedure as needed for the tenant.

See Also

Manage Certificates and User Roles in Service Provider Foundation

Administering Service Provider Foundation

Recommended Administrator Capabilities in Service Provider Foundation

Configuring Portals for Service Provider Foundation

Recommended Administrator Capabilities in Service Provider Foundation

This topic provides guidelines for administrator capabilities and roles for administering Service Provider Foundation.

Roles for database administrators

A database administrator (DBA) has full administrator rights on SQL Server, and operates as the SQL Server administrator. This administrator should be able to grant permissions to create databases in SQL Server or grant those permissions to the Service Provider Foundation Administrator (SPFA). This administrator should be able to do the following:

- Create database named SCSPFDB. The default database is set to SCSPFDB.
- Create a SQL Server logon and user for the Service Provider Foundation Administrator, and grant the user the permissions described in this table.

Permissions	Purpose
Alter	To be able to create tables.
Connect with Grant	To connect to the existing database.
Select with Grant, Update with Grant, Delete with Grant, Insert with Grant	To grant these permissions to application users.
Alter All logins	To create SQL Server logins for the application pool users.

Roles for Service Provider Foundation administrators

A Service Provider Foundation administrator is the user responsible for installing Service Provider Foundation, and should have administrative rights on the server where Service Provider Foundation is to be installed.

There are two database scenario configurations:

- Install Service Provider Foundation by using a connection to an existing database.
 The Service Provider Foundation administrator must verify that the permissions were granted by the database administrator as described in the previous section.
- Create a new database.

The database administrator must create the database (SCSPFDB) and then the Service Provider Foundation administrator must install Service Provider Foundation and have permission to configure the database as needed such as to add tables. Service Provider Foundation administrators must create the Service Provider Foundation Application Pool in Internet Information Services (IIS) and create a database user for an Application Pool User with the following permissions:

Permission	Purpose
Connect	To be able to connect to the Service Provider Foundation database.
Select, Update, Delete, Insert	To be able to perform basic operations.
Create the SQL Server logon for Application Pool User with default database set to SCSPFDB.	To be able to log on to SQL Server and access this database.

Roles for Application Pool users

This is the Application Pool user in IIS who must have full administrative privileges in System Center 2012 – Virtual Machine Manager (VMM). These users should have the permissions to perform Create, Read, Update, and Delete operations on the Service Provider Foundation database. For portal applications, these operations can be restricted to specific tables.

See Also

Manage Certificates and User Roles in Service Provider Foundation

Administering Service Provider Foundation

Walkthrough: Creating a Certificate and User Roles for Service Provider Foundation

Configuring Portals for Service Provider Foundation

This topic describes how to configure Service Provider Foundation with the following portal applications:

- Windows Azure Pack for Windows Server
- App Controller

All portal and client applications use the Service Provider Foundation services to deliver laaS. For more information, see **Manage Web Services and Connections in Service Provider Foundation**.

Configuring Windows Azure Pack for Windows Server

Service Provider Foundation provides services and connectivity for delivering laaS for Windows Azure Pack.

To register Service Provider Foundation in Windows Azure Pack

- On the sever that has Service Provider Foundation installed, make a note of the credential used for the Admin, VMM, Usage, and Provider Application Pool identity in Internet Information Services (IIS). You will need this credential for registering the endpoint in Windows Azure Pack.
- Continue with the procedure in Register the Service Provider Foundation Endpoint for Virtual Machine Clouds in the Windows Azure Pack for Windows Server documentation.

Configuring App Controller

If a tenant was not created, you can follow the procedures for creating a tenant that are described in Walkthrough: Creating a Certificate and User Roles for Service Provider Foundation.

To connect to App Controller as a Tenant

The hoster administrator has to provide the tenant's ID to connect to App Controller.
 If you need to determine the ID, you can use the
 T:Microsoft.SystemCenter.Foundation.Cmdlet.Get-SCSPFTenant cmdlet to obtain the ID as shown in the following example:

```
PS C:\> Get-SCSPFTenant -Name "ADatum" | Format-List -
```

Property ID

Id : 4ce5713a-50a1-434b-b47a-87caad75ba72

Copy the ID.

- 2. Using the browser, connect to the App Controller management portal.
- 3. Sign in by using your Windows credentials.
- 4. Under **Settings**, click **Connections**, click **Connect**, then select **Service Provider Foundation**.
- 5. In the Add an External Provider Connection dialog box, specify the following values:
 - Connection name:

This is the name of the product or service that is used by the tenant.

Description:

This description is optional.

Service location:

This is the Service Provider Foundation OData protocol URI for the VMM service, as shown the following example. The URI ends with tenant's ID:

https://contoso.muchspace.com:8090/SC2012R2/vmm/Microsoft.Management.Odata.svc/4ce5713a-50a1-434b-b47a-87caad75ba72

If you are using Service Provider Foundation System Center 2012 SP1, remove the 'R2' from SC2012R2.

Certificate file:

This is the location that you specified for the exported certificate (typically with a .pfx extension). For information about how to export the private key from a certificate for this step, see the <u>To export the private key</u> procedure in <u>Walkthrough: Creating a</u> Certificate and User Roles for Service Provider Foundation.

Password:

This is the password that was set in the steps to export the private key certificate.

For more information about how to connect a hosting provider to App Controller, see **How to Connect to a Hosting Provider in System Center 2012 SP1**

See Also

Portals in Service Provider Foundation

Administering Service Provider Foundation

Privacy Statement for System Center 2012 - Orchestrator

Microsoft is committed to protecting your privacy, while delivering software that brings you the performance, power, and convenience you desire in your personal computing. This privacy statement explains many of the data collection and use practices of Microsoft System Center 2012 - Orchestrator Release to Manufacturing ("Orchestrator").

System Center 2012 - Orchestrator Release to Manufacturing empowers IT Pros to easily develop Runbooks for Microsoft System Center that interoperate with 3rd party systems to automate data center workloads. Runbooks are created and tested in the Runbook Designer, a rich drag-and-drop style user interface optimized for use by IT Pros. These Runbooks bring Microsoft System Center products and 3rd party systems together to automate data center workloads through the use of a unique and powerful Publish/Subscribe mechanism. Once created these Runbooks are deployed to distributed Runbook Servers where they can be managed using the Orchestration Console, a rich Microsoft Silverlight application that allows the IT Pro to identify Runbook issues, start and stop runbooks and explore the runtime environment.

Collection and Use of Your Information

The information we collect from you will be used by Microsoft and its controlled subsidiaries and affiliates to enable the features you are using and provide the service(s) or carry out the transaction(s) you have requested or authorized. It may also be used to analyze and improve Microsoft products and services.

We may send certain mandatory service communications such as welcome letters, billing reminders, information on technical service issues, and security announcements. Some Microsoft services may send periodic member letters that are considered part of the service. We may occasionally request your feedback, invite you to participate in surveys, or send you promotional mailings to inform you of other products or services available from Microsoft and its affiliates.

In order to offer you a more consistent and personalized experience in your interactions with Microsoft, information collected through one Microsoft service may be combined with information obtained through other Microsoft services. We may also supplement the information we collect with information obtained from other companies. For example, we may use services from other companies that enable us to derive a general geographic area based on your IP address in order to customize certain services to your geographic area.

Except as described in this statement, personal information you provide will not be transferred to third parties without your consent. We occasionally hire other companies to provide limited services on our behalf, such as packaging, sending and delivering purchases and other mailings, answering customer questions about products or services, processing event registration, or performing statistical analysis of our services. We will only provide those companies the personal information they need to deliver the service, and they are prohibited from using that information for any other purpose.

Microsoft may access or disclose information about you, including the content of your communications, in order to: (a) comply with the law or respond to lawful requests or legal process; (b) protect the rights or property of Microsoft or our customers, including the enforcement of our agreements or policies governing your use of the services; or (c) act on a good faith belief that such access or disclosure is necessary to protect the personal safety of Microsoft employees, customers, or the public. We may also disclose personal information as part of a corporate transaction such as a merger or sale of assets.

Information that is collected by or sent to Microsoft by Orchestrator may be stored and processed in the United States or any other country in which Microsoft or its affiliates, subsidiaries, or service providers maintain facilities. Microsoft abides by the safe harbor framework as set forth by the U.S. Department of Commerce regarding the collection, use, and retention of data from the European Union, the European Economic Area, and Switzerland.

Collection and Use of Information about Your Computer

When you use software with Internet-enabled features, information about your computer ("standard computer information") is sent to the Web sites you visit and online services you use. Microsoft uses standard computer information to provide you Internet-enabled services, to help improve our products and services, and for statistical analysis. Standard computer information typically includes information such as your IP address, operating system version, browser version, and regional and language settings. In some cases, standard computer information may also include hardware ID, which indicates the device manufacturer, device name, and version. If a particular feature or service sends information to Microsoft, standard computer information will be sent as well.

The privacy details for each Orchestrator feature, software or service listed in this privacy statement describe what additional information is collected and how it is used.

Security of your information

Microsoft is committed to helping protect the security of your information. We use a variety of security technologies and procedures to help protect your information from unauthorized access, use, or disclosure. For example, we store the information you provide on computer systems with limited access, which are located in controlled facilities.

Changes to this privacy statement

We will occasionally update this privacy statement to reflect changes in our products, services, and customer feedback. When we post changes, we will revise the "last updated" date at the top of this statement. If there are material changes to this statement or in how Microsoft will use your personal information, we will notify you either by posting a notice of such changes prior to

implementing the change or by directly sending you a notification. We encourage you to periodically review this statement to be informed of how Microsoft is protecting your information.

For More Information

Microsoft welcomes your comments regarding this privacy statement. If you have questions about this statement or believe that we have not adhered to it, please contact us.

scorchprivacy@microsoft.com Microsoft Corporation One Microsoft Way Redmond, WA, 98052 USA

Specific Features

The remainder of this document will address the following specific features:

Workflow Data Logging

What this feature does:

Workflow activities that interact with systems can be configured to collect data that would be considered private.

Information collected, processed, or transmitted:

Examples of such systems would be Active Directory, BMC Remedy, and so on. By default, this data (referred to as "Object Specific Published Data" in the product) is not logged; however, the Enterprise can select an option to "Log Object Specific Published Data" in which case this data will be logged to the database and visible in the Operations Console/Designer/etc.

Use of information:

This information is not sent outside of the Enterprise.

Choice/Control:

To enable or disable this feature:

- 1. In the Runbook Designer, in the **Connections** pane, click the **Runbooks** folder.
- 2. If the runbook is stored in a folder, select the appropriate folder under Runbooks.
- 3. In the Design workspace of Runbook Designer, right-click the tab of a runbook to select **Properties**.
- 4. On the **Logging** tab of the **Runbook Properties** dialog box:

- a. Select the **Store Activity-Specific Published Data** check box to enable this feature. Clear the check box to disable the feature.
- b. Select the **Store Common Published Data** check box to enable this feature. Clear the check box to disable the feature.
- 5. To close the **Runbook Properties** dialog box, click Finish.

Important information

Workflows have a revision history associated with them. This history is stored in the database along with the Active Directory SID of the user that created the changes. This feature cannot be disabled.

Workflow Exports

What this Feature Does:

The Enterprise can export policies and other Orchestrator configuration information into an XML-formatted file intended to be used as a policy export/import mechanism.

Information collected, processed, or transmitted:

These exports will contain information about the workflows as required to import them at a subsequent date. Any information stored in an Orchestrator workflow would be present in the export. This would include form-field data configured into workflow objects ("Activities") as well as workflow owner information such as SIDs from Active Directory.

Use of information:

This information is not sent outside of the Enterprise.

Choice/Control

The export/import feature is only present in the Orchestrator Run Book Designer (the "Designer") and you may opt not to use this feature.

Interaction with Third Party Systems

What this feature does:

Orchestrator Integration Packs allow the Enterprise to interact with third party systems as part of an overall workflow.

Information collected, processed, or transmitted:

Orchestrator does not collect or send any such data, however third party systems might. Orchestrator may interact with third party systems such as to cause them to send information outside the Enterprise.

Use of information:

Orchestrator does not send any information outside the enterprise. Such information would be sent by a third party system in response to activity driven by Orchestrator.

Choice/Control:

Review the privacy features and policies of the third party systems that are interacting with Orchestrator. If a third party system does in fact transmit this sort of information as part of interacting with Orchestrator, consult the privacy statement as well as other documentation provided by the third party system vendor for instructions on how to disable such transmission.

Customer Experience Improvement Program

What this feature does:

The Customer Experience Improvement Program ("CEIP") collects basic information about your hardware configuration and how you use our software and services in order to identify trends and usage patterns. CEIP also collects the type and number of errors you encounter, software and hardware performance, and the speed of services. We will not collect your name, address, or other contact information.

Information collected, processed, or transmitted:

For more information about the information collected, processed, or transmitted by CEIP, see the CEIP privacy statement at http://www.microsoft.com/products/ceip/EN-US/privacypolicy.mspx.

Use of information:

We use this information to improve the quality, reliability, and performance of Microsoft software and services.

Choice/Control:

You are offered the opportunity to participate in CEIP during setup. You can change this setting after install through the Runbook Designer Help/About where an opt-in/opt-out is available.

Microsoft Error Reporting

What this feature does:

Microsoft Error Reporting provides a service that allows you to report problems you may be having with Orchestrator to Microsoft and to receive information that may help you avoid or solve such problems.

Information collected, processed, or transmitted:

For information about the information collected, processed, or transmitted by Microsoft Error Reporting, see the Microsoft Error Reporting privacy statement athttp://oca.microsoft.com/en/dcp20.asp.

Use of information:

We use the error reporting data to solve customer problems and improve our software and services.

Choice/Control:

You will be offered the opportunity to participate in Microsoft Error Reporting during install only. We also offer you the ability to queue reports prior to sending. You can change this setting after install through the Runbook Designer Help/About where an opt-in/opt-out is available.

When you choose to enable it, Microsoft Error Reporting will automatically report problems you encounter to Microsoft. When Microsoft needs additional data to analyze the problem, you will be prompted to review the data and choose whether or not to send it.

Important information:

Enterprise customers can use Group Policy to configure how Microsoft Error Reporting behaves on their computers. Configuration options include the ability to turn off Microsoft Error Reporting. If you are an administrator and wish to configure Group Policy for Microsoft Error Reporting, technical details are available at http://go.microsoft.com/fwlink/?LinkID=228751.

Microsoft Update

What this feature does:

Microsoft Update is a service that provides Windows updates as well as updates for other Microsoft software.

Information collected, processed, or transmitted:

For details about what information is collected and how it is used, see the Update Services Privacy Statement at http://go.microsoft.com/fwlink/?LinkID=228752.

Use of information:

For details about what information is collected and how it is used, see the Update Services Privacy Statement at http://go.microsoft.com/fwlink/?LinkID=228752.

Choice/Control:

Microsoft Update is not turned on as a default. It is controlled by a choice you make during the setup. You may later change it by accessing the Microsoft Update client under your Control Panel to turn updates on or off.

Integration Packs from third parties (non-Microsoft authored)

What This Feature Does:

Integration Packs from third parties extend the core Orchestrator 2012 platform to include new runbook activities not available from Microsoft integration packs.

Information Collected, Processed, or Transmitted:

For details about what information is collected and how it is used, refer to the privacy statement of the third party who provided the integration pack.

Use of Information:

For details about what information is collected and how it is used, refer to the privacy statement of the third party who provided the integration pack.

Choice/Control:

For details about what information is collected and how it is used, refer to the privacy statement of the third party who provided the integration pack.

Release Notes for System Center 2012 - Orchestrator

The following release notes apply to the appropriate version of Orchestrator in System Center 2012, and they contain descriptions and workarounds for known issues.

There are three versions of these release notes:

- Release Notes for System Center 2012 Orchestrator
- Release Notes for Orchestrator in System Center 2012 SP1
- Release Notes for Orchestrator in System Center 2012 R2

See Also

Orchestrator

Release Notes for System Center 2012 - Orchestrator

These release notes contain information that is required to successfully install System Center 2012 - Orchestrator. They contain information that is not available in the product documentation.

Before you install and use Orchestrator, read these release notes. These release notes apply to System Center 2012 - Orchestrator.

If you are looking for the Release Notes for Orchestrator in System Center 2012 Service Pack 1 (SP1), see Release Notes for Orchestrator in System Center 2012 SP1.

Known Issues

You receive a database validation error when you use a remote computer that is running SQL Server

Description: If you are using a remote computer that is running Microsoft SQL Server and that server has named pipes enabled (as opposed to TCP/IP), you cannot successfully install Orchestrator. Instead, you receive a database validation error during the last phase of installation.

Workaround: Enable TCP/IP for any Orchestrator installations that use a remote computer that is running SQL Server.

You must uninstall older versions of Orchestrator before you install the System Center 2012 - Orchestrator runbook server

Description: If you try to install or deploy a System Center 2012 - Orchestrator runbook server on a computer that has the Opalis Action Server, the Orchestrator 2012 Beta, or the Orchestrator 2012 Release Candidate runbook server installed, it leaves the runbook server in an unusable state. You must revert the deployment of the previous version by using the **Deployment**Manager, or in Control Panel, uninstall it by using **Programs and Features** before you install the new version.

You must also manually remove the OpalisRemotingService executable program by using the following procedure:

To remove the OpalisRemotingService

- 1. Open a command prompt with administrative credentials.
- 2. Stop the service by entering the command sc stop OpalisRemotingService.
- 3. Remove the service by entering the command sc delete OpalisRemotingService.
- 4. Navigate to C:\Windows\SysWOW64.
- 5. Delete the file OpalisRemotingService.exe.

The Orchestrator setup wizard may crash on an unsupported operating system

Description: If you run the Orchestrator setup wizard on an unsupported operating system, then you may receive an unexpected error message or system crash.

Workaround: See <u>System Requirements</u> for details on operating systems supported by Orchestrator.

The computer restarts after you deploy Runbook Designer

Description: When you deploy a Runbook Designer to localhost through the Deployment Manager, the computer will restart.

Workaround: None.

Authorization cache performance

Description: In the Release Candidate for System Center 2012 - Orchestrator, permission changes to runbooks and runbook folders were immediately propagated to the Orchestrator web service and the Orchestration console. If you added, imported, or deleted a runbook or a runbook folder, or changed the permissions on a runbook or runbook folder from the Runbook Designer, the changes were immediately visible in the Orchestration console.

The authorization cache table is included in the release version of Orchestrator. This table is cleared and Orchestrator re-computes the permissions every 10 minutes. You cannot view any

runbook or runbook folder changes until the cache is refreshed. After 10 minutes, you can refresh the Orchestration console and see the changes.

Workaround: It is not recommended to reduce the refresh interval of the authorization cache table because of the time required to re-compute the cache. If you require assistance in modifying the refresh interval of the authorization cache table, please contact customer support.

Certain automatic log purge settings do not work

Description: When you select Purge the logs from the Log History tab of a runbook, all of the logs for that runbook are deleted. The automatic log purge that occurs daily and that can be triggered manually by right clicking on the Server in the Connection pane generates an error. If you configure the Log Purge Options to Keep most recent entries, an error occurs and the log purge does not occur. If the error occurs during the scheduled daily log purge, the error is written to log history.

Workaround: Use the If the number of entries exceeds X delete the entries older than Y option.

Certain log purge settings for runbooks may not work

Description: In some scenarios, log purge settings do not work. This occurs most commonly when you use the Invoke Runbook activity.

The following settings are affected:

Keep Last X entries

Orchestrator cannot determine the relationship between Id and Parentld so the setting fails when you try to delete an Id that is also a Parentld.

Keep entries for last X days

Orchestrator cannot determine the relationship between Id and Parentld so the setting fails when you try to delete an Id that is also a Parentld.

If number of entries exceed X, delete entries older than Y days

The current behavior for this setting is identical to If total number of entries is greater than X, delete entries older than Y days.

Workaround: None.

Standard date/time variable format is yyyy-MM-ddTHH:mm:ss

Description: The standard date/time format used by Orchestrator is a 24-hour time format displayed as yyyy-MM-ddTHH:mm:ss. This date/time format conforms with ISO 8601.



Important

The variable string for the date/time format is case sensitive. If you use yyyy-MMddThh:mm:ss as the variable string, the wrong date/time is output.

Workaround: Use the format yyyy-MM-ddTHH:mm:ss.

The date/time format displayed in a property textbox is not always displayed in the locale-specific format

Description: Changes to the formatting of the date/time string can cause the date/time format in a property textbox to use a different format from the locale-specific format.

This occurs because an Orchestrator deployment can have a span of control that crosses different locales. You can choose to display dates using your locale-specific format. However, internally, Orchestrator uses a static format to ensure that the proper dates and times are used for activity property values.

The standard date/time format used by Orchestrator is a 24-hour time format displayed as yyyy-MM-ddTHH:mm:ss. This date/time format conforms with ISO 8601.

Example: You set your formatting to United Kingdom with a date format of DD/MM/YY. When you click the ellipsis button next to a date/time property in an activity, the date displayed is in DD/MM/YY format. When you click **OK** to save your changes, the date/time information displayed in the property is in the format YYYY-MM-DDTHH:MM:SS.

Workaround: If you have a runbook that contains activity properties configured with the older date/time formatting of MM/DD/YYYY HH:MM:SS AM, Orchestrator uses the old format. The runbook is properly interpreted by Orchestrator when the activity runs. If you change a property from the default date/time format, the date/time format changes to the new format.

Use UTC time when you filter on published date/time

Description: In your runbook workflow, use **Activity end time UTC** instead of **Activity end time** to filter on events. **Activity end time UTC** is the uniform time across your entire Orchestrator deployment. **Activity end time** represents the local time of the management server.

The standard date/time format used by Orchestrator is a 24-hour time format displayed as yyyy-MM-ddTHH:mm:ss. This date/time format conforms with ISO 8601.

Workaround: None.

Registration of an integration pack fails if you first attempt to register an Opalis 6.3 integration pack

Description: If you attempt to register an Integration Pack for Opalis 6.3, you get an unexpected failure because these integration packs are not supported in System Center 2012 - Orchestrator. If you then attempt to register a valid integration pack for System Center 2012 - Orchestrator, you receive the same error.

Workaround: You must close and restart Deployment Manager before registering a valid System Center 2012 - Orchestrator integration pack.

An integration pack must be registered and deployed before importing a runbook that uses it.

Description: If you import a runbook that uses an activity from an integration pack that is not registered and deployed, the activities from that integration pack will be marked with a question mark (?). Even after the integration pack is installed and registered, the activities will not work correctly.

Workaround: Import the runbook again after the integration pack is deployed and registered.

Different date/time formats between versions of integration packs

Description: The release candidate versions of integration packs for System Center 2012 use a different format for date/time published data values than the format used by the released version of integration packs for System Center 2012. Normally, you do not encounter this difference since it is only an issue if you subscribe to a date/time value property of a release candidate version of a System Center integration pack activity.

Workaround: If you encounter problems with date/time properties published with the System Center 2012 integration packs, use the <u>Format Date/Time</u> activity to translate between the two formats. The <u>Format Date/Time</u> activity has a **Details** pane with an **Input** section and **Output** section where you can specify a date format. You can enter an input and output format based on your translation requirements.

The formats are as follows:

- System Center 2012 Orchestrator integration packs for System Center 2012 Components
 RC: yyyy-MM-ddThh:mm:ss
- System Center 2012 integration packs for pre-System Center 2012 Products RC: M/d/yyyy
 h:m:s tt

System Center 2012 - Operations Manager integration pack

ManagementGroup and ManagementGroupId filters

Description: The ManagementGroup and ManagementGroupId filters in the **Get Alert** and **Monitor Alert** activities do not work.

Workaround: None.

See Also

Orchestrator

Release Notes for Orchestrator in System Center 2012 SP1

These release notes contain information that is required to successfully install Orchestrator in System Center 2012 Service Pack 1 (SP1). They contain information that is not available in the product documentation.

Before you install and use Orchestrator, read these release notes. These release notes apply to Orchestrator in System Center 2012 SP1.

If you are looking for the Release Notes for the original release of System Center 2012 - Orchestrator, see Release Notes for System Center 2012 - Orchestrator.

Known Issues

Setup program will fail when deploying IPs or executing runbooks on a computer running Windows Server 2012 without .NET 3.5 enabled.

Description: When trying to deploy an IP or execute a runbook on a computer running Windows Server 2012 on which .NET 3.5 is not enabled, the execution will fail.

Workaround: Enable .NET 3.5 manually on the computer running Windows Server 2012 and try again.

System Center 2012 - Service Manager IP: Custom Enum value that duplicates an existing Enum value causes activity to fail.

Description: If you create a custom Enum value that duplicates the name of an existing Enum value, the activity will fail.

Workaround: All customized Enum values must have names that are different from all other Enum values.

System Center 2012 - Service Manager IP: the Orchestrator server restarts after deploying the IP

Description: After Deploying the IP for Service Manager in System Center 2012 SP1 to the runbook server, the Orchestrator server is automatically restarted (without any further notice).

Workaround: None. The computer must be restarted so that the Service Manager IP can update some of the binary files that have been used.

When running Orchestrator on Windows Server 2012, the Run Program activity doesn't work in Interactive mode

Description: For example, on a runbook server that is running Windows Server 2012, start a runbook containing a Run Program activity that has been configured to run notepad.exe in Interactive mode. Notepad.exe is started as a background process instead of as a foreground process.

Workaround: In the registry key

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Windows, the value for the NoInteractiveServices subkey defaults to 1, which means that no service is allowed to run interactively, regardless of whether it has SERVICE_INTERACTIVE_PROCESS. When NoInteractiveServices is set to a 0, services with SERVICE_INTERACTIVE_PROCESS are allowed to run interactively. Change the value of the NoInteractiveServices subkey to 0, and then restart the computer.

See Also

Orchestrator