



Hewlett Packard
Enterprise

HPE Database and Middleware Automation

Ultimate Edition

Software Version: 10.40
Linux, Solaris, AIX, and HP-UX

Workflows for Microsoft SQL Server

Document Release Date: December 2015
Software Release Date: December 2015

Legal Notices

Warranty

The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HPE shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HPE required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 2015 Hewlett Packard Enterprise Development LP

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

<https://softwaresupport.hp.com>

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

<https://hpp12.passport.hp.com/hppcf/createuser.do>

Or click the **the Register** link at the top of the HPE Software Support page.

You will also receive updated or new editions if you subscribe to the appropriate product support service.

Contact your HPE sales representative for details.

Support

Visit the HPE Software Support Online web site at: **<https://softwaresupport.hp.com>**

This web site provides contact information and details about the products, services, and support that HPE Software offers.

HPE Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HPE support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

<https://hpp12.passport.hp.com/hppcf/createuser.do>

To find more information about access levels, go to:

<https://softwaresupport.hp.com/web/softwaresupport/access-levels>

HP Software Solutions Now accesses the HPESW Solution and Integration Portal Web site. This site enables you to explore HPE Product Solutions to meet your business needs, includes a full list of Integrations between HPE Products, as well as a listing of ITIL Processes. The URL for this Web site is **<http://h20230.www2.hp.com/sc/solutions/index.jsp>**

About this PDF Version of Online Help

This document is a PDF version of the online help. This PDF file is provided so you can easily print multiple topics from the help information or read the online help in PDF format. Because this content was originally created to be viewed as online help in a web browser, some topics may not be formatted properly. Some interactive topics may not be present in this PDF version. Those topics can be successfully printed from within the online help.

Contents

SQL Server	7
MS SQL - Compliance Audit	8
Prerequisites for this Workflow	9
How this Workflow Works	10
How to Run this Workflow	14
Sample Scenarios	17
Parameters for MS SQL - Compliance Audit	22
MS SQL - Install Patch	23
Prerequisites	23
Process Overview	23
Workflow: MS SQL - Install Patch	24
Solution pack	24
Parameters to expose	24
Input parameters	24
FAQs	26
How do I install the SQL Server patch on all instances on the server?	26
How do I install the SQL Server patch on multiple cluster nodes?	26
MS SQL - Install Cluster Patch	26
Prerequisites	26
Process Overview	27
Workflow: MS SQL - Install Cluster Patch	27
Solution pack	27
Parameters to expose	27
Input parameters	28
FAQs	29
How do I install the SQL Server patch on all instances on the server?	29
How do I install the SQL Server patch on multiple cluster nodes?	29
Provisioning Cluster	29
Prerequisites	29
Process Overview	30
Workflow 1: MS SQL - Install Clustered SQL Instance	30
Solution pack	30
Parameters to expose	31
Input parameters	31
Workflow 2: MS SQL - Add Node to Cluster	33
Solution pack	34
Parameters to expose	34
Input parameters	34
Workflow 3: MS SQL Create Database	35
Solution pack	35
Parameters to expose	36

Input parameters	36
Provisioning Standalone Instance	36
Prerequisites	37
Process Overview	37
Workflow 1: MS SQL - Install Standalone SQL Instance	38
Solution pack	38
Parameters to expose	38
Input parameters	38
Workflow 2: MS SQL Create Database	40
Solution pack	41
Parameters to expose	41
Input parameters	41
Refreshing Database	41
MS SQL - Backup Database	43
Prerequisites for this Workflow	44
How this Workflow Works	45
How to Run this Workflow	46
Sample Scenarios	47
Parameters for Backup MS SQL Database	49
MS SQL - Restore Database	52
Prerequisites for this Workflow	53
How this Workflow Works	54
How to Run this Workflow	57
Sample Scenarios	59
Parameters for Restore MS SQL Database	61
MS SQL - Backup and Restore Database	63
Prerequisites for this Workflow	65
How this Workflow Works	66
How to Run this Workflow	69
Sample Scenarios	71
Parameters for Backup and Restore MS SQL Database	75
DB Release for SQL Server	79
Prerequisites for this Workflow	81
How this Workflow Works	82
How to Run this Workflow	85
Sample Scenarios	89
Parameters for DB Release for SQL Server	92
MS SQL Drop Database	95
Prerequisites for this Workflow	96
How this Workflow Works	97
How to Run this Workflow	98
Parameters for MS SQL - Drop Database	99
MS SQL - Upgrade Standalone SQL Instance	99
Prerequisites for this Workflow	100
How this Workflow Works	101
How to Run this Workflow	103

- Parameters for MS SQL - Upgrade Standalone SQL Instance 104
- MS SQL Rollback Patch 105
 - Prerequisites for this Workflow 106
 - How this Workflow Works 107
 - How to Run this Workflow 109
 - Parameters for MS SQL Rollback Patch 110

- Send Documentation Feedback 111

SQL Server

Workflow type	Workflow name
Compliance	"MS SQL - Compliance Audit" on the next page
Provisioning	"Workflow 1: MS SQL - Install Clustered SQL Instance" on page 30
	"Workflow 2: MS SQL - Add Node to Cluster" on page 33
	"Workflow 3: MS SQL Create Database" on page 35
	"Workflow 1: MS SQL - Install Standalone SQL Instance" on page 38
	"Workflow 2: MS SQL Create Database" on page 40
	"MS SQL - Upgrade Standalone SQL Instance" on page 99
Patching	"MS SQL - Install Patch" on page 23
	"MS SQL - Install Cluster Patch" on page 26
	"MS SQL Rollback Patch" on page 105
Refreshing	"MS SQL - Backup Database" on page 43
	"MS SQL - Backup and Restore Database" on page 63
	"MS SQL - Restore Database" on page 52
Release Management	"DB Release for SQL Server" on page 79

MS SQL - Compliance Audit

The MS SQL - Compliance Audit workflow enables you to audit a Microsoft SQL Server instance for compliance with the following security benchmark requirements:

- Center for Internet Security (CIS) security configuration benchmarks
- Payment Card Industry (PCI) data security standard
- Sarbanes-Oxley (SOX) requirements

The workflow performs CIS Level 1 and Level 2 auditing for a SQL Server instance. The audit identifies compliance related problems with a SQL Server instance.

The workflow performs the checks included in the CIS benchmark and then maps those CIS checks to the benchmark type that you specify in the Compliance Type parameter. The audit summary email will match the Compliance Type that you specify.

To use this workflow in your environment, see the following information:

Topic	Information Included
"Prerequisites for this Workflow"	List of prerequisites that must be satisfied before you can run this workflow
"How this Workflow Works"	Information about what the workflow does, including validation checks performed, steps executed, and step descriptions
"How to Run this Workflow"	Instructions for running this workflow in your environment
"Sample Scenarios"	Examples of typical parameter values for this workflow
Parameters	List of input parameters for this workflow

Prerequisites for this Workflow

Be sure that the following prerequisites are satisfied before you run the MS SQL - Compliance Audit workflow:

- The latest HPE DMA solution packs require the latest HPE DMA platform. To use the latest solution packs, update the HPE DMA platform. HPE DMA10.40 solution packs are supported on HPE DMA10.40 (and later).
- You have installed the Database Compliance solution pack.

The workflow must be able to:

- Execute `reg.exe` (Windows Server command-line registry tool), `wmic.exe` (Windows Management Instrumentation Command-line tool), and “net” Windows utilities on the target server. These utilities are included in the base Windows Server installations.
- Log in to the SQL Server instance using Windows-authenticated login credentials.
- Read system tables and execute system procedures upon connecting to the SQL Server instance.

Any other dependencies or other requirements?

For more information about prerequisites for Microsoft SQL Server, refer to the [Microsoft SQL Server Documentation](#).

How this Workflow Works

This workflow performs the following actions:

- Prepares to run the workflow by gathering information about the target SQLServerInstance and validating parameter values.
- Audits the various configuration settings specified in the pertinent CIS, SOX, or PCI benchmark.
- Composes and sends an email containing the results of the audit.

Note: The emails are sent through the mail server configured on the HPE DMA server. You can configure the mail server in the path below:

DMA setup > Configuration > Outgoing Mail > Server.

Validation Checks Performed show

This workflow validates the following conditions:

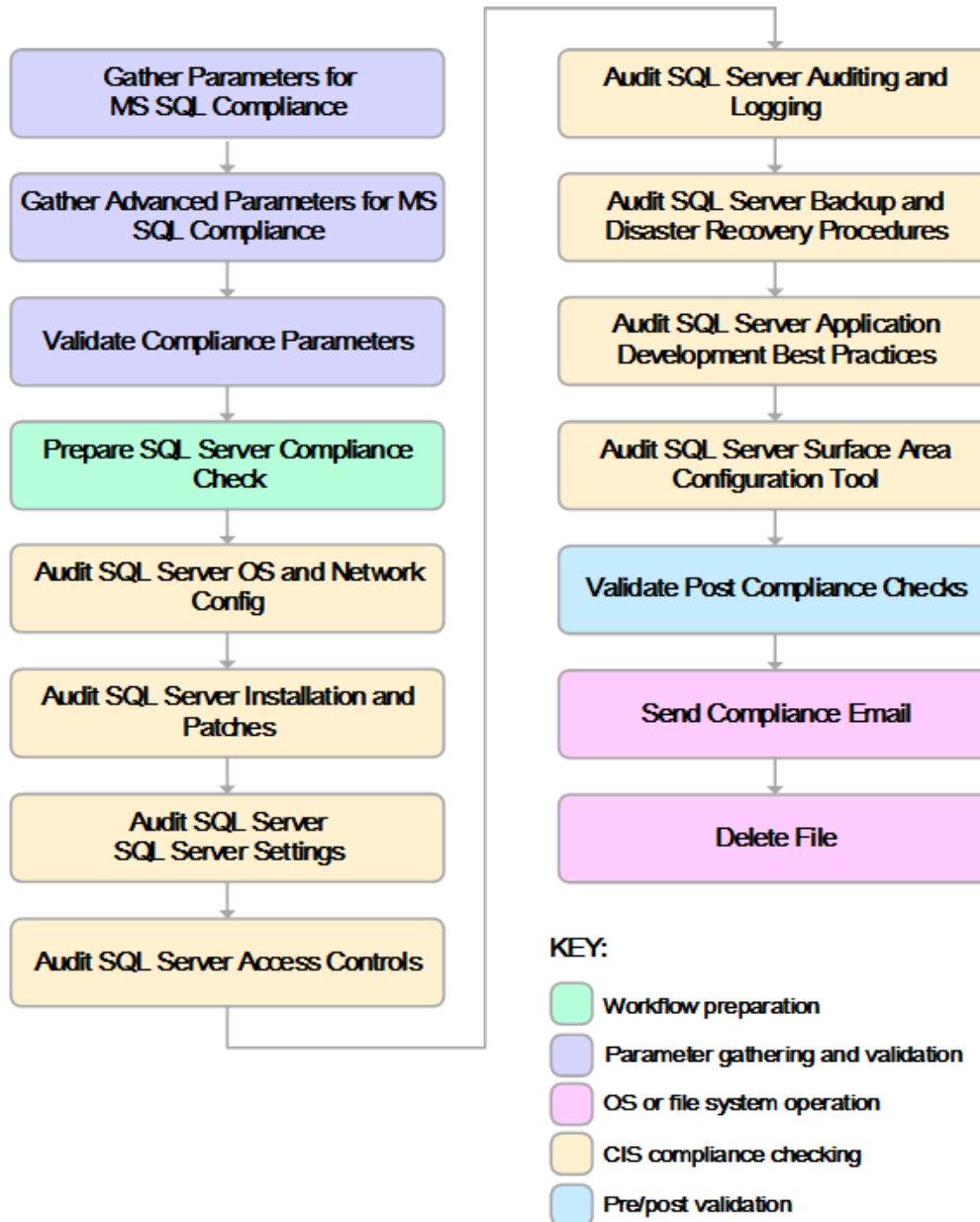
1. Either `sqlcmd.exe` or `osql.exe` must be installed on the target machine.
2. Any Excluded Checks specified by the user refer to actual CIS, SOX, or PCI benchmark checks.
3. Any email addresses specified are valid addresses.
4. The workflow can create the temporary file that will store the compliance check results.

Are these correct? Anything else?

Steps Executed show

The MS SQL - Compliance Audit workflow includes the following steps. Each step must complete successfully before the next step can start. If a step fails, the workflow reports a failure and all subsequent steps are skipped.

Click each box in the diagram to view additional information about that step.



Steps Used by Run MS SQL Compliance Audit

Workflow Step	Description
Gather Parameters for MS SQL	This step gathers two pieces of information: (1) the type of compliance audit to perform and (2) the list of compliance checks to exclude from the audit.

Steps Used by Run MS SQL Compliance Audit, continued

Workflow Step	Description
Compliance	
Gather Advanced Parameters for MS SQL Compliance	This step gathers the information that the workflow needs to create and deliver the compliance audit report via email. It also enables you to specify the name of the latest available SQL Server build and the Windows domain user.
Validate Compliance Parameters	<p>This step validates the input parameters specified in the previous steps. It validates the list of excluded checks to ensure that all specified checks in the list correspond to actual Center for Internet Security (CIS) benchmark items. It also validates the email information to ensure that all specified email addresses are valid.</p> <p>The step then creates the path to the temporary file that will store the results of the current audit as the workflow is running. This file is deleted after the audit report is sent.</p>
Prepare SQL Server Compliance Check	<p>This step determines whether workflow can perform the following actions on the target system:</p> <ul style="list-style-type: none"> • Check database connectivity • Query the registry • Check the registry for SQL Server • Execute Windows Management Instrumentation (WMI) API calls • Execute the <code>net user /?</code> command <p>If the workflow can perform all of these actions, it is capable of running the Center for Internet Security (CIS) Security Configuration Benchmark compliance tests.</p>
Audit SQL Server OS and Network Config	This step audits the scorable recommendations in Section 1, Operating System and Network Specific Configuration, of the Center for Internet Security (CIS) Security Configuration Benchmark for Microsoft SQL Server 2005, version 2.0.0 (December 2011).
Audit SQL Server Installation and Patches	This step audits the scorable recommendations in Section 2, SQL Server Installation and Patches, of the Center for Internet Security (CIS) Security Configuration Benchmark for Microsoft SQL Server 2005, version 2.0.0 (December 2011).
Audit SQL Server SQL Server Settings	This step audits the scorable recommendations in Section 3, SQL Server Settings, of the Center for Internet Security (CIS) Security Configuration Benchmark for Microsoft SQL Server 2005, version 2.0.0 (December 2011).
Audit SQL Server Access Controls	This step audits the scorable recommendations in Section 4, Access Controls, of the Center for Internet Security (CIS) Security Configuration Benchmarks for Microsoft SQL Server 2005, version 2.0.0 (December 2011).
Audit SQL Server Auditing and Logging	This step audits the scorable recommendations in Section 5, Auditing and Logging, of the Center for Internet Security (CIS) Security Configuration Benchmark for Microsoft SQL Server 2005, version 2.0.0 (December 2011).

Steps Used by Run MS SQL Compliance Audit, continued

Workflow Step	Description
Audit SQL Server Backup and Disaster Recovery Procedures	This step audits the scorable recommendations in Section 6, Backup and Disaster Recovery Procedures, of the Center for Internet Security (CIS) Security Configuration Benchmark for Microsoft SQL Server 2005, version 2.0.0 (December 2011).
Audit SQL Server Application Development Best Practices	This step audits the scorable recommendations in Section 8, Application Development Best Practices, of the Center for Internet Security (CIS) Security Configuration Benchmarks for Microsoft SQL Server 2005, version 2.0.0 (December 2011).
Audit SQL Server Surface Area Configuration Tool	This step audits the scorable recommendations in Section 9, Surface Area Configuration Tool, of the Center for Internet Security (CIS) Security Configuration Benchmarks for Microsoft SQL Server 2005, version 2.0.0 (December 2011).
Validate Post-Compliance Checks	<p>This step reads the temporary file that contains the results of the compliance audit and prints the audit results to the HPE DMA Console. It also creates (or updates) the compliance metadata fields for the target.</p> <p>If email addresses were specified, it also creates a report in HTML format that will be emailed to those addresses by a later step in the workflow.</p>
Send Compliance Email	If email addresses are provided, this step sends the previously generated compliance audit report to the specified email addresses.
Delete File	This step deletes the specified file on the target server.

Note: For input parameter descriptions and defaults, see "[Parameters for MS SQL - Compliance Audit](#)" on page 22.

How to Run this Workflow

The following instructions show you how to customize and run the MS SQL - Compliance Audit workflow in your environment.

The workflow provides default values for some parameters. These default values are usually sufficient for a "typical" installation. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios. Any parameters not explicitly specified in the deployment will have the default values listed in "[Parameters for MS SQL - Compliance Audit](#)" on page 22.

Note: Before following this procedure, review the "[Prerequisites for this Workflow](#)" on page 9, and ensure that all requirements are satisfied.

To use the Run MS SQL Compliance Audit workflow:

1. Create a deployable copy of the workflow.
2. Determine the values that you will specify for the following parameters:

Parameters Defined in this Step: Gather Parameters for SQL Server Compliance

Parameter Name	Default Value	Required	Description
Compliance Type	CIS	optional	Type of compliance report that will be generated by the workflow. Supported types are: CIS = Center for Internet Security (CIS) Security Configuration Benchmark PCI = Payment Card Industry (PCI) Data Security Standard SOX = Sarbanes-Oxley (SOX) sections 302.2, 302.4b, 302.4c, and 302.5 requirements
Excluded Checks	no default	optional	Comma-separated list of compliance checks to exclude from the audit. For example: 1.2, 2, 3.*, 5*, 6.1.2 <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <p>Note: Make sure that the checks specified here correspond with the compliance audit type (CIS, PCI, or SOX) that you are running.</p> </div>

Parameters Defined in this Step: Advanced Parameter for MS SQL Compliance

Parameter Name	Default Value	Required	Description
Email Addresses to Receive Report	no	optional	Comma-separated list

Parameters Defined in this Step: Advanced Parameter for MS SQL Compliance, continued

Parameter Name	Default Value	Required	Description
	default		of email addresses for those individuals or groups who will receive a copy of the compliance audit report.
Latest Build to Check for	no default	optional	The latest build of Microsoft SQL Server 2005, according to Microsoft. Ensure that instance is at least patched up to indicated build level. Example value would be "5058" for SQL 2012's SP2. If no value is given, the related Compliance check will be skipped.

Note: This is the minimum set of parameters required to run this workflow. You may need to expose additional parameters depending on your objectives.

See "[Parameters for MS SQL - Compliance Audit](#)" on page 22 for detailed descriptions of all input parameters for this workflow, including default values.

3. In the workflow editor, expose any additional parameters that you need. You will specify values for those parameters when you create the deployment or at runtime.
4. Save the changes to the workflow (click **Save** in the lower right corner).
5. Create a new deployment.
6. On the Parameters tab, specify values (or set the type to Runtime Value) for the required parameters listed in step 2 and any additional parameters that you have exposed. You do not need to specify values for those parameters whose default values are appropriate for your environment.
7. On the Targets tab, specify one or more targets for this deployment.
8. Save the deployment (click **Save** in the lower right corner).
9. Run the workflow using this deployment, specifying any runtime parameters.

To verify the results:

The workflow will complete and report SUCCESS on the Console if it has run successfully. If an error occurs during workflow execution, the error is logged, and the workflow terminates in the FAILURE state.

Information about each compliance check is displayed in the step output on the Console (and the History page) for each of the audit steps.

A summary of the compliance audit is also displayed in the step output for the Validate Post Compliance Checks step.

To view the reports:

A compliance audit summary in HTML format is emailed to all parties on the Email Addresses to Receive Report list.

After you run this workflow, you can generate two types of compliance reports on the Reports page:

- Database Compliance Report
- Database Compliance Detail Report

To access the Database Compliance reports:

1. Go to the Reports page.
2. At the bottom of the page, specify the following settings:
For the Database Compliance Report:
 - a. Select the Database Compliance report.
 - b. Select the organization where your target resides.
 - c. Because this report lists the latest compliance audit reports for all targets in the specified organization, you do not specify a Server, Database, or Time span.For the Database Compliance Detail Report:
 - a. Select the Database Compliance Details report.
 - b. Select the organization where your target resides.
 - c. Specify the Server and Instance that you selected when you created your deployment.
3. Click **Run report**.

Sample Scenarios

This topic shows you how to use various parameters to achieve the following compliance audit scenarios in your environment using the "[MS SQL - Compliance Audit](#)" workflow.

Scenario 1: Perform a Partial CIS Compliance Audit and Email the Results show

In the scenario, the following checks are excluded from the audit:

- Section 7: Replication
- Section 9: Surface Area Configuration Tool

A summary report is sent to the three parties listed in the Email Addresses to Receive Report parameter.

Parameter Name	Example Value	Description
Compliance Type	CIS	Type of compliance report that will be generated by the workflow. Supported types are: CIS = Center for Internet Security (CIS) Security Configuration Benchmark PCI = Payment Card Industry (PCI) Data Security Standard SOX = Sarbanes-Oxley (SOX) sections 302.2, 302.4b, 302.4c, and 302.5 requirements
Excluded Compliance Checks	7.*,9.*	Comma-separated list of compliance checks to exclude from the audit. For example: 1.2, 2, 3.*, 5*, 6.1.2 Note: Make sure that the checks specified here correspond with the compliance audit type (CIS, PCI, or SOX) that you are running.
Email Addresses to Receive Report	SQLDBAdminTeam@mycompany.com, SQLDBAdminMgr@mycompany.com, CustomerSupportTeam@mycompany.com	Comma-separated list of email addresses for those individuals or groups who will receive a copy of the compliance audit report.

Note: Some of these parameters are not exposed by default in the deployment.

Be sure that the default values for all remaining input parameters are appropriate for your environment (see ["Parameters for MS SQL - Compliance Audit"](#)).

Scenario 2: Perform a Full PCI Compliance Audit and Email the Results show

A summary report is sent to the three parties listed in the Email Addresses to Receive Report parameter.

Parameter Name	Example Value	Description
Compliance Type	PCI	Type of compliance report that will be generated by the workflow. Supported types are: CIS = Center for Internet Security (CIS) Security Configuration Benchmark PCI = Payment Card Industry (PCI) Data Security Standard SOX = Sarbanes-Oxley (SOX) sections 302.2, 302.4b, 302.4c, and 302.5 requirements
Email Addresses to Receive Report	SQLDBAdminTeam@mycompany.com, SQLDBAdminMgr@mycompany.com, CustomerSupportTeam@mycompany.com	Comma-separated list of email addresses for those individuals or groups who will receive a copy of the compliance audit report.

Note: Some of these parameters are not exposed by default in the deployment.

Be sure that the default values for all remaining input parameters are appropriate for your environment (see ["Parameters for MS SQL - Compliance Audit"](#)).

Scenario 3: Perform a Full SOX Compliance Audit, Email the Results, and Configure Windows Domain User Using Runtime Parameters show

A summary report is sent to the three parties listed in the Email Addresses to Receive Report parameter.

Note: You may want to run this workflow against a MS SQL instance that can only be accessed by a Windows user with a temporary password. By using a runtime parameter for the password, you can ensure that the password used is always the latest.

To specify the Windows domain user at the time you execute a deployment with runtime parameters, perform the following additional steps:

1. When you make a copy of the workflow, expand the appropriate step, and then set the Windows domain user parameters—Instance Account and Instance Password—to **- User selected -**.
2. When you create a deployment from the copy of the workflow, set the parameter types to **Runtime Value**.
3. When you execute the deployment, specify the Windows domain user account and password.

Parameter Name	Example Value	Description
Compliance Type	SOX	Type of compliance report that will be generated by the workflow. Supported types are: CIS = Center for Internet Security (CIS) Security Configuration Benchmark PCI = Payment Card Industry (PCI) Data Security Standard SOX = Sarbanes-Oxley (SOX) sections 302.2, 302.4b, 302.4c, and 302.5 requirements
Email Addresses to Receive Report	SQLDBAdminTeam@mycompany.com, SQLDBAdminMgr@mycompany.com, CustomerSupportTeam@mycompany.com	Comma-separated list of email addresses for those individuals or groups who will receive a copy of the compliance audit report.
Instance Account	Domain\DomainUserAcct Note: Enter at runtime.	The Windows account that will perform the compliance audit.
Instance Password	DomainUserPswd Note: Enter at runtime.	The password for the Windows account that will perform the compliance audit.
Latest Build to Check for	5058	The latest build of Microsoft SQL Server 2005, according to Microsoft. Ensure that instance

Parameter Name	Example Value	Description
		is at least patched up to indicated build level. Example value would be "5058" for SQL 2012's SP2. If no value is given, the related Compliance check will be skipped.

Note: Some of these parameters are not exposed by default in the deployment.

Be sure that the default values for all remaining input parameters are appropriate for your environment (see ["Parameters for MS SQL - Compliance Audit"](#)).

Scenario 4: Perform a Full CIS Compliance Audit and Display the Results on the HPE DMA Consoleshow

In the scenario, all scorable checks are performed, and the compliance audit report is displayed only on the HPE DMA Console. In this case, a summary report is not emailed. This scenario would be appropriate for initial testing.

It is not necessary to specify any input parameters in this scenario unless the SQL Server inventory file is located in a non-standard directory.

Parameter Name	Example Value	Description
Compliance Type	CIS	Type of compliance report that will be generated by the workflow. Supported types are: CIS = Center for Internet Security (CIS) Security Configuration Benchmark PCI = Payment Card Industry (PCI) Data Security Standard SOX = Sarbanes-Oxley (SOX) sections 302.2, 302.4b, 302.4c, and 302.5 requirements

Be sure that the default values for all remaining input parameters are appropriate for your environment (see ["Parameters for MS SQL - Compliance Audit"](#)).

Parameters for MS SQL - Compliance Audit

The following tables describe the required and optional input parameters for this workflow. Some of these parameters are not initially visible in a deployment. For most parameters, if you do not specify a value for a parameter, a default value is assigned.

Only those parameters that are configurable in a standard deployment are listed here. Input parameters that must be mapped to output parameters of previous steps are not listed.

Parameters Defined in this Step: Gather Parameters for MS SQL Compliance

Parameter Name	Default Value	Required	Description
Compliance Type	CIS	optional	Type of compliance report that will be generated by the workflow. Supported types are: CIS = Center for Internet Security (CIS) Security Configuration Benchmark PCI = Payment Card Industry (PCI) Data Security Standard SOX = Sarbanes-Oxley (SOX) sections 302.2, 302.4b, 302.4c, and 302.5 requirements
Excluded Checks	no default	optional	Comma-separated list of compliance checks to exclude from the audit. For example: 1.2, 2, 3.*, 5*, 6.1.2 Note: Make sure that the checks specified here correspond with the compliance audit type (CIS, PCI, or SOX) that you are running.

Parameters Defined in this Step: Gather Advanced Parameters for MS SQL Compliance

Parameter Name	Default Value	Required	Description
Email Addresses to Receive Report	no default	optional	Comma-separated list of email addresses for those individuals or groups who will receive a copy of the compliance audit report.
Instance Account	no default	optional	The Windows account that will perform the compliance audit.
Instance Password	no default	optional	The password for the Windows account that will perform the compliance audit.
Latest Build to Check for	no default	optional	The latest build of Microsoft SQL Server 2005, according to Microsoft. Ensure that instance is at least patched up to indicated build level. Example value would be "5058" for SQL 2012's SP2. If no value is given, the related Compliance check will be

Parameters Defined in this Step: Gather Advanced Parameters for MS SQL Compliance, continued

Parameter Name	Default Value	Required	Description
			skipped.

MS SQL - Install Patch

This section describes how to use Database and Middleware Automation (HPE DMA) to create a repeatable, standardized method to quickly and accurately install Microsoft Microsoft SQL Server patches on SQL Server installations across an enterprise to reach patch currency standards.

Tip: To patch more complex SQL Server clustered environments, see *Achieve Patch Currency for Microsoft SQL Server Clustered Environments Using HPE DMA*, available at: <https://softwaresupport.hp.com/>

Prerequisites

Before performing the procedures in this section, your environment must meet the following minimum requirements:

- A server running Windows 2008, 2008 R2, or 2012.
- A SQL Server instance—version 2005, 2008, 2008R2, or 2012—is provisioned and ready to be patched.
- Patch media:
 - The SQL Server patch file from Microsoft.
 - Patch installation media must be available locally or available for download from the software repository.
- Storage: A staging directory with 1 gigabyte available.
- Licenses for SQL Server and HPE DMA.

Process Overview

Installing a SQL Server patch to a Microsoft SQL Server installation with HPE DMA is a simple, one-step process. All required checks and steps have been implemented in a single HPE DMA workflow.

Use the following HPE DMA workflow to standardize the process of installing a SQL Server patch:

HPE DMA can install any of the following types of SQL Server patches:

- Hot Fixes
- Cumulative Updates
- Service Packs

Note: This workflow patches a single SQL Server instance unless you use the use the advanced parameter Patch All Instances on Server. The advanced parameter is demonstrated in this section.

Tip: To patch multiple SQL Server cluster nodes, run MS SQL - Install Patch once for each node, or for an easier process, use the MS SQL - Install Cluster Patch workflow that is described in *Achieve Patch Currency for Microsoft SQL Server Clustered Environments Using HPE DMA*, available at: <https://softwaresupport.hp.com/>

Workflow: MS SQL - Install Patch



This section provides detailed information required to run the MS SQL - Install Patch workflow.

Tip: To patch multiple SQL Server cluster nodes, run MS SQL - Install Patch once for each.

Solution pack

This workflow requires the Database Patching Solution Pack.

Parameters to expose

If you want to patch all SQL Server instances, in the workflow's MS SQL - Advanced Parameters - Install Patch step, expose the Patch All Instances on Server parameter.¹

Input parameters

When you deploy the MS SQL - Install Patch workflow, specify input parameter values for the following steps.

Bold text in the following tables indicates that you must specify a value for the parameter.

¹This parameter is hidden by default and must be exposed when you make a copy of the workflow.

Step: MS SQL - Parameters - Install Patch

Parameter	Description	Example Value
Download From Software Directory	<p>Required: The name of the SQL Server patch file obtained from Microsoft.¹</p> <p>Note: This must be an EXE file. If you obtain a ZIP file from Microsoft, unzip it to retrieve the EXE file.</p>	SQL12_SP1.exe
Download Target Destination	<p>Required: The local directory where the SQL Server patch file is stored:</p> <p>If patch file is in the software repository: Location where Download From Software Directory will be downloaded</p> <p>If patch file is on the target: Location where the Microsoft SQL Server patch file already exists</p> <p>Upon a successful workflow completion, all downloaded files are cleaned up.</p>	C:\temp
Web Service Password	Required: Password for the HPE DMA Discovery web service API.	●●●
Web Service User	Required: User who is capable of modifying the managed environment by using the HPE DMA Discovery web service API.	dmawebuser

Step: MS SQL - Advanced Parameters - Install Patch

Parameter	Description	Example Value
Patch All Instances on Server	Optional: Flag to determine whether all SQL Server instances on the server will be patched. Valid values: Yes or No. Default: No.	Yes

Step: MS SQL Kill Processes

Parameter	Description	Example Value
Instance Account	Optional: The Windows account	

¹If the file is not found on the target server(s), it will be downloaded from the software repository.

Step: MS SQL Kill Processes, continued

Parameter	Description	Example Value
	that will terminate the SQL Server processes.	

FAQs

How do I install the SQL Server patch on all instances on the server?

To install the SQL Server patch on all instances on the server, set the Patch All Instances on Server parameter to Yes before you execute the deployment:

Workflow: MS SQL - Install Patch

Step: MS SQL - Advanced Parameters - Install Patch

Parameter: Patch All Instances on Server¹

How do I install the SQL Server patch on multiple cluster nodes?

To install the SQL Server patch on multiple cluster nodes, run the MS SQL - Install Patch workflow once on each cluster node.

MS SQL - Install Cluster Patch

This section describes how to use Database and Middleware Automation (HPE DMA) to create a repeatable, standardized method to quickly and accurately install Microsoft SQL Server patches on SQL Server clustered installations across an enterprise to reach patch currency standards.

Tip: To patch SQL Server standalone environments, see *Achieve Patch Currency for Microsoft SQL Server Environments Using HPE DMA*, available at: softwaresupport.hp.com

Prerequisites

Before performing the procedures in this section, your environment must meet the following minimum requirements:

- A server running Windows 2008, 2008 R2, or 2012.
- A SQL Server clustered instance—version 2008, 2008 R2, or 2012—is provisioned and ready to be patched.

¹This parameter is hidden by default and must be exposed when you make a copy of the workflow.

- Patch media:
The SQL Server patch file from Microsoft.
Patch installation media must be available locally or available for download from the software repository.
- Storage: A staging directory with 1 gigabyte available.
- Licenses for SQL Server and HPE DMA.

Process Overview

Installing a SQL Server patch to a Microsoft SQL Server clustered installation with HPE DMA is a simple, one-step process. All required checks and steps have been implemented in a single HPE DMA workflow.

HPE DMA can install any of the following types of SQL Server patches:

- Hot Fixes
- Cumulative Updates
- Service Packs

Note: To execute the workflow, only one of the nodes in the SQL Server cluster needs to be a target for the deployment. The workflow discovers all cluster members and patches each one.

The following section provides detailed information required to run the workflow.

Workflow: MS SQL - Install Cluster Patch



This section provides detailed information required to run the MS SQL - Install Cluster Patch workflow.

Solution pack

This workflow requires the Database Patching Solution Pack.

Parameters to expose

None

Input parameters

When you deploy the MS SQL - Install Cluster Patch workflow, specify input parameter values for the following steps.

Step: MS SQL - Parameters - Install Patch

Parameter	Description	Example Value
Download From Software Directory	<p>Required: The name of the SQL server patch file obtained from Microsoft.¹</p> <p>Note: This must be an EXE file. If you obtain a ZIP file from Microsoft, unzip it to retrieve the EXE file.</p>	SQL12_SP1.exe
Download Target Destination	<p>Required: The local directory where the SQL server patch file is stored:</p> <p>If patch file is in the software repository: Location where Download From Software Directory will be downloaded</p> <p>If patch file is on the target: Location where the Microsoft SQL server patch file already exists</p> <p>Upon a successful workflow completion, all downloaded files are cleaned up.</p>	C:\temp
Web Service Password	Required: Password for the HPE DMA Discovery web service API.	●●●

Note: The step Run Subflow - MS SQL - Install Patch runs first to patch all passive nodes.

Step: Run Subflow - MS SQL - Install Patch

Parameter	Description	Example Value
Server Parallel Execution	Optional: Flag to determine whether the workflow is to execute in parallel. Set to False if you would like the workflow to execute serially. Default is True.	True

Note: The step Run Subflow - MS SQL - Install Patch runs again to patch the active node.

¹If the file is not found on the target server(s), it will be downloaded from the software repository.

Step: Run Subflow - MS SQL - Install Patch

Parameter	Description	Example Value
Server Parallel Execution	Optional: Flag to determine whether the workflow is to execute in parallel. Set to False if you would like the workflow to execute serially. Default is True.	True

FAQs

How do I install the SQL Server patch on all instances on the server?

To install the SQL Server patch on all instances on the server, set the Patch All Instances on Server parameter to Yes before you execute the deployment:

Workflow: MS SQL - Install Patch

Step: MS SQL - Advanced Parameters - Install Patch

Parameter: Patch All Instances on Server¹

How do I install the SQL Server patch on multiple cluster nodes?

To install the SQL Server patch on multiple cluster nodes, run the MS SQL - Install Patch workflow once on each cluster node.

Provisioning Cluster

This section describes how to use Database and Middleware Automation (HPE DMA) to create a repeatable, standardized “gold image” for provisioning Microsoft SQL Server, including the SQL Server clustered instance on multiple nodes and a SQL Server database on that clustered instance.

Tip: To provision a SQL Server standalone environment, see *Standardize Microsoft SQL Server Standalone Provisioning Using HPE DMA*, available at: softwaresupport.hp.com

Prerequisites

Before performing the procedures in this section, your environment must meet the following minimum requirements:

¹This parameter is hidden by default and must be exposed when you make a copy of the workflow.

- An existing Windows 2008, 2008 R2, or 2012 cluster
- Installation software:
The SQL Server 2008, 2008 R2, or 2012 software installation files, obtained from Microsoft.
The installation media must be available locally or available for download from the software repository.
- Storage:
An available shared disk for SQL Server shared files
A staging directory with 4 gigabytes available to unzip the SQL Server software
- Permissions to create an SQL Server database:
System Stored Procedures (SP)
CREATE LOGIN
If using a non-default database owner, the `sp_changedbowner` process is available
If a non-default database owner is specified and does not exist, permission to create the appropriate login
- .NET 3.5 is installed.

Note: For additional information, see "Run as a Windows Domain User" in the *HPE DMA Installation Guide*, available at: <https://softwaresupport.hp.com/>

- Licenses for SQL Server and HPE DMA.

For additional requirements, see the following Microsoft documentation:

SQL Server version	Microsoft documentation
2008	Hardware and Software Requirements for Installing SQL Server 2008
2008 R2	Hardware and Software Requirements for Installing SQL Server 2008 R2
2012	Hardware and Software Requirements for Installing SQL Server 2012

Process Overview

Patching SQL Server Cluster Infrastructure with HPE DMA is a simple, one-step process. All required checks and steps to patch an SQL Server Cluster infrastructure—including the Cluster Home.

Workflow 1: MS SQL - Install Clustered SQL Instance

This section provides detailed information required to run the MS SQL - Install Clustered SQL Instance workflow.

Tip: To provision multiple SQL Server clustered instances on the same Windows OS cluster, run MS SQL - Install Clustered SQL Instance once for each.

Solution pack

This workflow requires the Database Provisioning Solution Pack.

Parameters to expose

None

Input parameters

When you deploy the MS SQL - Install Clustered SQL Instance workflow, specify input parameter values for the following steps.

Bold text in the following tables indicates that you must specify a value for the parameter.

Step: MS SQL - Gather Parameters For Install Clustered SQL Instance

Parameter	Description	Example Value
Cluster Administrator Account	Required: The Windows domain user that will run the setup operation. This user requires elevated administrator privileges on the cluster. Format: <DOMAIN>\<USERNAME>	Win12\Administrator
Cluster Administrator Password	Required: Password for the Windows domain user that will run the setup operation. This user requires elevated administrator privileges on the cluster. Must be a strong Windows password.	●●●
Download From Software Directory	Optional: The name of the ZIP file that contains the SQL Server installation software files obtained from Microsoft. ¹ Note: If necessary, manually zip the installation software files up.	SQL12.zip
Download Target Destination	Required: The local directory where the SQL Server setup files are stored: ² If source files are in the software repository: Location where Download From Software Directory will be downloaded and extracted	C:\temp

¹If the file is not found on the target server(s), it will be downloaded from the software repository. For additional information, see [Alternative methods for specifying input files](#).

²For additional information, see [Alternative methods for specifying input files](#).

Step: MS SQL - Gather Parameters For Install Clustered SQL Instance, continued

Parameter	Description	Example Value
	<p>If source files are on the target: Location where the Microsoft SQL Server installation files already exist—not zipped up</p> <p>Upon a successful workflow completion, all downloaded and extracted files are cleaned up.</p>	
Instance Name	<p>Required: The name of the newly created virtual server and instance.</p> <p>Format: <Virtual Server>\<Instance Name></p> <p>Use MSSQLSERVER for the default instance and any other alphanumeric value for a named instance.</p>	SQL-CLUSTER\InstanceA
Public IP Address	<p>Required: Public IP Address.</p> <p>For SQL Server 2012 set to DHCP.</p>	DHCP
Public IP Network Name	<p>Required: IP Network Name for the clusters.</p> <p>Format: <Network Name>:<Subnet Mask></p> <p>For example: Public:255.255.255.0</p>	
SQL Agent Account	<p>Required: The login account for the SQL Server Agent service. Can be a local Windows user, a domain user, or a built-in account (for example: NT AUTHORITY\NETWORK SERVICE).</p> <p>If not a built-in account, also specify SQL Agent Password.</p> <p>This parameter is optional for SQL Server 2008 or 2008 R2.</p>	Win12\Administrator
SQL Agent Password	<p>Required: Specify if SQL Agent Account is specified.</p> <p>This parameter is optional for SQL Server 2008 or 2008 R2.</p>	●●●
SQL Service Account	<p>Required: The login account for the SQL service. Can be a local</p>	Win12\Administrator

Step: MS SQL - Gather Parameters For Install Clustered SQL Instance, continued

Parameter	Description	Example Value
	Windows user, a domain user, or a built-in account (for example: NT AUTHORITY\NETWORK SERVICE). If not a built-in account, also specify SQL Service Password. This parameter is optional for SQL Server 2008 or 2008 R2.	
SQL Service Password	Required: Specify if SQL Service Account is specified. This parameter is optional for SQL Server 2008 or 2008 R2.	●●●
SQL Sysadmin Accounts	Required: A comma-delimited list of user accounts that will be set as system administrators. Each account must either be a local Windows user or a domain user. This parameter is optional for SQL Server 2008 or 2008 R2.	Win12\Administrator

Step: MS SQL - Gather Advanced Parameters For Install Clustered SQL Instance

Parameter	Description	Example Value
-----------	-------------	---------------

Step: Discover SQL Databases

Parameter	Description	Example Value
Web Service Password	Required: Password for the HPE DMA Discovery web service API.	●●●
Web Service User	Required: User who is capable of modifying the managed environment by using the HPE DMA Discovery web service API.	dmawebuser

Workflow 2: MS SQL - Add Node to Cluster

This section provides detailed information required to run the MS SQL - Add Node to Cluster workflow.

Tip: To provision additional SQL Server nodes, run MS SQL - Add Node to Cluster once for each additional node.

Solution pack

This workflow requires the Database Provisioning Solution Pack.

Parameters to expose

None

Input parameters

When you deploy the MS SQL - Add Node to Cluster workflow, specify input parameter values for the following steps.

Bold text in the following tables indicates that you must specify a value for the parameter.

HPE DMA uses the latest (highest version number) of steps.

Step: MS SQL - Parameters - Add Node to Cluster

Parameter	Description	Example Value
Cluster Administrator Account	Required: The Windows domain user that will run the setup operation. This user requires elevated administrator privileges on the cluster. Format: <DOMAIN>\<USERNAME>	Win12\Administrator <i>Use the same value specified for Cluster Administrator Account in the MS SQL - Install Clustered SQL Instance deployment.</i>
Cluster Administrator Password	Required: Password for the Windows domain user that will run the setup operation. This user requires elevated administrator privileges on the cluster. Must be a strong Windows password.	●●● <i>Use the same value specified for Cluster Administrator Password in the MS SQL - Install Clustered SQL Instance deployment.</i>
Download From Software Directory	Optional: The name of the ZIP file that contains the (Undefined variable: DMAVariables_Global.DMA_SuppProduct) SQL Server installation software files obtained from Microsoft. ¹ Note: If necessary, manually zip the installation software files up.	SQL12.zip <i>Use the same value specified Download From Software Directory in the MS SQL - Install Clustered SQL Instance deployment.</i>
Download Target Destination	Required: The local directory where	C:\temp

¹If the file is not found on the target server(s), it will be downloaded from the software repository.

Step: MS SQL - Parameters - Add Node to Cluster, continued

Parameter	Description	Example Value
	<p>the SQL Server setup files are stored:</p> <p>If source files are in the software repository: Location where Download From Software Directory will be downloaded and extracted</p> <p>If source files are on the target: Location where the Microsoft SQL Server installation files already exist—not zipped up</p> <p>Upon a successful workflow completion, all downloaded and extracted files are cleaned up.</p>	
Instance Name	Required: The name of the newly created instance. Use MSSQLSERVER for the default instance and any other alphanumeric value for a named instance.	InstanceA <i>Use the <Instance Name> portion of the Instance Name parameter in the MS SQL - Install Clustered SQL Instance deployment.</i>

Step: MS SQL - Advanced Parameters - Add Node to Cluster

Parameter	Description	Example Value
-----------	-------------	---------------

Step: Discover SQL Databases

Parameter	Description	Example Value
Web Service Password	Required: Password for the HPE DMA Discovery web service API.	●●●
Web Service User	Required: User who is capable of modifying the managed environment by using the HPE DMA Discovery web service API.	dmawebuser

Workflow 3: MS SQL Create Database

This section provides detailed information required to run the MS SQL Create Database workflow.

Tip: You only need to run this workflow once per clustered instance. You can run it on any node.

Solution pack

This workflow requires the Database Provisioning Solution Pack.

Parameters to expose

None

Input parameters

When you deploy the MS SQL Create Database workflow, specify input parameter values for the following step.

Bold text in the following tables indicates that you must specify a value for the parameter.

HPE DMA uses the latest (highest version number) of steps.

Step: MS SQL Parameters Create Database

Parameter	Description	Example Value
Database Name	Required: Name of the new database.	NewDatabase
Web Service Password	Required: Password for the HPE DMA Discovery web service API.	●●●
Web Service User	Required: User who is capable of modifying the managed environment by using the HPE DMA Discovery web service API.	dmawebuser

Step: MS SQL Advanced Parameters Create Database

Parameter	Description	Example Value
SQL Instance Account	Optional: Either a Windows or SQL Server user that can log in. Used if the default Windows user is unable to log in to the instance.	?
SQL Instance Password	Optional: The password to the SQL Instance Account. Used if the default Windows user is unable to log in to the instance.	?

Provisioning Standalone Instance

This section describes how to use Database and Middleware Automation (HPE DMA) to create a repeatable, standardized “gold image” for provisioning Microsoft SQL Server, including the SQL Server standalone instance and database.

Tip: To provision more complex SQL Server clustered environments, see *Standardize Microsoft SQL Server Cluster Provisioning Using HPE DMA*, available at: softwaresupport.hp.com

Prerequisites

Before performing the procedures in this section, your environment must meet the following minimum requirements:

- A server running Windows 2008, 2008 R2, or 2012.
- Installation software:
 - The SQL Server 2008, 2008 R2, or 2012 software installation files, obtained from Microsoft.
 - The installation media must be available locally or available for download from the software repository.
- Storage: A staging directory with 4 gigabytes available to unzip the SQL Server software
- Permissions to create an SQL Server database:
 - System Stored Procedures (SP)
 - CREATE LOGIN
 - If using a non-default database owner, the `sp_changedbowner` process is available
 - If a non-default database owner is specified and does not exist, permission to create the appropriate login
- If you plan to run as a Windows domain user, .NET 3.5 is installed.

Note: For additional information, see "Run as a Windows Domain User" in the *HPE DMA Installation Guide*, available at: softwaresupport.hp.com

- Licenses for SQL Server and HPE DMA.

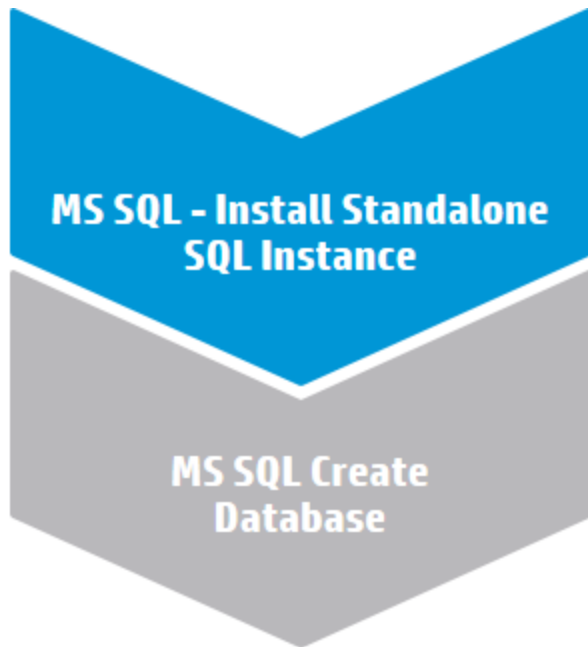
For additional requirements, see the following Microsoft documentation:

SQL Server version	Microsoft documentation
2008	Hardware and Software Requirements for Installing SQL Server 2008
2008 R2	Hardware and Software Requirements for Installing SQL Server 2008 R2
2012	Hardware and Software Requirements for Installing SQL Server 2012
2014	Hardware and Software Requirements for Installing SQL Server 2014

Process Overview

Patching SQL Server Grid Infrastructure with HPE DMA is a simple, one-step process. All required checks and steps to patch an SQL Server Grid infrastructure—including the Grid Home, all Database Homes, and the databases—have been implemented in a single HPE DMA workflow.

Workflow 1: MS SQL - Install Standalone SQL Instance



This section provides detailed information required to run the MS SQL - Install Standalone SQL Instance workflow.

Tip: To provision multiple SQL Server standalone instances, run MS SQL - Install Standalone SQL Instance once for each.

To create the configuration, you will run this workflow once to create InstanceA and a second time to create InstanceB. All other parameter values should be identical.

Solution pack

This workflow requires the Database Provisioning Solution Pack.

Parameters to expose

None

Input parameters

When you deploy the MS SQL - Install Standalone SQL Instance workflow, specify input parameter values for the following steps.

Bold text in the following tables indicates that you must specify a value for the parameter.

HPE DMA uses the latest (highest version number) of steps.

Step: MS SQL - Parameters - Install Standalone

Parameter	Description	Example Value
Download From Software Directory	<p>Optional: The name of the ZIP file that contains the SQL Server installation software files obtained from Microsoft.¹</p> <p>Note: If necessary, manually zip the installation software files up.</p>	SQL12.zip
Download Target Destination	<p>Required: The local directory where the SQL Server setup files are stored:</p> <p>If source files are in the software repository: Location where Download From Software Directory will be downloaded and extracted</p> <p>If source files are on the target: Location where the Microsoft SQL Server installation files already exist—not zipped up</p> <p>Upon a successful workflow completion, all downloaded and extracted files are cleaned up.</p>	C:\temp
Instance Name	<p>Required: The name of the newly created instance. Use MSSQLSERVER for the default instance and any other alphanumeric value for a named instance.</p>	<p>InstanceA</p> <p><i>When you run the workflow again for the second instance use the value InstanceB.</i></p>

Step: MS SQL - Advanced Parameters - Install Standalone

Parameter	Description	Example Value
Installer Account	Optional: The Windows account that will perform the installation.	?
Installer Password	Optional: The password of the Windows account that will perform the installation	?
SQL Service Account	Optional: The login account for the SQL Server service. Can be a local Windows user, a domain user, or a	Win12\Administrator

¹If the file is not found on the target server(s), it will be downloaded from the software repository.

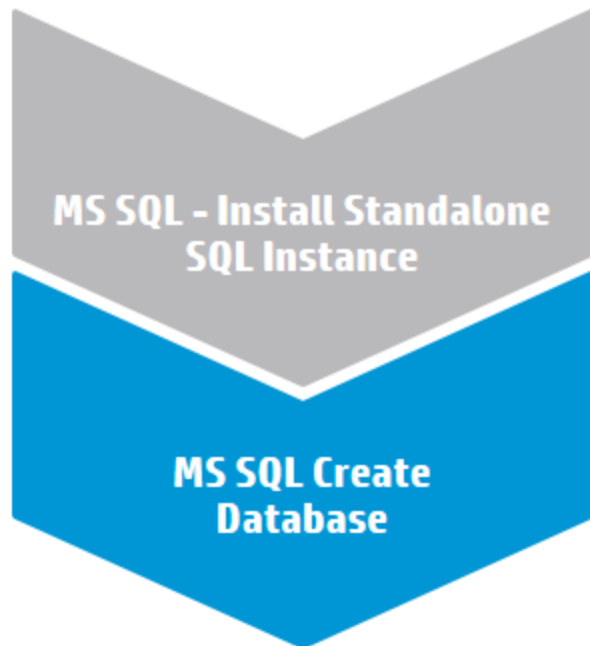
Step: MS SQL - Advanced Parameters - Install Standalone, continued

Parameter	Description	Example Value
	built-in account (for example, NT AUTHORITY\NETWORK SERVICE). If not a built-in account, also specify SQL Service Password.	
SQL Service Password	Required if SQL Service Account is specified and is not a built-in account.	●●●

Step: Discover SQL Databases

Parameter	Description	Example Value
Web Service Password	Required: Password for the HPE DMA Discovery web service API.	●●●
Web Service User	Required: User who is capable of modifying the managed environment by using the HPE DMA Discovery web service API.	dmawebuser

Workflow 2: MS SQL Create Database



This section provides detailed information required to run the MS SQL Create Database workflow.

Solution pack

This workflow requires the Database Provisioning Solution Pack.

Parameters to expose

None

Input parameters

When you deploy the MS SQL Create Database workflow, specify input parameter values for the following step.

Bold text in the following tables indicates that you must specify a value for the parameter.

HPE DMA uses the latest (highest version number) of steps.

Step: MS SQL Parameters Create Database

Parameter	Description	Example Value
Database Name	Required: Name of the new database.	NewDatabase
Web Service Password	Required: Password for the HPE DMA Discovery web service API.	●●●
Web Service User	Required: User who is capable of modifying the managed environment by using the HPE DMA Discovery web service API.	dmawebuser

Refreshing Database

This section describes the SQL Server workflows included in the HPE Database and Middleware Automation (HPE DMA) Database Refresh solution pack.

Database refresh involves copying the contents of one database into a database in the same or another SQL Server instance. This is useful, for example, if you want to move a database from a traditional IT infrastructure to a private cloud. It is also useful if you want to duplicate production data in a test environment for application development or troubleshooting purposes.

The workflows in this solution pack enable you to automate and simplify the following operations:

- Extracting the contents of a database into a backup file
- Restoring a database from an existing backup file
- Extracting the contents of one database and loading them into another database using a single **bridged execution** workflow that performs both steps

The workflows perform extensive validation checks prior to and immediately after the database backup and restore operations to ensure that the refresh is successful.

After a refresh is completed, the restore workflows can re-create any existing database users and roles.

The workflows can create or utilize a database backup file that is compressed, encrypted, or both.

MS SQL - Backup Database

This workflow enables you to backup a SQL Server database into file (the backup file) that is stored either locally or on a network share.

You can specify various options for the backup operation, including whether the backup file is compressed or encrypted with a password.

The workflow performs extensive validation checks prior to and immediately after the backup operation to ensure that the backup file is valid. The workflow will perform an additional integrity check on the backup file if you set the Perform Integrity Check parameter to YES.

To use this workflow in your environment, see the following information:

Topic	Information Included
"Prerequisites for this Workflow"	List of prerequisites that must be satisfied before you can run this workflow
"How this Workflow Works"	Information about what the workflow does, including validation checks performed, steps executed, and a high-level process flow
"How to Run this Workflow"	Instructions for running this workflow in your environment
"Sample Scenarios"	Examples of typical parameter values for this workflow
"Parameters for Backup MS SQL Database"	List of input parameters for this workflow

The process of deploying and running this workflow is the same for all scenarios, but the parameters required will differ depending on the specific scenario that you are implementing.

The workflow provides default values for most parameters. These default values are usually sufficient for a "typical" database backup. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios.

Any parameters not explicitly specified in the deployment will have the default values listed in ["Parameters for Backup MS SQL Database" on page 49](#) .

Prerequisites for this Workflow

The following prerequisites must be satisfied before you can run the ["MS SQL - Backup Database"](#) workflow:

1. The service login for the SQL Server service must have read and write permissions on the backup path.
2. The server management agent must have login access to the SQL Server instance in which the target database resides. It must also have permission to perform database consistency check (DBCC) commands on the target database.
3. There must be sufficient space available on the target data and log disks. The workflow checks for this, and will fail if sufficient space is not available.

Additional Considerations

For information about prerequisites for SQL Server, refer to the [SQL Server Product Documentation](#).

How this Workflow Works

This topic contains the following information about the ["MS SQL - Backup Database"](#) workflow:

Validation Checks Performed

The workflow checks the following things prior to dumping the database. If any of these checks fails, the workflow fails.

1. All required parameters have values. If any required parameter does not have a value—either a value that you specify or a default value—the workflow fails in the Run MS SQL Pre-Backup Validation step.
2. The Target Backup Path is accessible, either locally or on a network share.
If the Target Backup Path is on a network share, the Windows Share User has read and write access the share.
3. The target database exists, and the workflow can connect to it.
4. Adequate disk space is available to store the database backup file.
5. If the Target Backup Path does not currently exist, it will be created prior to creating the backup file.

Steps Executed

The ["MS SQL - Backup Database"](#) workflow includes the following steps. Each step must complete successfully before the next step can start. If a step fails, the workflow reports a failure, and all subsequent steps are skipped.

Process Flow

This workflow performs the following tasks:

1. Performs the preliminary [validation checks](#) described above.
2. Performs the database backup operation to create the backup file.
3. Performs post-backup validation checks to ensure that all required parameters had valid values.
4. If Perform Integrity Check was set to YES, performs an integrity check on the backup file.

Tips and Best Practices

It is good practice to run basic database consistency checks (DBCCs) on the source database before running this workflow to ensure that there are no internal errors in the database.

If you find errors in the source database, be sure to fix them before running this workflow. The workflow does not have the ability to diagnose or remediate problems in the database prior to performing the database backup.

How to Run this Workflow

This topic explains how to customize and run the "MS SQL - Backup Database" workflow in your environment.

Note: Prior to running this workflow, review the "Prerequisites for this Workflow", and ensure that all requirements are satisfied.

To customize and run the Backup MS SQL Database workflow:

1. Create a deployable copy of the workflow.
2. Determine the values that you will specify for the following parameter. This is the minimum set of parameters required to run this workflow.

Parameter Name	Default Value	Description
Target Backup Path	no default	<p>Where the database backup file will be stored, either locally or on a network share. You can specify both the path and file name, or you can specify only the path.</p> <ul style="list-style-type: none"> • If you specify a file name, it must end in .bak. • If you do not specify a file name, the backup file name will have the following form: <code><dataBaseName>_<dateTime>.bak</code> where <code><dataBaseName></code> represents the name of the target database specified when the workflow runs, and <code><dateTime></code> is the date and time when the Run MS SQL Pre-Backup Validation step is executed. <p>If the file will be stored on a network share, the Windows Share User must have read and write access to that share.</p>

3. See "Parameters for Backup MS SQL Database" on page 49 for detailed descriptions of all input parameters for this workflow, including default values. In the workflow editor, expose any additional parameters that you need. You will specify values for those parameters when you create the deployment or at runtime.
4. Save the changes to the workflow (click **Save** in the lower right corner).
5. Create a new deployment.
6. On the Parameters tab, specify values (or set the type to Runtime Value) for the required parameters listed in step 2 and any additional parameters that you have exposed. You do not need to specify values for those parameters whose default values are appropriate for your environment.
7. On the Targets tab, specify one or more targets for this deployment.
8. Save the deployment (click **Save** in the lower right corner).
9. Run the workflow using this deployment, specifying any runtime parameters.

The workflow will complete and report "Success" on the Console if it has run successfully. If an invalid parameter value is specified, an error is logged, and the workflow terminates in the "Failure" state.

Sample Scenarios

This topic shows you how to use various parameters to achieve the following database backup scenarios in your environment using the ["MS SQL - Backup Database"](#) workflow:

Scenario 1: Create a Backup File that is Not Encrypted or Compressed

This is the simplest SQL Server database backup scenario. In this example, the backup file is stored on a network share.

Step Name	Parameter Name	Example Value
Gather Parameters for MS SQL Database Backup	Target Backup Path	\\WIN-DOMAIN-CTRL\Backups
Gather Advanced Parameters for MS SQL Database Backup	Windows Share Password	WinSharePwd To avoid having to re-enter passwords whenever they change, you can create a policy to provide them to the workflow.
	Windows Share User	WIN\Administrator

Be sure that the default values for all remaining parameters are appropriate for your environment (see ["Parameters for Backup MS SQL Database" on page 49](#)).

Scenario 2: Create a Backup File that is Encrypted and Compressed

This scenario requires you to specify the encryption password and compression option for the database backup file. In this example, the backup file is stored in locally on the server that hosts the target database.

Step Name	Parameter Name	Example Value
Gather Parameters for MS SQL Database Backup	Target Backup Path	c:\Backups\mytestdb_03122012.bak
Gather Advanced Parameters for MS SQL Database Backup	Backup Encryption Password	EncryptMyBackup
	Compress Backup File	YES

Be sure that the default values for all remaining parameters are appropriate for your environment (see ["Parameters for Backup MS SQL Database" on page 49](#)).

Scenario 3: Create a Backup File, Perform an Integrity Check, and Configure Windows Domain User Using Runtime Parameters

This scenario runs an integrity check on the backup file after the backup is performed. In this example, the backup file is stored locally on the server that hosts the target database.

Note: You may want to run this workflow against a MS SQL instance that can only be accessed by a Windows user with a temporary password. By using a runtime parameter for the password, you can ensure that the password used is always the latest.

To specify the Windows domain user at the time you execute a deployment with runtime parameters, perform the following additional steps:

1. When you make a copy of the workflow, expand the appropriate step, and then set the Windows domain user parameters—Instance Account and Instance Password—to **- User selected -**.
2. When you create a deployment from the copy of the workflow, set the parameter types to **Runtime Value**.
3. When you execute the deployment, specify the Windows domain user account and password.

Step Name	Parameter Name	Example Value
Gather Parameters for MS SQL Database Backup	Target Backup Path	c:\Backups\mytestdb_03122012.bak
Gather Advanced Parameters for MS SQL Database Backup	Perform Integrity Check	YES
	Instance Account	Domain\DomainUserAcct Note: Enter at runtime.
	Instance Password	DomainUserPswd Note: Enter at runtime.

Be sure that the default values for all remaining parameters are appropriate for your environment (see ["Parameters for Backup MS SQL Database" on the next page](#)).

Parameters for Backup MS SQL Database

The following tables describe the required and optional input parameters for this workflow. Most of these parameters are not initially visible in a deployment. For most parameters, if you do not specify a value for a parameter, a default value is assigned

Parameters Defined in this Step: Gather Parameters for MS SQL Database Backup

Parameter Name	Default Value	Required	Description
Target Backup Path	no default	required	<p>Where the database backup file will be stored, either locally or on a network share. You can specify both the path and file name, or you can specify only the path.</p> <ul style="list-style-type: none"> If you specify a file name, it must end in <code>.bak</code>. If you do not specify a file name, the backup file name will have the following form: <code><dataBaseName>_<dateTime>.bak</code> where <code><dataBaseName></code> represents the name of the target database specified when the workflow runs, and <code><dateTime></code> is the date and time when the Run MS SQL Pre-Backup Validation step is executed. <p>If the file will be stored on a network share, the Windows Share User must have read and write access to that share.</p>

Additional Parameters Defined in this Step: Gather Advanced Parameters for MS SQL Database Backup

Parameter Name	Default Value	Required	Description
Backup Description	no default	optional	Text that describes this backup (up to 255 characters).
Backup Encryption Password	no default	optional	<p>To encrypt the backup file with a password, specify the password in this parameter.</p> <p>If you perform the backup using a password, you must also specify that password when you perform the restore.</p>
Backup Name	no default	optional	The name of this backup (up to 128 characters).
Compress Backup File	NO	optional	<p>If you specify YES, the backup file will be compressed. Valid values: YES or NO.</p> <p>Compression is supported on SQL Server 2008 Enterprise and later. If you are running SQL 2005, and this parameter is set to</p>

Additional Parameters Defined in this Step: Gather Advanced Parameters for MS SQL Database Backup, continued

Parameter Name	Default Value	Required	Description
			YES, the workflow will ignore this value and continue without compression.
Expiration Date	no default	optional	<p>Date and time when the backup file expires and the backup data is no longer considered relevant. After this date and time, SQL Server is not prevented from overwriting this backup file.</p> <p>The Expiration Date must be specified in a format compatible with the configured system datetime format.</p> <p>If both the Retention Days and the Expiration Date parameters are specified, the Retention Days parameter takes precedence.</p>
Instance Account	no default	optional	The Windows account that will perform the backup operation.
Instance Password	no default	optional	The password for the Windows account that will perform the backup operation.
Perform Integrity Check	NO	optional	If you specify YES, the workflow will perform an integrity check on the database backup file. Valid values: YES or NO.
Retention Days	no default	optional	<p>Number of days after which the backup data is no longer considered relevant. After this number of days, SQL Server is not prevented from overwriting this backup file.</p> <p>If both the Retention Days and the Expiration Date parameters are specified, the Retention Days parameter takes precedence.</p>
Windows Share Password	no default	optional	Password for the user specified in Windows Share User.

Additional Parameters Defined in this Step: Gather Advanced Parameters for MS SQL Database Backup, continued

Parameter Name	Default Value	Required	Description
Windows Share User	no default	optional	Windows user who can access the specified Windows network share and who will own (and write) the backup file.

MS SQL - Restore Database

This workflow enables you to restore a SQL Server database from a previously created database backup file that is stored locally, on a network share, or in the software repository.

If the database does not exist in the target instance, the workflow will create it. If the database already exists, you can specify whether you want the workflow to overwrite its contents. You can also specify whether existing database users should be re-created after the restore operation—in which case, any users included in the backup file are ignored.

Note: The parameters required to activate these options are hidden by default.

This workflow also provides a "simulation mode" where the Run MS SQL Pre-Restore Validation step is executed, but the restore is not performed. This is useful for testing or troubleshooting your parameter values.

The workflow performs extensive validation checks prior to and immediately after the restore operation to ensure that both the backup file and the restored database are valid.

The process of deploying and running this workflow is the same for all scenarios, but the parameters required will differ depending on the specific scenario that you are implementing.

The workflow provides default values for most parameters. These default values are usually sufficient for a "typical" database refresh. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios.

Topic	Information Included
"Prerequisites for this Workflow"	List of prerequisites that must be satisfied before you can run this workflow
"How this Workflow Works"	Information about what the workflow does, including validation checks performed, steps executed, and a high-level process flow
"How to Run this Workflow"	Instructions for running this workflow in your environment
"Sample Scenarios"	Examples of typical parameter values for this workflow
"Parameters for Restore MS SQL Database"	List of input parameters for this workflow

The process of deploying and running this workflow is the same for all scenarios, but the parameters required will differ depending on the specific scenario that you are implementing.

The workflow provides default values for most parameters. These default values are usually sufficient for a "typical" database restore. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios.

Any parameters not explicitly specified in the deployment will have the default values listed in ["Parameters for Restore MS SQL Database" on page 61](#) .

Prerequisites for this Workflow

The following prerequisites must be satisfied before you can run the ["MS SQL - Restore Database"](#) workflow:

1. The service login for the SQL Server service must have read and write permissions on the backup file.
2. The server management agent must have login access to the target SQL Server instance. It must also have permission to create a new database and perform database consistency check (DBCC) commands on the restored database.
3. There must be sufficient space available on the target data and log disks. The workflow checks for this, and will fail if sufficient space is not available.

Additional Considerations

For information about prerequisites for SQL Server, refer to the [SQL Server Product Documentation](#).

How this Workflow Works

This topic contains the following information about the "[MS SQL - Restore Database](#)" workflow:

Validation Checks Performed

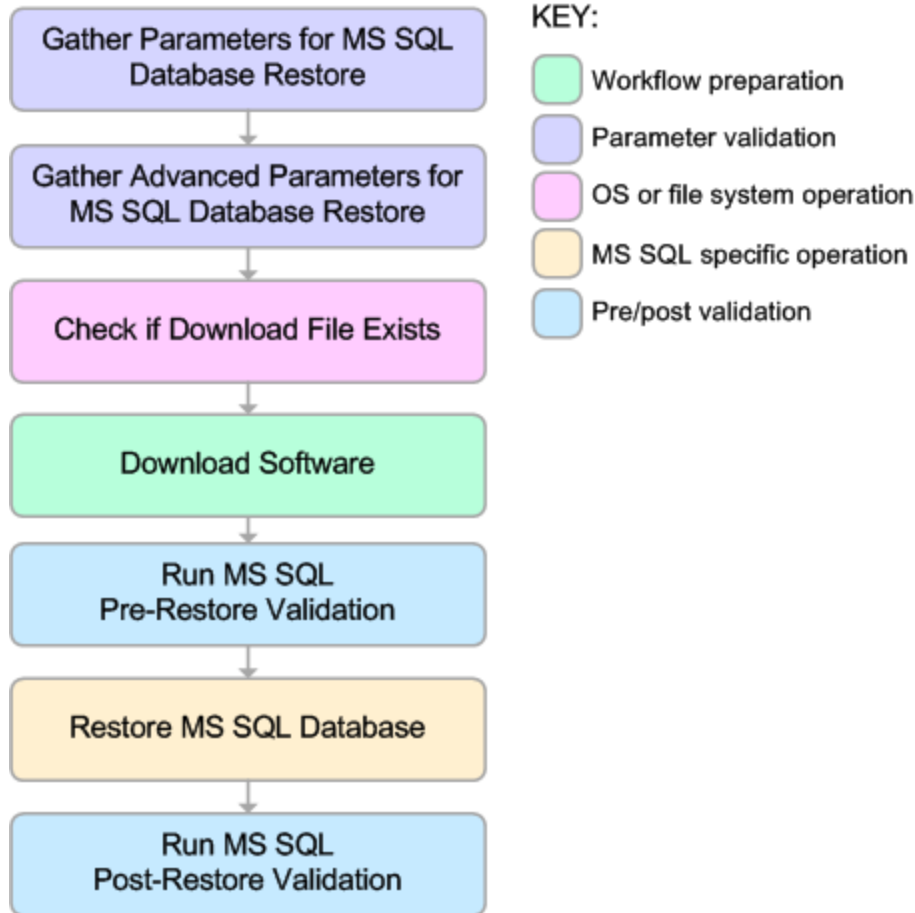
The workflow checks the following things prior to dumping the database. If any of these checks fails, the workflow fails.

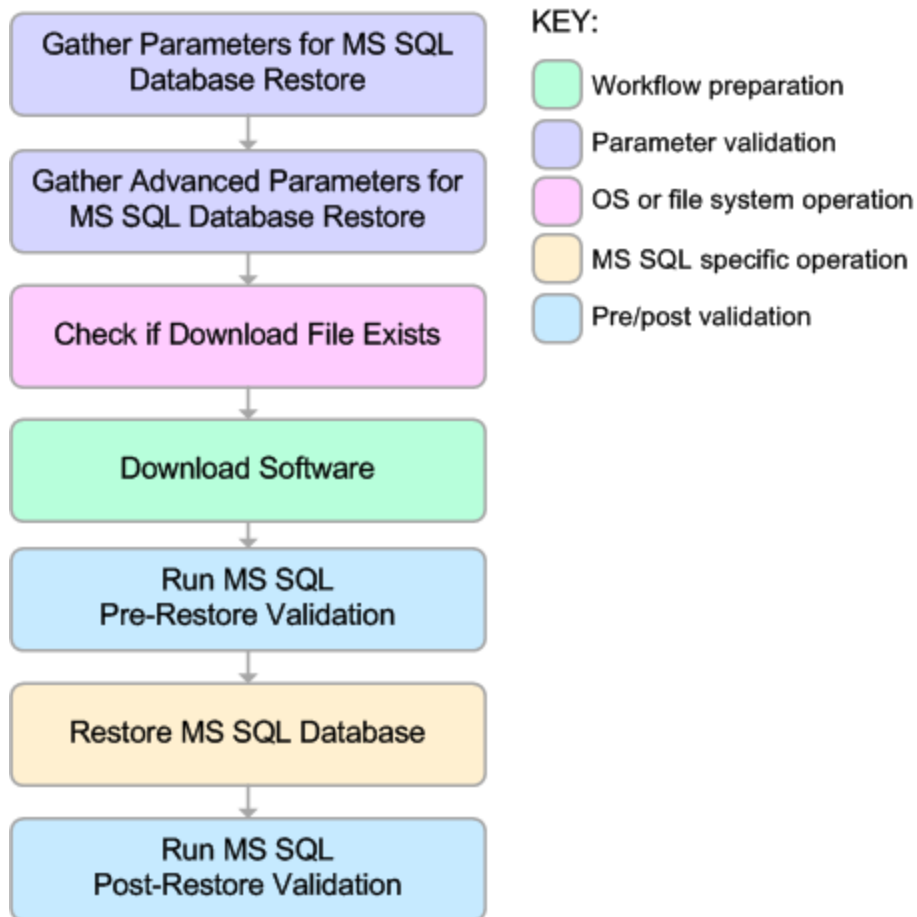
1. All required parameters have values. If any required parameter does not have a value—either a value that you specify or a default value—the workflow fails in the Run MS SQL Pre-Restore Validation step.
2. The specified backup file either exists in the Download Target Destination directory or can be downloaded from the software repository.
3. The backup file is compatible with the target instance.
4. If the Custom Database Name parameter is specified, this database name complies with SQL Server database naming conventions.
5. The Download Target Destination is accessible, either locally or on a network share.
If the Download Target Destination is on a network share, the Windows Share User has read and write access to the share.
6. The target instance exists, and the workflow can connect to it.
7. Adequate disk space is available to restore the data and log files.
8. If custom paths are specified for the data or log files, the Run MS SQL Pre-Restore Validation step checks that they exist (and creates them if they don't), and ensures that the quantity of paths specified match the quantity of files in the backup file.

Steps Executed

The "MS SQL - Restore Database" workflow includes the following steps. Each step must complete successfully before the next step can start. If a step fails, the workflow reports a failure, and all subsequent steps are skipped.

Click each box in the diagram to view additional information about that step in a new window.





Process Flow

This workflow performs the following tasks:

1. Performs the preliminary [validation checks](#) described above.
2. If Preserve Users and Roles was set to YES, creates the Roles Creation Script and the Users Creation Script script.
3. If not in simulation mode, performs the database restore operation to load the contents of the backup file.
4. Performs post-restore validation checks to ensure that the restored database is sound.
5. If Preserve Users and Roles was set to YES, re-creates any existing database users and roles.
6. If Reindex Restored Database was set to YES, re-indexes the database.

Tips and Best Practices

It is good practice to run basic database consistency checks (DBCCs) on the source database before you create the database backup to ensure that there are no internal errors in the database.

If you find errors in the source database, be sure to fix them before you create the database backup. This workflow does not have the ability to diagnose or remediate problems in the database prior to performing the database backup.

How to Run this Workflow

This topic explains how to customize and run the "MS SQL - Restore Database" workflow in your environment.

Note: Prior to running this workflow, review the "Prerequisites for this Workflow", and ensure that all requirements are satisfied.

To customize and run the Restore MS SQL Database workflow:

1. Create a deployable copy of the workflow (see "Create a Deployable Workflow" in HPE DMA Quick Start Tutorial).
2. Determine the values that you will specify for the following parameters. This is the minimum set of parameters required to run this workflow.

Parameter Name	Default Value	Description
Database Backup File	no default	<p>Path where the database backup file is (or will be) stored, either locally or on a network share.</p> <p>If the file already exists locally or on a network share, specify the file name in this parameter and the path in the Download Target Destination parameter.</p> <p>If the file does not yet exist locally or on a network share, it will be downloaded into this location from the software repository.</p> <p>If the file is (or will be) stored on a network share, the Windows Share User must have read and write access to that share.</p> <p>Note: Windows Share User and Windows Share Password are not exposed by default.</p>
Download Target Destination	no default	<p>The directory where the database backup file will be stored.</p> <p>If the database backup file does not yet exist in this directory, it will be downloaded from the software repository and stored in this directory.</p>

See "Parameters for Restore MS SQL Database" on page 61 for detailed descriptions of all input parameters for this workflow, including default values.

3. In the workflow editor, expose any additional parameters that you need. You will specify values for those parameters when you create the deployment or at runtime.
4. Save the changes to the workflow (click **Save** in the lower right corner).
5. Create a new deployment.
6. On the Parameters tab, specify values (or set the type to Runtime Value) for the required parameters listed in step 2 and any additional parameters that you have exposed. You do not need to specify values for those parameters whose default values are appropriate for your environment.
7. On the Targets tab, specify one or more targets for this deployment.
8. Save the deployment (click **Save** in the lower right corner).
9. Run the workflow using this deployment, specifying any runtime parameters.

The workflow will complete and report “Success” on the Console if it has run successfully. If an invalid parameter value is specified, an error is logged, and the workflow terminates in the “Failure” state.

Sample Scenarios

This topic shows you how to use various parameters to achieve the following database backup scenarios in your environment using the ["MS SQL - Restore Database"](#) workflow:

Scenario 1: Restore from a Backup File that is Not Encrypted or Compressed

This is the simplest SQL Server database restore scenario. In this example, the backup file has been stored on a network share (or will be downloaded from the software repository and stored on the share).

Note that the Windows Share User and Windows Share Password are specified in this scenario. This is not required, but it facilitates the disk space check on the network path. If you do not specify this parameter, this check is skipped.

Step Name	Parameter Name	Example Value
Gather Parameters for MS SQL Database Restore	Database Backup File	mytestdb_03122012.bak
	Download Target Destination	\\WIN-DOMAIN-CTRL\Backups
Gather Advanced Parameters for MS SQL Database Restore	Windows Share Password	WinSharePwd Tip: To avoid having to re-enter passwords whenever they change, you can create a policy to provide them to the workflow.
	Windows Share User	WIN\Administrator

Be sure that the default values for all remaining parameters are appropriate for your environment (see ["Parameters for Restore MS SQL Database" on page 61](#)).

Scenario 2: Restore from a Backup File that is Encrypted and Compressed

This scenario requires you to specify the encryption password for the database backup file. The workflow automatically handles the compression, so there is no need to specify the compression parameter. In this example, the backup file is stored locally on the server where the target instance resides.

Step Name	Parameter Name	Example Value
Gather Parameters for MS SQL Database Restore	Database Backup File	c:\Backups\mytestdb_03122012.bak
Gather Advanced Parameters for MS SQL Database Restore	Backup Encryption Password	EncryptMyBackup

Be sure that the default values for all remaining parameters are appropriate for your environment (see ["Parameters for Restore MS SQL Database" on page 61](#)).

Scenario 3: Overwrite an Existing Database, Restore Users, and Configure Windows Domain User Using Runtime Parameters

This scenario overwrites an existing database and restores any existing users after the restore is performed. In this example, the backup file is stored locally on the server where the target database resides.

Note: You may want to run this workflow against a MS SQL instance that can only be accessed by a Windows user with a temporary password. By using a runtime parameter for the password, you can ensure that the password used is always the latest.

To specify the Windows domain user at the time you execute a deployment with runtime parameters, perform the following additional steps:

1. When you make a copy of the workflow, expand the appropriate step, and then set the Windows domain user parameters—Instance Account and Instance Password—to **- User selected -**.
2. When you create a deployment from the copy of the workflow, set the parameter types to **Runtime Value**.
3. When you execute the deployment, specify the Windows domain user account and password.

Step Name	Parameter Name	Example Value
Gather Parameters for MS SQL Database Restore	Database Backup File	c:\Backups\mytestdb_03122012.bak
Gather Advanced Parameters for MS SQL Database Restore	Overwrite Existing Database	YES
	Preserve Users and Roles	YES
	Instance Account	Domain\DomainUserAcct Note: Enter at runtime.
	Instance Password	DomainUserPswd Note: Enter at runtime.

Be sure that the default values for all remaining parameters are appropriate for your environment (see ["Parameters for Restore MS SQL Database" on the next page](#)).

Parameters for Restore MS SQL Database

The following tables describe the required and optional input parameters for this workflow. Most of these parameters are not initially visible in a deployment. For most parameters, if you do not specify a value for a parameter, a default value is assigned

Parameters Defined in this Step: Gather Parameters for MS SQL Database Restore

Parameter Name	Default Value	Required	Description
Database Backup File	no default	required	<p>Path where the database backup file is (or will be) stored, either locally or on a network share.</p> <p>If the file already exists locally or on a network share, specify the file name in this parameter and the path in the Download Target Destination parameter.</p> <p>If the file does not yet exist locally or on a network share, it will be downloaded into this location from the software repository.</p> <p>If the file is (or will be) stored on a network share, the Windows Share User must have read and write access to that share.</p>
Download Target Destination	no default	required	<p>The directory where the database backup file will be stored.</p> <p>If the database backup file does not yet exist in this directory, it will be downloaded from the software repository and stored in this directory.</p>

Additional Parameters Defined in this Step: Gather Advanced Parameters for MS SQL Database Restore

Parameter Name	Default Value	Required	Description
Backup Encryption Password	no default	optional	To decrypt a backup file that was encrypted with a password, specify the password in this parameter.
Data File Locations	no default	optional	<p>Comma-delimited list of directories or full file paths for each data file in the backup file.</p> <p>Use Run Simulation Only mode to discover the number of data files in the backup file. If this parameter is not specified, the original data file names and paths will be used.</p>
Database Name	no default	optional	To restore the database from the backup file using a different database name, specify that name here. If this parameter is not specified, the original database name will be used.
Instance Account	no default	optional	The Windows account that will perform the restore operation.
Instance Password	no default	optional	The password for the Windows account that will perform the restore operation.
Log File	no	optional	Comma-delimited list of directories or full file paths for each log file

Additional Parameters Defined in this Step: Gather Advanced Parameters for MS SQL Database Restore , continued

Parameter Name	Default Value	Required	Description
Locations	default		in the backup file. Use Run Simulation Only mode to discover the number of log files in backup file. If this parameter is not specified, the original log file names and paths will be used.
Overwrite Existing Database	NO	optional	If set to YES, and the database already exists, the workflow will overwrite the database. Valid values: YES or NO. If set to NO, and the database already exists, the workflow will fail.
Preserve Users and Roles	NO	optional	If set to YES, and the database already exists, the workflow will try to preserve the database users and role. Valid values: YES or NO.
Reindex Restored Database	NO	optional	If set to YES, the workflow will re-index the database after the restore operation is successfully completed. Valid values: YES or NO. Re-indexing improves database performance. More specifically , it recreates all the table look-ups and performance tunes them according to the new environment. This is important when you are restoring a database in a new environment that it has never seen before.
Run Simulation Only	NO	optional	If set to YES, the workflow will only run the Pre-Restore Validation step. It will not attempt to restore the database. Use this mode to discover the original data and log files used for the database backup. Valid values: YES or NO.

MS SQL - Backup and Restore Database

This workflow enables you to backup the contents of a SQL Server database (the **source database**) into a file and restore a database in another instance (the **target instance**) using the contents of that backup file. The source database and target instance are specified at run time.

This is a **bridged execution** workflow. The first group of steps performs the backup on the specified source database. The second group of steps performs the restore on the specified database in the specified target instance.

You can specify various options, including whether the backup file is compressed or encrypted with a password.

Note: Bridged execution workflows work on one target level (server, instance, or database). This workflow runs on the database level at all times. When choosing a target instance at run time, you will actually see a list of databases that reside on each instance. You can select any database in the target instance where you want to perform the restore.

If you specify the RESTORE - Database Name parameter, the workflow will use that database. If you do not specify the RESTORE - Database Name parameter, the workflow will use the original database name from the backup.

If the database specified in the Database Name parameter does not exist in the target instance, the workflow will create it. If the database already exists, you can specify whether you want the workflow to overwrite its contents. You can also specify whether existing database users should be re-created after the restore operation—in which case, any users included in the backup file are ignored .

This workflow also provides a "simulation mode" where the Run MS SQL Pre-Restore Validation step is executed, but the restore is not performed. This is useful for testing or troubleshooting your parameter values.

The workflow performs extensive validation checks prior to and immediately after both the backup and restore operations to ensure that both the backup file and the restored database are valid.

See "[Parameters for Backup and Restore MS SQL Database](#)" on page 75 for a list of backup and restore options that you can specify. Many of these parameters are hidden by default

The process of deploying and running this workflow is the same for all scenarios, but the parameters required will differ depending on the specific scenario that you are implementing.

The workflow provides default values for most parameters. These default values are usually sufficient for a "typical" database refresh. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios.

Topic	Information Included
"Prerequisites for this Workflow"	List of prerequisites that must be satisfied before you can run this workflow
"How this Workflow Works"	Information about what the workflow does, including validation checks performed, steps executed, and a high-level process flow
"How to Run this Workflow"	Instructions for running this workflow in your environment
"Sample Scenarios"	Examples of typical parameter values for this workflow
"Parameters for Backup and Restore MS SQL Database"	List of input parameters for this workflow

The process of deploying and running this workflow is the same for all scenarios, but the parameters required will differ depending on the specific scenario that you are implementing.

The workflow provides default values for most parameters. These default values are usually sufficient for a "typical" database backup and restore. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios.

Any parameters not explicitly specified in the deployment will have the default values listed in ["Parameters for Backup and Restore MS SQL Database" on page 75](#).

Prerequisites for this Workflow

The following prerequisites must be satisfied before you can run the ["MS SQL - Restore Database"](#) workflow:

1. The service login for the SQL Server service must have read and write permissions on the location where the backup file will be stored.
2. The server management agent must have login access to the target SQL Server instance. It must also have permission to create a new database and perform database consistency check (DBCC) commands on the restored database.
3. There must be sufficient space available to create the backup file and restore the database (including both data and logs). The workflow checks for this, and will fail if sufficient space is not available.

Additional Considerations

For information about prerequisites for SQL Server, refer to the [SQL Server Product Documentation](#).

How this Workflow Works

This topic contains the following information about the "[MS SQL - Backup and Restore Database](#)" workflow:

Validation Checks Performed

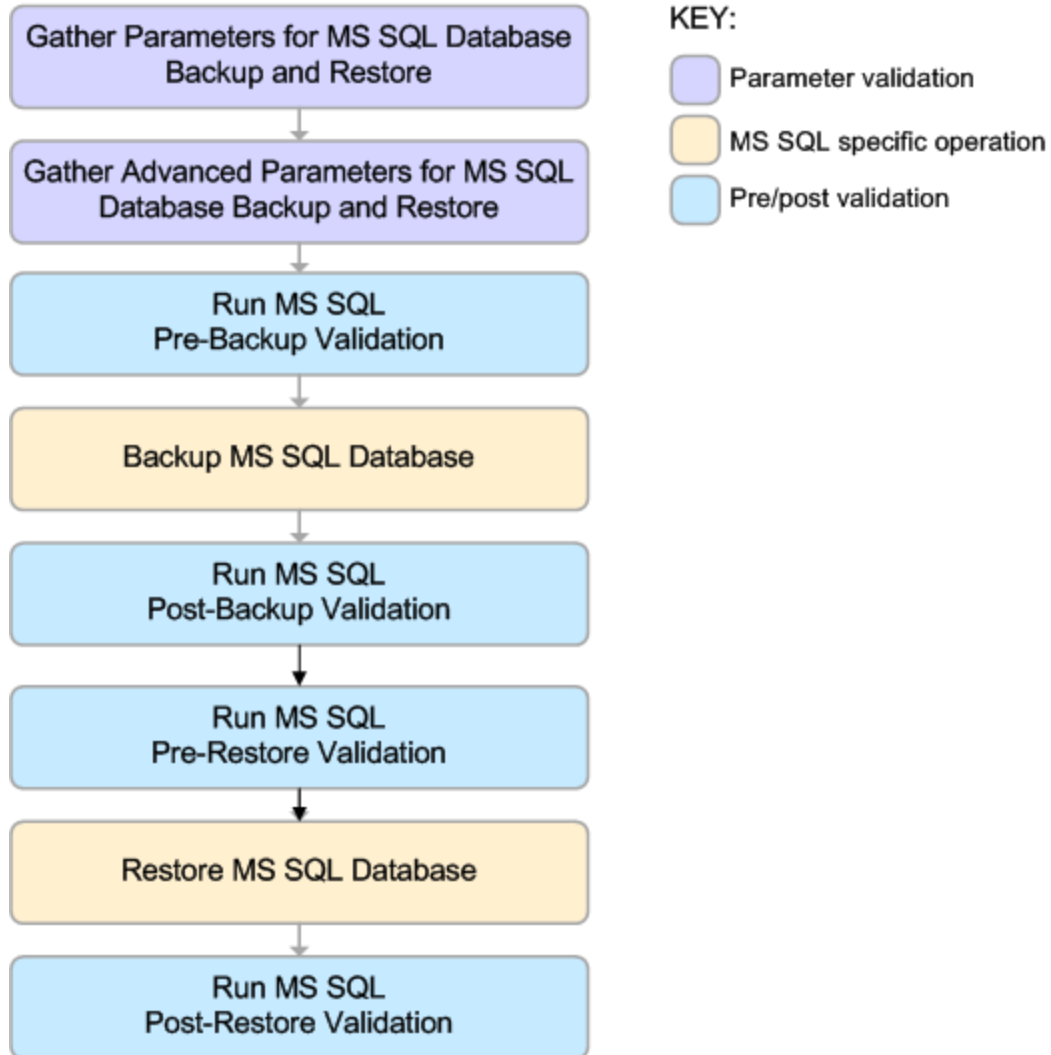
The workflow checks the following things prior to dumping the database. If any of these checks fails, the workflow fails.

1. All required parameters have values. If any required parameter does not have a value—either a value that you specify or a default value—the workflow fails in either the Run MS SQL Pre-Backup Validation step or the Run MS SQL Pre-Restore Validation step.
2. The Working Path is accessible, either locally or on a network share.
If the Working Path is on a network share, the BACKUP - Windows Share User has read and write access the share.
3. The source database is compatible with the target instance.
4. If the RESTORE - Database Name parameter is specified, this database name complies with SQL Server database naming conventions.
5. The target instance exists, and the workflow can connect to it.
6. Adequate disk space is available to backup and restore the data and log files.

Steps Executed

The "MS SQL - Backup and Restore Database" workflow includes the following steps. Each step must complete successfully before the next step can start. If a step fails, the workflow reports a failure, and all subsequent steps are skipped.

Click each box in the diagram to view additional information about that step in a new window.



Process Flow

This workflow performs the following tasks:

1. Performs the preliminary [validation checks](#) described above.
2. If RESTORE - Preserve Users and Roles was set to YES, creates the Roles Creation and Users Creation scripts.
3. Performs the database backup operation to create the backup file.
4. Performs post-backup validation checks to ensure that all required parameters had valid values.
5. If BACKUP - Perform Integrity Check was set to YES, performs an integrity check on the backup file.
6. If not in simulation mode, performs the database restore operation to load the contents of the backup file.
7. Performs post-restore validation checks to ensure that the restored database is sound.
8. If RESTORE - Preserve Users and Roles was set to YES, re-creates any existing database users and roles.
9. If RESTORE - Reindex Restored Database was set to YES, re-indexes the database.

Tips and Best Practices

It is good practice to run basic database consistency checks (DBCCs) on the source database before you create the database backup to ensure that there are no internal errors in the database.

If you find errors in the source database, be sure to fix them before you run this workflow. This workflow does not have the ability to diagnose or remediate problems in the database prior to performing the database backup.

How to Run this Workflow

This topic explains how to customize and run the "MS SQL - Backup and Restore Database" workflow in your environment.

Note: Prior to running this workflow, review the "Prerequisites for this Workflow", and ensure that all requirements are satisfied.

To customize and run the Backup and Restore MS SQL Database workflow:

11. Create a deployable copy of the workflow.
2. Determine the values that you will specify for the following parameter. This is the minimum set of parameters required to run this workflow.

Parameter Name	Default Value	Description
Working Path	no default	<p>The directory where the database backup file will be stored. This can be a directory or a full file path. This path must be accessible to both the source and target servers.</p> <p>Be sure to specify this parameter in network share notation (for example: \\<network share>\). A network path can be located on a target server, but it should always be referenced using network share notation instead of local folder notation (for example: C:\<folder>).</p> <p>You specify this parameter in the deployment.</p>

See "Parameters for Backup and Restore MS SQL Database" on page 75 for detailed descriptions of all input parameters for this workflow, including default values.

3. In the workflow editor, expose any additional parameters that you need. You will specify values for these parameters when you create the deployment or at runtime.
4. Save the changes to the workflow (click **Save** in the lower right corner).
5. Create a new deployment.
 - a. On the Targets tab, select all the target servers—both source and destination—that will participate in this database refresh. The targets that you select here will be available in the Target Parameters drop-down menus on the Run page (see [step 7](#)).
 - b. On the Parameters tab, specify values (or set the type to Runtime Value) for the required parameters listed in [step 2](#) and any additional parameters that you exposed in [step 3](#). You do not need to specify values for those parameters whose default values are appropriate for your environment.
6. Save the deployment (click **Save** in the lower right corner).

7. Run the workflow using this deployment, specifying any runtime parameters .

On the Run page, select the following targets from the respective drop-down menus:

Parameter Name	Default	Description
Source Database	no default	The database from which the backup file will be created. You specify this parameter at run time.
Target Instance	no default	The instance where the database will be restored from the backup file. You specify this parameter at run time. <div style="background-color: #f0f0f0; padding: 5px;"> <p>Note: Bridged execution workflows work on one target level (server, instance, or database). This workflow runs on the database level at all times. When choosing a target instance at run time, you will actually see a list of databases that reside on each instance. You can select any database in the target instance where you want to perform the restore.</p> <p>If you specify the RESTORE - Database Name parameter, the workflow will use that database. If you do not specify the RESTORE - Database Name parameter, the workflow will use the original database name from the backup.</p> </div>

The workflow will complete and report “Success” on the Console if it has run successfully. If an invalid parameter value is specified, an error is logged, and the workflow terminates in the “Failure” state.

Sample Scenarios

This topic shows you how to use various parameters to achieve the following database backup scenarios in your environment using the "[MS SQL - Backup and Restore Database](#)" workflow:

Scenario 1: Backup and Restore Using a Backup File that is Not Encrypted or Compressed

This is the simplest SQL Server database backup and restore scenario. In this example, the backup file is stored on a network share.

Step Name	Parameter Name	Example Value
Gather Parameters for MS SQL Database Backup and Restore	Source Database	Specified at run time.
	Target Instance	Specified at run time.
	Working Path	\\WIN-DOMAIN-CTRL\Backups

Scenario 2: Backup and Restore—Overwrite Existing Database and Preserve Existing Users

This scenario requires you to specify the two restore parameters that instruct the workflow to overwrite the existing database and then re-create existing users and roles. In this example, the backup file is stored on a network share.

Note that the BACKUP - Windows Share User and BACKUP - Windows Share Password are specified. This is not required, but it facilitates the disk space check on the network path. If you do not specify this parameter, this check is skipped.

Step Name	Parameter Name	Example Value
Gather Parameters for MS SQL Database Backup and Restore	Source Database	Selected at run time.
	Target Instance	Selected at run time.
	Working Path	\\WIN-DOMAIN-CTRL\Backups
Gather Advanced Parameters for MS SQL Database Backup and Restore	BACKUP - Windows Share User	WIN\Administrator
	BACKUP - Windows Share Password	WinSharePwd
	RESTORE - Overwrite Existing Database	YES
	RESTORE - Preserve Users and Roles	YES

Scenario 3: Perform a Backup, Simulate a Restore, and Configure Windows Domain User Using Runtime Parameters

This scenario overwrites an existing database and restores any existing users after the restore is performed. In this example, the backup file is stored on a network share.

Note: You may want to run this workflow against a MS SQL instance that can only be accessed by a Windows user with a temporary password. By using a runtime parameter for the password, you can ensure that the password used is always the latest.

To specify the Windows domain user at the time you execute a deployment with runtime parameters, perform the following additional steps:

1. When you make a copy of the workflow, expand the appropriate step, and then set the following Windows domain user parameters to - **User selected** -:
 BACKUP - Instance Account
 BACKUP - Instance Password
 RESTORE - Instance Account
 RESTORE - Instance password
2. When you create a deployment from the copy of the workflow, set the parameter types to **Runtime Value**.
3. When you execute the deployment, specify the Windows domain user account and password.

Step Name	Parameter Name	Example Value
Gather Parameters for MS SQL Database Backup and Restore	Source Database	Selected at run time.
	Target Instance	Selected at run time.
	Working Path	\\WIN-DOMAIN-CTRL\Backups
Gather Advanced Parameters for MS SQL Database Backup and Restore	BACKUP - Windows Share User	WIN\Administrator
	BACKUP - Windows Share Password	WinSharePwd Tip: To avoid having to re-enter passwords whenever they change, you can create a policy to provide them to the workflow.
	ALL - Run Simulation Only	YES
	BACKUP - Instance Account	Domain\DomainUserAcct Note: Enter at runtime.
	BACKUP - Instance	DomainUserPswd

Step Name	Parameter Name	Example Value
	Password	Note: Enter at runtime.
	RESTORE - Instance Account	Domain\DomainUserAcct Note: Enter at runtime.
	RESTORE - Instance Password	DomainUserPswd Note: Enter at runtime.

Be sure that the default values for all remaining parameters are appropriate for your environment (see Parameters for Backup and Restore MS SQL Database).

Parameters for Backup and Restore MS SQL Database

The following tables describe the required and optional input parameters for this workflow. Most of these parameters are not initially visible in a deployment. For most parameters, if you do not specify a value for a parameter, a default value is assigned

Parameters Defined in this Step: Gather Parameters for Backup and Restore MS SQL Database

Parameter Name	Default Value	Required	Description
Source Database	no default	required	The database from which the backup file will be created. You specify this parameter at run time.
Target Instance	no default	required	The instance where the database will be restored from the backup file. You specify this parameter at run time. <div style="background-color: #f0f0f0; padding: 5px;"> <p>Note: Bridged execution workflows work on one target level (server, instance, or database). This workflow runs on the database level at all times. When choosing a target instance at run time, you will actually see a list of databases that reside on each instance. You can select any database in the target instance where you want to perform the restore.</p> <p>If you specify the RESTORE - Database Name parameter, the workflow will use that database. If you do not specify the RESTORE - Database Name parameter, the workflow will use the original database name from the backup.</p> </div>
Working Path	no default	required	The directory where the database backup file will be stored. This can be a directory or a full file path. This path must be accessible to both the source and target servers. Be sure to specify this parameter in network share notation (for example: \\<network share>\). A network path can be located on a target server, but it should always be referenced using network share notation instead of local folder notation (for example: C:\<folder>). You specify this parameter in the deployment.

Additional Parameters Defined in this Step: Gather Advanced Parameters for Backup and Restore MS SQL Database

Parameter Name	Default Value	Required	Description
ALL - Encryption Password	no default	optional	Password used to encrypt and decrypt the backup file. To decrypt a backup file that was encrypted with a password, specify the password in this parameter.

Additional Parameters Defined in this Step: Gather Advanced Parameters for Backup and Restore MS SQL Database, continued

Parameter Name	Default Value	Required	Description
ALL - Run Simulation Only	No	optional	If set to YES, the workflow will only run the Pre-Restore Validation step. It will not attempt to restore the database. Use this mode to discover the original data and log files used for the database backup. Valid values: YES or NO.
BACKUP - Backup Description	no default	optional	Text that describes this backup (up to 255 characters).
BACKUP - Backup Name	no default	optional	The name of this backup (up to 128 characters).
BACKUP - Compress Backup File	NO	optional	If you specify YES, the backup file will be compressed. Valid values: YES or NO. Compression is supported on SQL Server 2008 Enterprise and later.
BACKUP - Expiration Date	no default	optional	Date and time when the backup file expires and the backup data is no longer considered relevant. After this date and time, SQL Server is not prevented from overwriting this backup file. The Expiration Date must be specified in a format compatible with the configured system datetime format. If both the Retention Days and the Expiration Date parameters are specified, the Retention Days parameter takes precedence.
BACKUP - Instance Account	no default	optional	The Windows account that will perform the backup operation.
BACKUP - Instance Password	no default	optional	The password for the Windows account that will perform the backup operation.
BACKUP - Perform Integrity Check	NO	optional	If you specify YES, the

Additional Parameters Defined in this Step: Gather Advanced Parameters for Backup and Restore MS SQL Database, continued

Parameter Name	Default Value	Required	Description
			workflow will perform an integrity check on the database backup file. Valid values: YES or NO.
BACKUP - Retention Days	no default	optional	Number of days that must elapse before this backup file can be overwritten by SQL Server. If both the Retention Days and the Expiration Date parameters are specified, the Retention Days parameter takes precedence.
BACKUP - Windows Share Password	no default	optional	Password for the user specified in Windows Share User.
BACKUP - Windows Share User	no default	optional	Windows user who can access the specified Windows network share and who will own (and write) the backup file.
RESTORE - Data File Locations	no default	optional	Comma-delimited list of directories or full file paths for each data file in the backup file. Use Run Simulation Only to discover the number of data files in backup file. If this parameter is not specified, the original data file name will be used.
RESTORE - Database Name	no default	optional	To restore the database from the backup file using a different database name, specify that name here. If this parameter is not specified, the original database name will be used.
RESTORE - Download Target Destination	no default	optional	The directory where the database backup file will be stored.
RESTORE - Instance Account	no default	optional	The Windows account that will perform the restore operation.
RESTORE - Instance Password	no	optional	The password for the Windows

Additional Parameters Defined in this Step: Gather Advanced Parameters for Backup and Restore MS SQL Database, continued

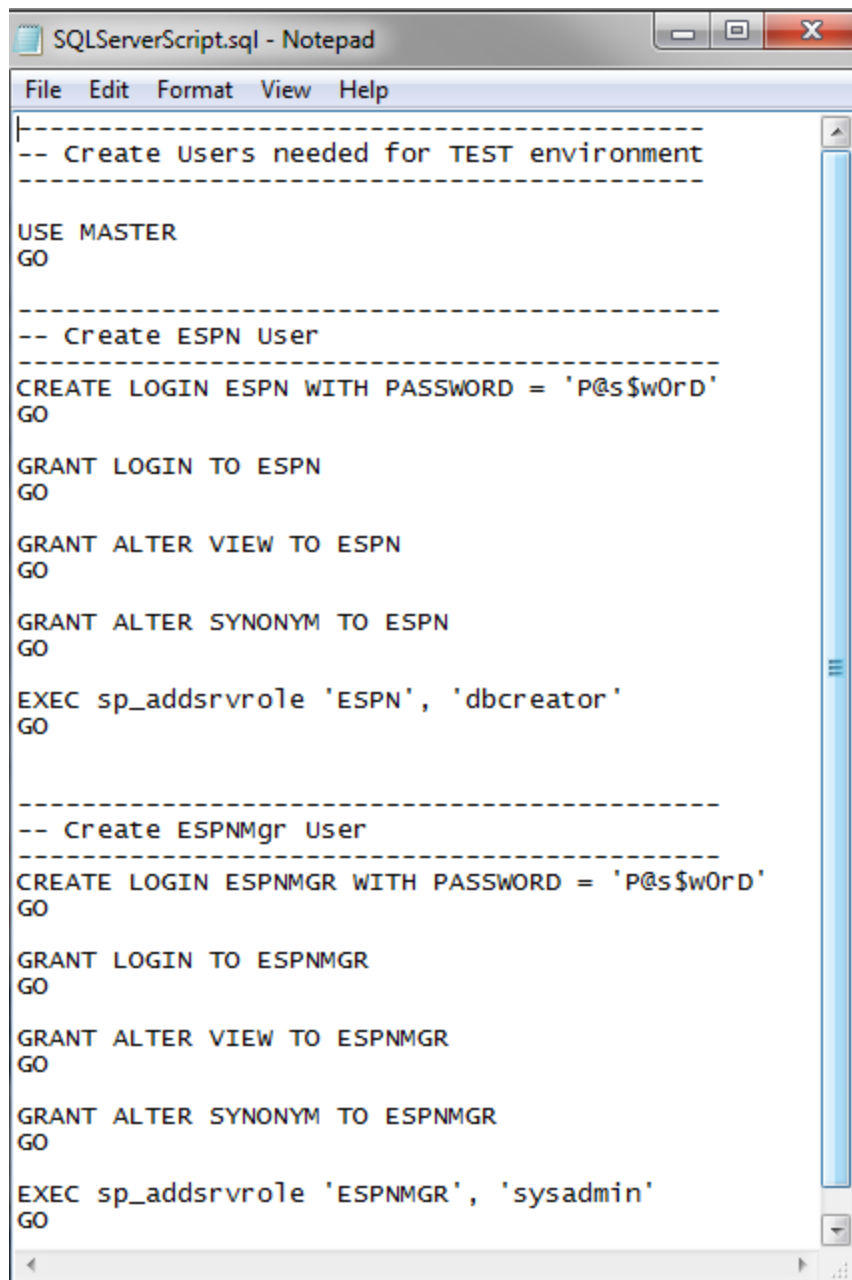
Parameter Name	Default Value	Required	Description
	default		account that will perform the restore operation.
RESTORE - Log File Locations	no default	optional	Comma-delimited list of directories or full file paths for each log file in the backup file. Use Run Simulation Only mode to discover the number of log files in backup file. If this parameter is not specified, the original log file name will be used.
RESTORE - Overwrite Existing Database	NO	optional	If set to YES, and the database already exists, the workflow will overwrite the database. Valid values: YES or NO.
RESTORE - Preserve Users and Roles	NO	optional	If set to YES, and the database already exists, the workflow will overwrite the database. Valid values: YES or NO.
RESTORE - Reindex Restored Database	NO	optional	<p>If set to YES, the workflow will re-index the database after the restore operation is successfully completed. Valid values: YES or NO.</p> <p>Re-indexing improves database performance. More specifically , it recreates all the table look-ups and performance tunes them according to the new environment. This is important when you are restoring a database in a new environment that it has never seen before.</p>

DB Release for SQL Server

This workflow will check a list of T-SQL script files for disallowed commands, check the syntax, then execute the files on the target Microsoft SQL Server databases if they pass all required tests.

This workflow is designed for SQL script transactions to be deployed and executed against target SQL Server databases. SQL scripts are stored and downloaded from the HPE DMA software repository.

Before running the DB Release for SQL Server workflow you need to create the SQL script file (or files). For example:



```
SQLServerScript.sql - Notepad
File Edit Format View Help
-----
-- Create Users needed for TEST environment
-----

USE MASTER
GO

-----
-- Create ESPN User
-----

CREATE LOGIN ESPN WITH PASSWORD = 'P@s$w0rD'
GO

GRANT LOGIN TO ESPN
GO

GRANT ALTER VIEW TO ESPN
GO

GRANT ALTER SYNONYM TO ESPN
GO

EXEC sp_addsrvrole 'ESPN', 'dbcreator'
GO

-----
-- Create ESPNMGR User
-----

CREATE LOGIN ESPNMGR WITH PASSWORD = 'P@s$w0rD'
GO

GRANT LOGIN TO ESPNMGR
GO

GRANT ALTER VIEW TO ESPNMGR
GO

GRANT ALTER SYNONYM TO ESPNMGR
GO

EXEC sp_addsrvrole 'ESPNMGR', 'sysadmin'
GO
```

You can customize what the workflow checks in the SQL scripts:

- SQL advanced features
- SQL database commands
- SQL database links
- SQL syntax
- SQL system grants
- A regular expression

If all the tests pass, the SQL scripts may be deployed and executed against the target SQL Server databases.

Note: This workflow does not provide any rollback capability.

To use this workflow in your environment, see the following information:

Topic	Information Included
"Prerequisites for this Workflow"	List of prerequisites that must be satisfied before you can run this workflow
"How this Workflow Works"	Information about what the workflow does, including validation checks performed, and steps executed
"How to Run this Workflow"	Instructions for running this workflow in your environment
"Sample Scenarios"	Examples of typical parameter values for this workflow
Parameters	List of input parameters for this workflow

Prerequisites for this Workflow

Be sure that the following prerequisites are satisfied before you run the ["DB Release for SQL Server"](#) workflow.

Dependencies

- The latest HPE DMA solution packs require the latest HPE DMA platform. To use the latest solution packs, update the HPE DMA platform. HPE DMA10.40 solution packs are supported on HPE DMA10.40 (and later).
- You have installed the Database Compliance solution pack.
- An SQL Server instance and its databases should already be provisioned and added to the Environment section—this can be accomplished by using Discovery.
- The SQL script must be in the HPE DMA software repository.
- You have installed the `osql` or `SQLCMD` utility and made it accessible via the user/password settings stored in the metadata. Check the Environment page for those settings. If there is no metadata, the connection will use Windows authentication.
- You need an SA (System Administrator) role to perform any server level or database level updates.

Supported Versions of SQL Server

2008, 2008 R2, 2012, 2014

SQL Scripts

You need to create the SQL script file (or files) that manage the release. The files may contain the normal SQL Server DML and DDL commands.

Tip: List the SQL script files in the SQL scripts parameter in the order in which they need to be executed.

SQL Server Documentation

For more information about prerequisites for SQL Server, refer to the [Microsoft SQL Server Documentation](#).

How this Workflow Works

The following information describes how the "DB Release for SQL Server" workflow works.

Overview show

The workflow starts by gathering input parameters.

If the SQL scripts do not exist on the specified target location, they are downloaded from the software repository.

Based on the parameters you set when you create your deployment, the workflow will do the following:

- Check the SQL code for SQL advanced features—unless specified in the exception list. If any are found, the workflow will exit with a failure code.
- Check the SQL code for SQL database commands—unless specified in SQL commands to be excluded from the check. If any are found, the workflow will exit with a failure code.
- Check the SQL code for any SQL database links—if any are found, the workflow will exit with a failure code.
- Check the SQL code for syntax errors—if any are found, the workflow will exit with a failure code.
- Check the SQL code for any SQL system grants—unless specified in the exception list. If any are found, the workflow will exit with a failure code.
- Check the SQL code for a regular expression that you specify—if any matches are found, the workflow will exit with a failure code.

If there were no errors in the checks and the Run Flag is set, the workflow uses the `osql` or `SQLCMD` utility to execute the SQL script files.

Validation Checks Performed show

This workflow validates the SQL scripts in the following ways:

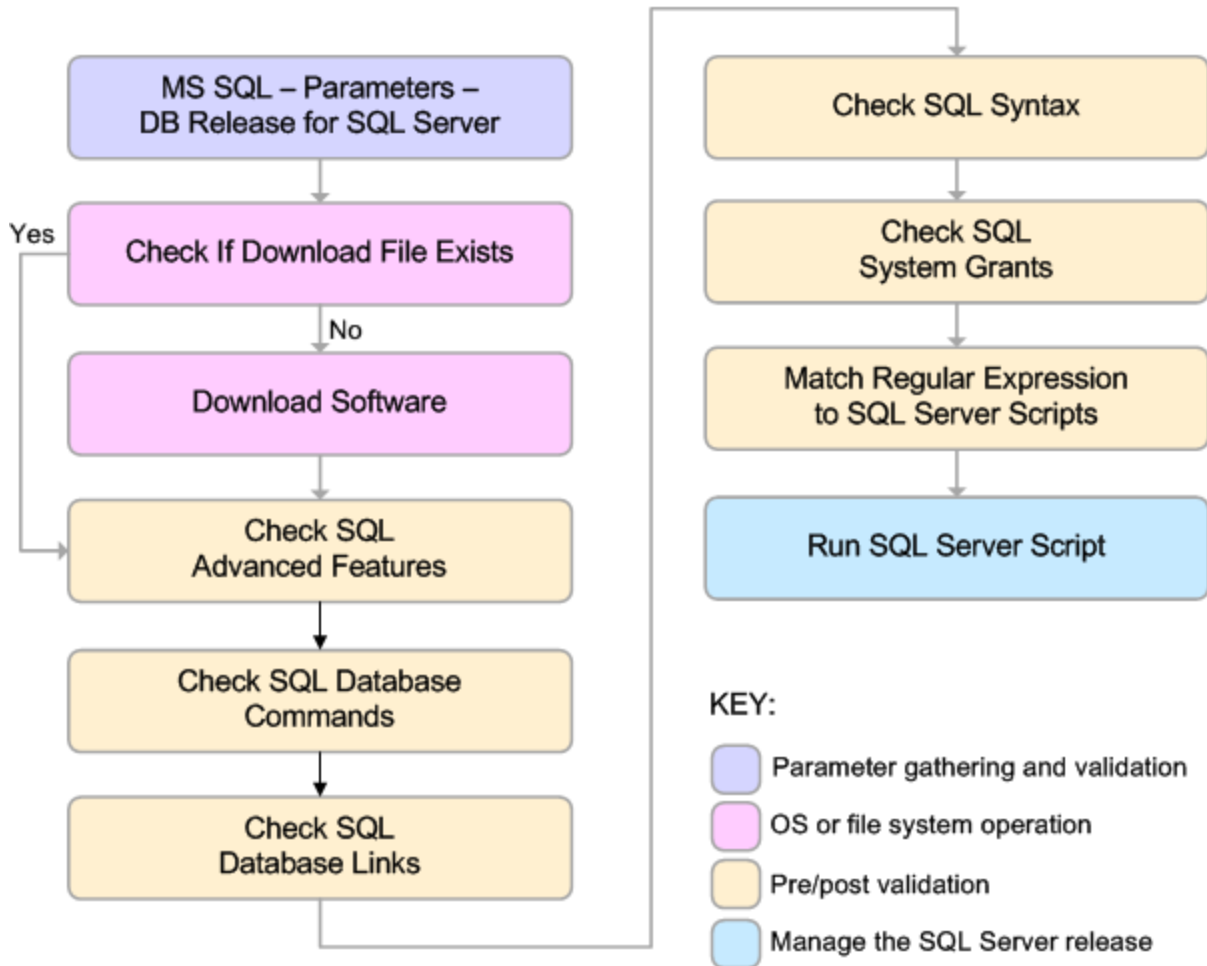
1. If you set the Run Flag to Check SQL Advanced Features, the workflow searches for any instance configuration options—unless included in your exclusion list. These are instance level settings that most users shouldn't be changing, for example, startup procs and `xp_cmdshell`.
2. If you set the Run Flag to Check SQL Database Commands, the workflow searches the SQL statements for the commands that you specify in SQL Commands.
3. If you set the Run Flag to Check SQL Database Links, the workflow searches the SQL statements for `OPENQUERY`, `OPENROWSET`, and `OPENDATASOURCE` statements. It also checks for this pattern: `[server].[instance].[owner].[database]`
4. If you set the Run Flag to Check SQL Syntax, the workflow verifies that all the SQL statements have valid syntax.
5. If you set the Run Flag to Check SQL System Grants, the workflow searches the SQL statements for any system level (server role) grants—unless included in your exclusion list. For example: `GRANT CONTROL SERVER TO SOMEUSER`
6. If you set the Run Flag to Match Regular Expression to SQL Server Scripts and you specify a regular expression, the workflow searches the SQL statements for any regex matches.

If any of the validations fail, the workflow will output the offending SQL line to `stdout`, return an error status, and the SQL scripts will not be executed.

Steps Executed show

The "DB Release for SQL Server" workflow includes the following steps. Each step must complete successfully before the next step can start. If a step fails, the workflow reports a failure and all subsequent steps are skipped.

Click each box in the diagram to view additional information about that step.



Steps Used in DB Release for SQL Server

Workflow Step	Description
MS SQL - Parameters - DB Release for SQL Server	This step accepts the basic input parameters for the workflow. The parameters will be used in subsequent steps.
Check if Download File Exists	This step determines whether one or more specified files already exist on the target server.
Download Software	This step downloads a list of files to a specified location on the target server.
Check SQL Advanced Features	This step checks the SQL scripts for any advanced feature non-default setting. An exception list can be specified to exclude specific advanced features from the check.

Steps Used in DB Release for SQL Server, continued

Workflow Step	Description
Check SQL Database Commands	This step checks the SQL scripts to ensure that specific types of SQL database commands—as specified in the SQL Commands parameter—are not included.
Check SQL Database Links	This step checks an SQL Script for any database link usage.
Check SQL Syntax	This step verifies the syntax of an SQL Server Script. The step assumes that a <code>go</code> statement on its own line signifies the end of a code block.
Check SQL System Grants	This step checks an SQL Script for any system level (server role) grants. An exception list can be specified to exclude specific privileges from the check.
Match Regular Expression to SQL Server Scripts	This step applies a regular expression to each SQL statement in an SQL Script file. If any regex matches are found, they are output to <code>stdout</code> and an error is returned.
Run SQL Server Script	This step executes SQL Scripts using <code>osql</code> or <code>SQLCMD</code> . This step is only executed if all the previous checks passed.

Note: For input parameter descriptions and defaults, see "[Parameters for DB Release for SQL Server](#)" on page 92.

How to Run this Workflow

The following instructions show you how to customize and run the ["DB Release for SQL Server"](#) workflow in your environment.

The workflow provides default values for some parameters. These default values are usually sufficient for a "typical" installation. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios. Any parameters not explicitly specified in the deployment will have the default values listed in ["Parameters for DB Release for SQL Server" on page 92](#).

Note: Before following this procedure, review the ["Prerequisites for this Workflow" on page 81](#), and ensure that all requirements are satisfied.

To use the DB Release for SQL Server workflow:

1. Create a deployable copy of the workflow.
2. Determine the values that you will specify for the following parameters:

Input Parameters for MS SQL - Parameters - DB Release for SQL Server

Parameter Name	Default Value	Required	Description
Display SQL Length	2000	optional	The number of characters of a SQL batch that is displayed when an error occurs. Enter "0" to display the entire code. Note: Displaying the entire code may cause performance issues for your browser.
File List	no default	required	Comma-separated list of the files that contain the SQL scripts that will be checked. Note: List the SQL script files in the order in which they need to be executed.
Staging Directory	C:\Temp\	optional	The directory that contains the SQL scripts that will be checked.

Input Parameters for Check SQL Advanced Features

Parameter Name	Default Value	Required	Description
Exception List	see description	optional	Comma-separated list of advanced features that will be allowed. For example, if you specify CURSOR THRESHOLD, QUERY WAIT, those advanced features will be allowed—any other advanced features that occur in the code will cause the workflow to fail. The default is to check all of the normal

Input Parameters for Check SQL Advanced Features, continued

Parameter Name	Default Value	Required	Description
			advanced features.
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Input Parameters for Check SQL Database Commands

Parameter Name	Default Value	Required	Description
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).
SQL Commands	shutdown, sp_configure, create database, drop database, create login, create user, drop login, drop user, sp_grantdbaccess, sp_addlogin, sp_droplogin	optional	Comma-separated list of SQL commands that are not allowed. The default shows an example of how to fill out the list. You may want to customize this list for your configuration.

Input Parameters for Check SQL Database Links

Parameter Name	Default Value	Required	Description
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Input Parameters for Check SQL Syntax

Parameter Name	Default Value	Required	Description
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Input Parameters for Check SQL System Grants

Parameter Name	Default Value	Required	Description
Exception List	grant db_owner, grant ddladmin, grant sysadmin, grant securityadmin, grant serveradmin, grant processadmin, grant diskadmin, grant dbcreator	optional	Comma-separated list of SQL system privileges that will be allowed. For example, if you specify , those system privileges will be allowed—any other system privileges that occur in the code will cause the workflow to fail. The default shows an example of how to fill out the list. You may want to customize this list for your configuration.
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Input Parameters for Match Regular Expression to SQL Server Scripts

Parameter Name	Default Value	Required	Description
Regular Expression		optional	The regular expression to be searched for in all of the SQL scripts to be deployed. If the specified regular expression is found, the workflow exits with a failure. For example: drop\s+table will match all statements that drop a table.
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Input Parameters for Run SQL Server Script

Parameter Name	Default Value	Required	Description
Database Name	master	optional	The name of the database to which the specified SQL scripts will be applied.
Run Flag	Y	optional	Flag to indicate whether the workflow should run the SQL Server

Input Parameters for Run SQL Server Script, continued

Parameter Name	Default Value	Required	Description
			script. Valid values are Y (run the check) or N (do not run the check).

Note: See "[Parameters for DB Release for SQL Server](#)" on page 92 for detailed descriptions of all input parameters for this workflow, including default values.

3. In the workflow editor, expose any additional parameters that you need. You will specify values for those parameters when you create the deployment or at runtime.
4. Save the changes to the workflow (click **Save** in the lower right corner).
5. Create a new deployment.
6. On the Parameters tab, specify values (or set the type to Runtime Value) for the required parameters listed in step 2 and any additional parameters that you have exposed. You do not need to specify values for those parameters whose default values are appropriate for your environment.
7. On the Targets tab, specify one or more targets for this deployment.
8. Save the deployment (click **Save** in the lower right corner).
9. Run the workflow using this deployment, specifying any runtime parameters.

To verify the results:

The workflow will complete and report SUCCESS on the Console if it has run successfully. If an error occurs during workflow execution, the error is logged, and the workflow terminates in the FAILURE state.

Optional: If you want to further verify the results:

Log in to your database to make sure that whatever you created or modified was actually done.

To view the output:

The workflow writes the execution output for SQL script execution in the HPE DMA Steplog.

Sample Scenarios

This topic shows you typical parameter values for different use cases for the ["DB Release for SQL Server"](#) workflow.

Scenario 1: Check the SQL script files for disallowed commands, check the syntax, then deploy and execute the scripts show

You only need to specify the File List and the Staging Directory since this scenario takes advantage of many parameter defaults. The workflow will check the SQL script files for:

- All of the normal advanced features
- All of the SQL database commands that are in the default SQL Commands parameter
- SQL database links
- SQL syntax
- All the SQL system grants—except those in the default Exception List parameter
- No regular expression—since none is specified by default

As long as no error is discovered in the checks, the SQL scripts will be deployed and executed on the target SQL Server databases.

Determine the values that you will specify for the following parameters:

Input Parameters for MS SQL - Parameters - DB Release for SQL Server

Parameter Name	Example Value	Description
File List	sqlserverscript.sql	Comma-separated list of the files that contain the SQL scripts that will be checked. Note: List the SQL script files in the order in which they need to be executed.
Staging Directory	C:\Temp\	The directory that contains the SQL scripts that will be checked.

Be sure that the default values for all remaining input parameters are appropriate for your environment (see ["Parameters for DB Release for SQL Server"](#) on page 92).

Scenario 2: Check the SQL script files for disallowed commands, check the syntax, configure Windows domain user using runtime parameters, but do not deploy and execute the scripts show

This scenario takes advantage of many parameter defaults and also demonstrates some optional parameters. The workflow will check the SQL script files for:

- All of the SQL database commands that are in the default SQL Commands parameter
- SQL database links
- SQL syntax
- All the SQL system grants—except those in the default Exception List parameter
- The regular expression drops+table
- A database to which the SQL scripts will be applied

Note: You may want to run this workflow against a MS SQL instance that can only be accessed by a Windows user with a temporary password. By using a runtime parameter for the password, you can ensure that the password used is always the latest.

To specify the Windows domain user at the time you execute a deployment with runtime parameters, perform the following additional steps:

1. When you make a copy of the workflow, expand the appropriate step, and then set the Windows domain user parameters—Instance Account and Instance Password—to **- User selected -**.
2. When you create a deployment from the copy of the workflow, set the parameter types to **Runtime Value**.
3. When you execute the deployment, specify the Windows domain user account and password.

This workflow run will only report the results of the checks. The SQL scripts will NOT be deployed and executed on the target SQL Server databases.

Determine the values that you will specify for the following parameters:

Input Parameters for MS SQL - Parameters - DB Release for SQL Server

Parameter Name	Example Value	Description
File List	sqlserverscript.sql	Comma-separated list of the files that contain the SQL scripts that will be checked. Note: List the SQL script files in the order in which they need to be executed.
Instance Account	Domain\DomainUserAcct Note: Enter at runtime.	The Windows account that will perform the release management.
Instance Password	DomainUserPswd Note: Enter at runtime.	The password for the Windows account that will perform the release management.

Input Parameters for MS SQL - Parameters - DB Release for SQL Server, continued

Parameter Name	Example Value	Description
Staging Directory	C:\Temp\	The directory that contains the SQL scripts that will be checked.

Input Parameters for Match Regular Expression to SQL Server Scripts

Parameter Name	Example Value	Description
Regular Expression	drop\s+table	The regular expression to be searched for in all of the SQL scripts to be deployed. If the specified regular expression is found, the workflow exits with a failure. For example: drop\s+table will match all statements that drop a table.
Run Flag	Y	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Input Parameters for Run SQL Server Script

Parameter Name	Example Value	Description
Database Name	mydb	The name of the database to which the specified SQL scripts will be applied.
Run Flag	N	Flag to indicate whether the workflow should run the SQL Server script. Valid values are Y (run the check) or N (do not run the check).

Note: Some of these parameters are not exposed by default in the deployment.

Be sure that the default values for all remaining input parameters are appropriate for your environment (see ["Parameters for DB Release for SQL Server" on the next page](#)).

Parameters for DB Release for SQL Server

The following tables describe the required and optional input parameters for this workflow. Some of these parameters may not be initially visible in a deployment. For some parameters, if you do not specify a value for a parameter, a default value is assigned.

Note: Only those parameters that are configurable in a standard deployment are listed here. Input parameters that must be mapped to output parameters of previous steps are not listed.

Input Parameters Defined in this Step: MS SQL - Parameters - DB Release for SQL Server

Parameter Name	Default Value	Required	Description
Display SQL Length	2000	optional	The number of characters of a SQL batch that is displayed when an error occurs. Enter "0" to display the entire code. Note: Displaying the entire code may cause performance issues for your browser.
File List	no default	required	Comma-separated list of the files that contain the SQL scripts that will be checked. Note: List the SQL script files in the order in which they need to be executed.
Instance Account	no default	optional	The Windows account that will perform the release management.
Instance Password	no default	optional	The password for the Windows account that will perform the release management.
Staging Directory	C:\Temp\	optional	The directory that contains the SQL scripts that will be checked.

Additional Input Parameters Defined in this Step: Check SQL Advanced Features

Parameter Name	Default Value	Required	Description
Exception List	see description	optional	Comma-separated list of advanced features that will be allowed. For example, if you specify CURSOR THRESHOLD, QUERY WAIT, those advanced features will be allowed—any other advanced features that occur in the code will cause the workflow to fail. The default is to check all of the normal advanced features.
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Additional Input Parameters Defined in this Step: Check SQL Database Commands

Parameter Name	Default Value	Required	Description
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).
SQL Commands	shutdown, sp_configure, create database, drop database, create login, create user, drop login, drop user, sp_grantdbaccess, sp_addlogin, sp_droplogin	optional	Comma-separated list of SQL commands that are not allowed. The default shows an example of how to fill out the list. You may want to customize this list for your configuration.

Additional Input Parameters Defined in this Step: Check SQL Database Links

Parameter Name	Default Value	Required	Description
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Additional Input Parameters Defined in this Step: Check SQL Syntax

Parameter Name	Default Value	Required	Description
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Additional Input Parameters Defined in this Step: Check SQL System Grants

Parameter Name	Default Value	Required	Description
Exception List	grant db_owner, grant ddladmin, grant sysadmin, grant securityadmin, grant serveradmin, grant processadmin, grant diskadmin, grant dbcreator	optional	Comma-separated list of SQL system privileges that will be allowed. For example, if you specify , those system privileges will be allowed—any other system privileges that occur in the code will cause the workflow to fail. The default shows an example of how to fill out the list. You may want to customize this list for your

Additional Input Parameters Defined in this Step: Check SQL System Grants, continued

Parameter Name	Default Value	Required	Description
			configuration.
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Additional Input Parameters Defined in this Step: Match Regular Expression to SQL Server Scripts

Parameter Name	Default Value	Required	Description
Regular Expression		optional	The regular expression to be searched for in all of the SQL scripts to be deployed. If the specified regular expression is found, the workflow exits with a failure. For example: drop\s+table will match all statements that drop a table.
Run Flag	Y	optional	Flag to indicate whether the workflow should run this check. Valid values are Y (run the check) or N (do not run the check).

Additional Input Parameters Defined in this Step: Run SQL Server Script

Parameter Name	Default Value	Required	Description
Database Name	master	optional	The name of the database to which the specified SQL scripts will be applied.
Run Flag	Y	optional	Flag to indicate whether the workflow should run the SQL Server script. Valid values are Y (run the check) or N (do not run the check).

MS SQL Drop Database

This workflow is supported on the Windows operating system platform. The MS SQL Drop Database enables you to remove the target database from the MS SQL instance and from the DMA environment..

To use this workflow in your environment, see the following information:

Topic	Information Included
"Prerequisites for this Workflow" on the next page	List of prerequisites that must be satisfied before you can run this workflow
"How this Workflow Works" on page 97	Information about what the workflow does, including validation checks performed, steps executed, and step descriptions
"How to Run this Workflow" on page 98	Instructions for running this workflow in your environment
"Parameters for MS SQL - Drop Database" on page 99	List of input parameters for this workflow

Prerequisites for this Workflow

Be sure that the following prerequisites are satisfied before you run the MS SQL Drop Database workflow:

- This solution requires HPE DMA version 10.30 (or later).
The latest HPE DMA solution packs require the latest HPE DMA platform. To use the latest solution packs, update the HPE DMA platform. HPE DMA10.40 solution packs are supported on HPE DMA10.40 (and later).
- You have installed the Database Compliance solution pack.

The workflow must be able to:

- MS SQL service should be up and running.
- Log in to the MS SQL instance using MS SQL login credentials.
- It should drop the database upon connecting to the MS SQL instance.

The information presented here assumes the following: show assumptions

- HPE DMA is installed and operational.
- At least one suitable target server is available.
- You are logged in to the HPE DMA web interface.
- You have permission to create, edit, and deploy copies of the workflows included in this solution pack.

For more information about prerequisites for MS SQL database, refer to the [MS SQL Server Documentation](#).

How this Workflow Works

This workflow performs the following actions:

Drops a MS SQL database and removes it from the DMA environment.

Steps Executed by the Workflow

The MS SQL Drop Database workflow includes the following steps. Each step must complete successfully before the next step can start. If a step fails, the workflow reports a failure and all subsequent steps are skipped.

Steps Used by MS SQL

Workflow Step	Description
Gather Advanced Parameters for MS SQL Drop Database	This step gathers the parameters required to drop an MS SQL database.
MS SQL Check Database Exists	This step validates the existence of the database. Access to the master database is required for validation.
MS SQL Kill Processes	This step kills all the currently running user processes on the target database.
MS SQL Drop Database	This step drops the target database. To run this step, ensure that there are no active connections prior to running this step by running the "MS SQL: Kill Processes" step.
MS SQL Check Database Exists	This step validates the existence of a database. Access to the master database is required for validation.
Remove Database from Environment V2	This step removes the database from the DMA environment. This step takes the Instance Name and Database Name as input parameters. If the Instance Name and Database Name are not provided as input parameters, then the database against which the workflow is being executed will be removed from the DMA environment.

Note: For input parameter descriptions and defaults, see ["Parameters for MS SQL - Drop Database" on page 99](#).

How to Run this Workflow

The following instructions show you how to customize and run the MS SQL Drop Database workflow in your environment.

The workflow provides default values for some parameters. These default values are usually sufficient for a "typical" installation. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios. Any parameters not explicitly specified in the deployment will have the default values listed in ["Parameters for MS SQL - Drop Database" on the next page](#).

Note: Before following this procedure, review the ["Prerequisites for this Workflow" on page 96](#), and ensure that all requirements are satisfied.

To use the Run MySQL Drop Database workflow:

1. Create a deployable copy of the workflow.
2. Determine the values that you will specify for the following parameters:

Note: There are no mandatory parameters required to run this workflow. All parameters are optional. You may need to expose additional parameters depending on your objectives.

See ["Parameters for MS SQL - Drop Database" on the next page](#) for detailed descriptions of all input parameters for this workflow, including default values.

3. In the workflow editor, expose any additional parameters that you need. You will specify values for those parameters when you create the deployment or at runtime.
4. Save the changes to the workflow (click **Save** in the lower right corner).
5. Create a new deployment.
6. On the Parameters tab, specify values (or set the type to Runtime Value) for the required parameters listed in step 2 and any additional parameters that you have exposed. You do not need to specify values for those parameters whose default values are appropriate for your environment.
7. On the Targets tab, specify one or more targets for this deployment.
8. Save the deployment (click **Save** in the lower right corner).
9. Run the workflow using this deployment, specifying any runtime parameters.

To verify the results:

The workflow will complete and report SUCCESS on the Console if it has run successfully. If an error occurs during workflow execution, the error is logged, and the workflow terminates in the FAILURE state. Also verify by checking that the target database no longer appears in the DMA Environment section.

Parameters for MS SQL - Drop Database

There are no mandatory parameters required to run this workflow. All parameters are optional. Some of these parameters are not initially visible in a deployment. For most parameters, if you do not specify a value for a parameter, a default value is assigned.

MS SQL - Upgrade Standalone SQL Instance

This workflow is supported on the Windows operating system platform. The MS SQL - Upgrade Standalone SQL Instance enables you to update and existing instance of SQL Server.

To use this workflow in your environment, see the following information:

Topic	Information Included
"Prerequisites for this Workflow" on the next page	List of prerequisites that must be satisfied before you can run this workflow
"How this Workflow Works" on page 101	Information about what the workflow does, including validation checks performed, steps executed, and step descriptions
"How to Run this Workflow" on page 103	Instructions for running this workflow in your environment
"Parameters for MS SQL - Upgrade Standalone SQL Instance" on page 104	List of input parameters for this workflow

Prerequisites for this Workflow

Be sure that the following prerequisites are satisfied before you run the MS SQL - Upgrade Standalone SQL Instance workflow:

- This solution requires HPE DMA version 10.30 (or later).
The latest HPE DMA solution packs require the latest HPE DMA platform. To use the latest solution packs, update the HPE DMA platform. HPE DMA10.40 solution packs are supported on HPE DMA10.40 (and later).
- You have installed the Database Compliance solution pack.

The workflow must be able to:

- MS SQL service should be up and running.
- Log in to the MS SQL instance using MS SQL login credentials.
- It should drop the database upon connecting to the MS SQL instance.

The information presented here assumes the following: show assumptions

- HPE DMA is installed and operational.
- At least one suitable target server is available.
- You are logged in to the HPE DMA web interface.
- You have permission to create, edit, and deploy copies of the workflows included in this solution pack.

For more information about prerequisites for MS SQL database, refer to the [MS SQL Server Documentation](#).

How this Workflow Works

This workflow performs the following actions:

Upgrades an existing standalone instance of SQL Server 2005/08/08R2/12 to SQL Server 2008/08R2/12/14 on an existing Windows 2008/08R2/12/12 R2 server.

Steps Executed by the Workflow

The MS SQL - Upgrade Standalone SQL Instance workflow includes the following steps. Each step must complete successfully before the next step can start. If a step fails, the workflow reports a failure and all subsequent steps are skipped.

Steps Used by MS SQL

Workflow Step	Description
MS SQL - Parameters - Upgrade Standalone	This step gathers all the required parameters for a standalone SQL Server upgrade.
MS SQL - Advanced Parameters - Upgrade Standalone	This step gathers all the optional parameters for a standalone SQL Server upgrade. All advanced parameters are hidden in the deployment screen by default. In order to activate an advanced parameter, go into the Workflow, and change the parameter mapping from on this step from Blank to User Input.
Check If Download File Exists	This step is designed to facilitate the complicated methodologies that various companies use to distribute their software bundles for installation.
MS SQL - Create Install or Upgrade Template	This step verifies that all required parameters are provided, and writes any optional parameters to the template file if they are non-blank.
Unzip Archive	This step is to unzip a zip archive, verify if the input file exists, ensure the output directory exists, creates required directories, and deploys archived files.
MS SQL - Simulate - Install or Upgrade	This step verifies that all required parameters are provided, and the system meets minimum requirements.
MS SQL - Install or Upgrade	This step installs SQL Server 2008 by running the setup.exe program located on the installation media.
MS SQL Verify SQL Installation	This step determines if the target instance name of SQL Server is currently installed.
Windows	This step checks for any pending reboots.

Steps Used by MS SQL , continued

Workflow Step	Description
Check for Pending Reboot	
Discover SQL Databases	This step audits the server's physical environment looking for SQLServer instances and databases.
Windows Restart Server	This step restarts a system.
Windows Wait for Restart	This step is to wait 8 minutes for Windows server to finish restart.
MS SQL - Install or Upgrade	This installs SL Server 2008 by running the setup.exe program located on the installation media.
MS SQL Verify SQL Installation	This step determines if the target instance name of SQL Server is currently installed.

Note: For input parameter descriptions and defaults, see ["Parameters for MS SQL - Upgrade Standalone SQL Instance" on page 104.](#)

How to Run this Workflow

The following instructions show you how to customize and run the MS SQL Upgrade Standalone SQL Instance workflow in your environment.

The workflow provides default values for some parameters. These default values are usually sufficient for a "typical" installation. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios. Any parameters not explicitly specified in the deployment will have the default values listed in ["Parameters for MS SQL - Upgrade Standalone SQL Instance" on the next page](#).

Note: Before following this procedure, review the ["Prerequisites for this Workflow" on page 100](#), and ensure that all requirements are satisfied.

To use the Run MS SQL - Upgrade Standalone SQL Instance workflow:

1. Create a deployable copy of the workflow.
2. Determine the values that you will specify for the following parameters:

Note: There are no mandatory parameters required to run this workflow. All parameters are optional. You may need to expose additional parameters depending on your objectives.

See ["Parameters for MS SQL - Upgrade Standalone SQL Instance" on the next page](#) for detailed descriptions of all input parameters for this workflow, including default values.

3. In the workflow editor, expose any additional parameters that you need. You will specify values for those parameters when you create the deployment or at runtime.
4. Save the changes to the workflow (click **Save** in the lower right corner).
5. Create a new deployment.
6. On the Parameters tab, specify values (or set the type to Runtime Value) for the required parameters listed in step 2 and any additional parameters that you have exposed. You do not need to specify values for those parameters whose default values are appropriate for your environment.
7. On the Targets tab, specify one or more targets for this deployment.
8. Save the deployment (click **Save** in the lower right corner).
9. Run the workflow using this deployment, specifying any runtime parameters.

To verify the results:

The workflow will complete and report SUCCESS on the Console if it has run successfully. If an error occurs during workflow execution, the error is logged, and the workflow terminates in the FAILURE state. Also verify by checking that the target database no longer appears in the DMA Environment section.

Parameters for MS SQL - Upgrade Standalone SQL Instance

The following tables describe the required and optional input parameters for this workflow. Some of these parameters may not be initially visible in a deployment. For some parameters, if you do not specify a value for a parameter, a default value is assigned.

Note: Only those parameters that are configurable in a standard deployment are listed here. Input parameters that must be mapped to output parameters of previous steps are not listed.

Parameters Defined in this Step: MS SQL - Parameters - Upgrade Standalone

Parameter Name	Default Value	Required	Description
Download From Software Directory	no default	optional	The name of the ZIP file containing the SQL Server setup files
Download Target Destination	no default	required	The local directory where the SQL Setup files should be stored.
Instance Name	MSSQLSERVER	required	The name of the newly created instance. Use MSSQLSERVER for the default instance, any other alphanumeric value for a named instance.

Additional Parameters Defined in this Step: MS SQL - Advanced Parameters - Upgrade Standalone

Parameter Name	Default Value	Required	Description
Additional Template Parameters	no default	optional	Pipe-delimited (") list of additional template parameters and values. SQMREPORTING 1 INSTANCEDIR "D:\SQLDirectory
Installation Path	no default	optional	Specifies the location for the SQL Server program files.
Installer Account	no default	optional	The Windows account that will be performing the installation.
Installer Password	no default	optional	The password of the Windows account that will be performing the installation.
Product Key	no default	optional	Specifies the product key for the edition of SQL Server. If this parameter is not specified, Evaluation is used.
Skip Simulation	no default	optional	If set to "YES", workflow will skip Simulate step and proceed directly to install/upgrade step.

MS SQL Rollback Patch

This workflow is supported on the Windows operating system platform. The MS SQL Rollback Patch enables you to uninstall a SQL Server patch on a standalone 2005/2008/2008R2 instance.

To use this workflow in your environment, see the following information:

Topic	Information Included
"Prerequisites for this Workflow" on the next page	List of prerequisites that must be satisfied before you can run this workflow
"How this Workflow Works" on page 107	Information about what the workflow does, including validation checks performed, steps executed, and step descriptions
"How to Run this Workflow" on page 109	Instructions for running this workflow in your environment
"Parameters for MS SQL Rollback Patch" on page 110	List of input parameters for this workflow

Prerequisites for this Workflow

Be sure that the following prerequisites are satisfied before you run the MS SQL Rollback Patch workflow:

- This solution requires HPE DMA version 10.30 (or later).
The latest HPE DMA solution packs require the latest HPE DMA platform. To use the latest solution packs, update the HPE DMA platform. HPE DMA10.40 solution packs are supported on HPE DMA10.40 (and later).
- You have installed the Database Compliance solution pack.

The workflow must be able to:

- MS SQL service should be up and running.
- Log in to the MS SQL instance using MS SQL login credentials.
- It should drop the database upon connecting to the MS SQL instance.

The information presented here assumes the following: show assumptions

- HPE DMA is installed and operational.
- At least one suitable target server is available.
- You are logged in to the HPE DMA web interface.
- You have permission to create, edit, and deploy copies of the workflows included in this solution pack.

For more information about prerequisites for MS SQL database, refer to the [MS SQL Server Documentation](#).

How this Workflow Works

This workflow performs the following actions:

Uninstalls a SQL Server patch on a standalone 2005/2008/2008R2 instance. The default deployment will only show required parameters.

Steps Executed by the Workflow

The MS SQL Rollback Patch workflow includes the following steps. Each step must complete successfully before the next step can start. If a step fails, the workflow reports a failure and all subsequent steps are skipped.

Steps Used by MS SQL Rollback Patch

Workflow Step	Description
MS SQL Parameters Rollback Patch	This step gathers all the required parameters for a rollback (uninstall) of a SQL Server patch.
MS SQL Gather Advanced Parameters for Rollback Patch	This step gathers all the advanced parameters for a rollback (uninstall) of a SQL Server patch.
Windows Check for Pending Reboot	This step check for any pending reboots.
Check If Download File Exists	This step is designed to facilitate the complicated methodologies that various companies use to distribute their software bundles for installation.
MS SQL Verify Patch Rollback	This step verifies that a rollback of a Windows or SQL Server patch was successful.
Delete Directory	This step deletes a directory (folder).
Windows Restart Server	This step Restart a system
Download Software	This step automates the transfer of files from the HP SA Software Library to individual managed servers for use in downstream workflow steps.
MS SQL Rollback Patch	This step performs a rollback on a Windows or SQL Server patch.
Windows Wait for Restart	This step is to wait 8 minutes for Windows server to finish restart.
Unzip Archive	This step is to unzip a zip archive, verify if the input file exists, ensure the output directory exists, creates required directories, and deploys archived files.
MS SQL Verify Patch Rollback	This step verifies that a rollback of a Windows or SQL Server patch was successful.
Delete File	This step deletes the specified file.
Windows Check for Pending Reboot	This step checks for any pending reboots.

Steps Used by MS SQL Rollback Patch, continued

Workflow Step	Description
Delete Directory	This step deletes a directory (folder).
Windows Restart Server	This step restarts a system.
Discover SQL Databases	This step audits the server's physical environment looking for SQLServer instances and databases.
Windows Wait for Restart	This step is to wait 8 minutes for Windows server to finish restart.

Note: For input parameter descriptions and defaults, see [Parameters for MS SQL - Drop Database](#).

How to Run this Workflow

The following instructions show you how to customize and run the MS SQL Rollback Patch workflow in your environment.

The workflow provides default values for some parameters. These default values are usually sufficient for a "typical" installation. You can override the defaults by specifying parameter values in the deployment. You can also expose additional parameters in the workflow, if necessary, to accomplish more advanced scenarios. Any parameters not explicitly specified in the deployment will have the default values listed in ["Parameters for MS SQL Rollback Patch" on the next page](#).

Note: Before following this procedure, review the ["Prerequisites for this Workflow" on page 106](#), and ensure that all requirements are satisfied.

To use the Run MS SQL Rollback Patch workflow:

1. Create a deployable copy of the workflow.
2. Determine the values that you will specify for the following parameters:

Note: There are no mandatory parameters required to run this workflow. All parameters are optional. You may need to expose additional parameters depending on your objectives.

See [Parameters for MS SQL - Drop Database](#) for detailed descriptions of all input parameters for this workflow, including default values.

3. In the workflow editor, expose any additional parameters that you need. You will specify values for those parameters when you create the deployment or at runtime.
4. Save the changes to the workflow (click **Save** in the lower right corner).
5. Create a new deployment.
6. On the Parameters tab, specify values (or set the type to Runtime Value) for the required parameters listed in step 2 and any additional parameters that you have exposed. You do not need to specify values for those parameters whose default values are appropriate for your environment.
7. On the Targets tab, specify one or more targets for this deployment.
8. Save the deployment (click **Save** in the lower right corner).
9. Run the workflow using this deployment, specifying any runtime parameters.

To verify the results:

The workflow will complete and report SUCCESS on the Console if it has run successfully. If an error occurs during workflow execution, the error is logged, and the workflow terminates in the FAILURE state. Also verify by checking that the target database no longer appears in the DMA Environment section.

Parameters for MS SQL Rollback Patch

The following tables describe the required and optional input parameters for this workflow. Some of these parameters may not be initially visible in a deployment. For some parameters, if you do not specify a value for a parameter, a default value is assigned.

Note: Only those parameters that are configurable in a standard deployment are listed here. Input parameters that must be mapped to output parameters of previous steps are not listed.

Input Parameters Defined in this Step: MS SQL Parameters Rollback Patch

Parameter Name	Default Value	Required	Description
Patch Name	no default	required	Name of the patch, the KB number of the patch, or "Latest Patch" to automatically rollback latest patch on instance. This field is case-insensitive.

Send Documentation Feedback

If you have comments about this document, you can [contact the documentation team](#) by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Workflows for Microsoft SQL Server (Database and Middleware Automation 10.40)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to hpe_dma_docs@hpe.com.

We appreciate your feedback!