

D E L P H I X

Quick Start Guide for SQL Server (Microsoft SQL Server on Windows)

April 2019

Quick Start Guide for SQL Server (Microsoft SQL Server on Windows)

You can find the most up-to-date technical documentation at:

docs.delphix.com The Delphix Web site also provides the latest product updates.

If you have comments about this documentation, submit your feedback to: infodev@delphix.com

© 2019 Delphix Corp. All rights reserved.

Delphix and the Delphix logo and design are registered trademarks or trademarks of Delphix Corp. in the United States and/or other jurisdictions.

All other marks and names mentioned herein may be trademarks of their respective companies.

Delphix Corp.

1400 Seaport Blvd, Suite 200

Redwood City, CA 94063

Quick Start Guide for SQL Server (Microsoft SQL Server on Windows)

- Overview
- Deploy OVA on VMware
- Setup Network Access to Delphix Engine
- Setting Up the Delphix Engine
 - Administrators
 - System Time
 - Network
 - Storage
 - Serviceability
 - Authentication
 - Registration
 - Summary
- Source Environment Requirements
 - Windows Domain User Requirements
 - Database User Requirements
 - Source SQL Server Database User Requirements
- Target Environment Requirements
 - Windows User Requirements
- Validated Sync Environment Requirement
 - Requirements for SQL Server Validated Sync Target Environments
 - Windows Domain User Requirements
- Add the Validated Sync Environment
- Add the Target Environment
- Add the Source Environment
- Linking a SQL Server Data Source (dSource)
- Provisioning a SQL Server Virtual Database (VDB)
 - Provisioning by Snapshot or LogSync
- Next Steps

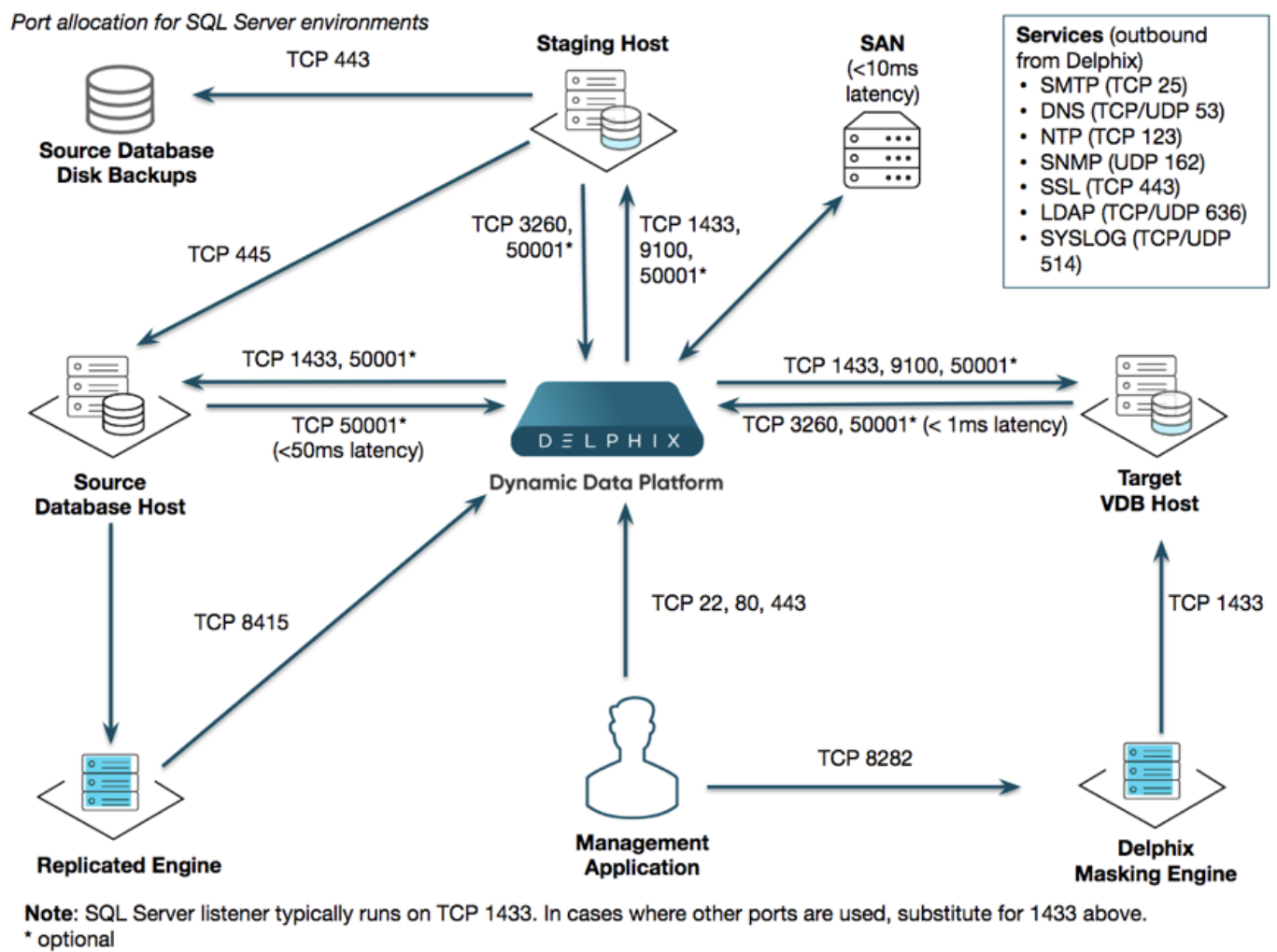
This quick start guide, which is excerpted from the larger User Guide, is intended to provide you with a quick overview of working with SQL Server database objects in the Delphix Engine. It does not cover any advanced configuration options or best practices, which can have a significant impact on performance. It assumes that you are working in a Lab/Dev setting and attempting to quickly test Delphix functionality. It assumes you will use the VMware Hypervisor. It assumes you are running supported combinations of software as explained here: [Supported OS](#), [SQL Server](#), and [Backup Software Versions for SQL Server](#).

Overview

In this guide, we will walk through deploying a Delphix Engine, starting with configuring Source, Staging (aka Validated Sync), and Target database environments on Windows servers. We will then create a dSource, and provision a VDB.

The following diagram describes the Delphix topology for SQL Server environments. It illustrates the recommended ports to be open from Delphix to remote services, to the Delphix Engine, and to the Source, Target, and Validated Sync Environments.

For purposes of the Quick Start, you can ignore any references to Replication or Masking, such as the engines shown in the diagram below.



Deploy OVA on VMware

Use the Delphix-supplied OVA file to install the Delphix Engine. The OVA file is configured with many of the minimum system requirements and deploys to one 300GB hard disk with 8 vCPUs and 64GB RAM. The underlying storage for the install is assumed to be redundant SAN storage.

- Download the OVA file from <https://download.delphix.com>. You will need a support login from your sales team or a welcome letter.
 - Navigate to "Virtual Appliance" and download the appropriate OVA. If you are unsure, use the **HWv10 OVA** type.
- Login using the vSphere client to the vSphere server (or vCenter Server) where you want to install the Delphix Engine.
- In the vSphere Client, click **File**.
- Select **Deploy OVA Template**.
- Browse to the OVA file.
- Click **Next**.
- Select a **hostname** for the Delphix Engine.
This hostname will also be used in configuring the Delphix Engine network.
- Select the **data center** where the Delphix Engine will be located.
- Select the **cluster** and the **ESX host**.
- Select one (1) **data store** for the **Delphix OS**. This datastore can be **thin-provisioned** and must have enough free space to accommodate the 300GB comprising the Delphix operating system.
- Select four (4) or more **data stores** for Database Storage for the Delphix Engine. The Delphix Engine will stripe all of the Database Storage across these VMDKs, so for optimal I/O performance each VMDK must be equal in size and be configured **Thick Provisioned - Eager Zeroed**. Additionally, these VMDKs should be distributed as evenly as possible across all four SCSI I/O controllers.
- Select the **virtual network** you want to use.
If using multiple physical NICs for link aggregation, you must use vSphere NIC teaming. Do not add multiple virtual NICs to the Delphix Engine itself. The Delphix Engine should use a single virtual network. For more information, see [Optimal Network Architecture for the Delphix Engine](#).
- Click **Finish**.
The installation will begin and the Delphix Engine will be created in the location you specified

If your source database is 4 TB, you probably need 4 TB of storage for the Delphix Engine. Add at least 4 data disks of similar size for the Delphix VM. For example: for a source database of 4 TB, create 4 VMDKs of 1 TB each.

For a full list of requirements and best practice recommendations, see [Virtual Machine Requirements for VMware Platform](#).

Setup Network Access to Delphix Engine

1. Power on the Delphix Engine and open the Console.
2. Wait for the Delphix Management Service and Delphix Boot Service to come online.
This might take up to 10 minutes during the first boot. Wait for the large orange box to turn green.
3. Press any key to access the sysadmin console.
4. Enter `sysadmin@SYSTEM` for the username and `sysadmin` for the password.
5. You will be presented with a description of available network settings and instructions for editing.

Delphix Engine Network Setup

To access the system setup through the browser, the system must first be configured for networking in your environment. From here, you can configure the primary interface, DNS, hostname, and default route. When DHCP is configured, all other properties are derived from DHCP settings.

To see the current settings, run "get." To change a property, run "set <property>=<value>." To commit your changes, run "commit." To exit this setup and return to the standard CLI, run "discard."

`defaultRoute` IP address of the gateway for the default route -- for example, "1.2.3.4."

`dhcp` Boolean value indicating whether DHCP should be used for the primary interface. Setting this value to "true" will cause all other properties (address, hostname, and DNS) to be derived from the DHCP response

`dnsDomain` DNS Domain -- for example, "delphix.com"

`dnsServers` DNS server(s) as a list of IP addresses -- for example, "1.2.3.4,5.6.7.8."

`hostname` Canonical system hostname, used in alert and other logs -- for example, "myserver"

`primaryAddress` Static address for the primary interface in CIDR notation -- for example, "1.2.3.4/22"

Current settings:

```
defaultRoute: 192.168.1.1
dhcp: false
dnsDomain: example.com
dnsServers: 192.168.1.1
hostname: Delphix
primaryAddress: 192.168.1.100/24
```

6. Configure the hostname. If you are using DHCP, you can skip this step.

```
delphix network setup update *> set hostname=<hostname>
```

Use the same hostname you entered during the server installation.

7. Configure DNS. If you are using DHCP, you can skip this step.

```
delphix network setup update *> set dnsDomain=<domain>
delphix network setup update *> set
dnsServers=<server1-ip>[,<server2-ip>,...]
```

8. Configure either a static or DHCP address.

DHCP Configuration

```
delphix network setup update *> set dhcp=true
```

Static Configuration

```
delphix network setup update *> set dhcp=false
delphix network setup update *> set
primaryAddress=<address>/<prefix-len>
```

The static IP address must be specified in CIDR notation (for example, 192.168.1.2/24)

9. Configure a default gateway. If you are using DHCP, you can skip this step.

```
delphix network setup update *> set defaultRoute=<gateway-ip>
```

10. Commit your changes. Note that you can use the `get` command prior to committing to verify your desired configuration.

```
delphix network setup update *> commit
Successfully committed network settings. Further setup can be done
through the browser at:

    http://<address>

Type "exit" to disconnect, or any other commands to continue using the
CLI.
```

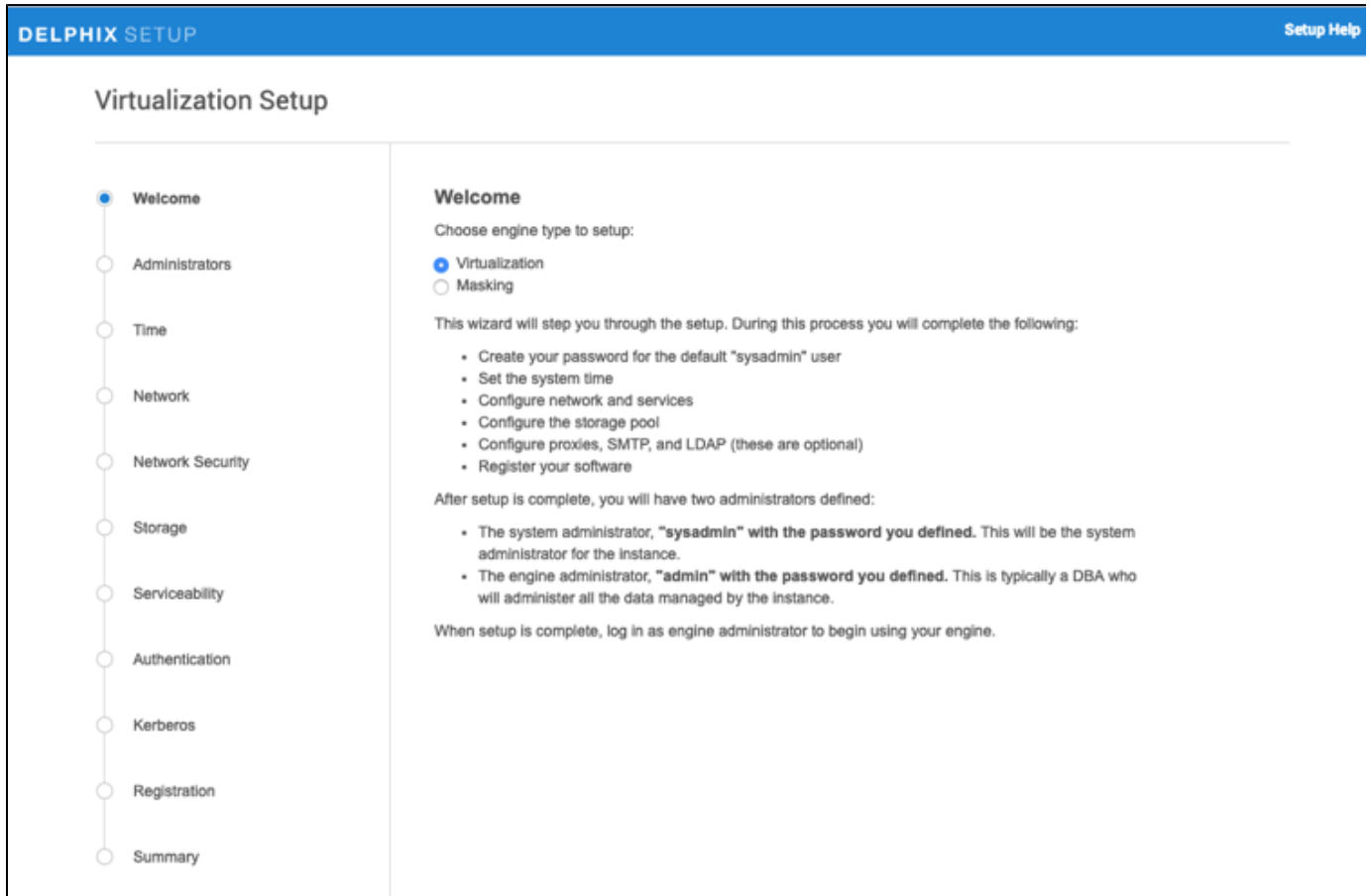
11. Check that you can now access the Delphix Engine through a Web browser by navigating to the displayed IP address, or hostname if using DNS.
12. Exit setup.

```
delphix> exit
```

Setting Up the Delphix Engine

Once you setup the network access for Delphix Engine, navigate to the Delphix Engine URL in your browser for server setup.

The welcome screen below will appear for you to begin your Delphix Engine setup.



Delphix Engine Setup Screen

The setup procedure uses a wizard process to take you through a set of configuration screens:

- Administrators
- Time
- Network
- Network Security
- Storage
- Serviceability
- Authentication
- Kerberos
- Registration
- Summary

1. Connect to the Delphix Engine at <http://<Delphix Engine>/login/index.html#serverSetup>. The **Delphix Setup** application will launch when you connect to the server. Enter your **sysadmin** login credentials, which initially defaults to the username **sysadmin**, with the initial default password of **sysadmin**. On first login, you will be prompted to change the initial default password.
2. Click **Next**.

Administrators

The Delphix Engine supports two types of administrators:

- System Administrator (**sysadmin**) - this is the engine system administrator. The sysadmin password is defined here.
- Engine Administrator (**admin**) - this is typically a DBA who will administer all the data managed by the engine.

The default domain user created on Delphix Engines from 5.3.1 is known as **admin** instead of `delphix_admin`. When engines created before 5.3.1 are upgraded to 5.3.1 or later they will retain their old username 'delphix_admin'. To avoid complications Delphix recommends creating users with an admin role and then Disabling `delphix_admin`.

System Time

Choose your option to setup system time in this section.

For a Quick Start, simply set the time and your timezone. You can change this later.

Network

The initial out-of-the-box network configuration in the OVA file is set to use a single VMXNET3 network adapter.

You have already configured this in the initial configuration. Delphix supports more advanced configurations, but you can enable those later.

Storage

You should see the data storage VMDKs or RDMs you created during the OVA installation. Click **Next** to configure these for data storage.

Serviceability

Choose your options to configure serviceability settings.

For a Quick Start, accept the defaults. You can change this later.

Authentication

Choose your options to configure authentication services.

For a Quick Start, accept the defaults. You can change this later.

Registration

If the Delphix Engine has access to the external Internet (either directly or through a web proxy), then you can auto-register the Delphix Engine:

1. Enter your **Support Username** and **Support Password**.
2. Click **Register**.

If external connectivity is not immediately available, you must perform manual registration.

1. Copy the **Delphix Engine registration code** in one of two ways:
 - a. Manually highlight the registration code and copy it to clipboard. Or,
 - b. Click **Copy Registration Code to Clipboard**.
2. Transfer the Delphix Engine's registration code to a workstation with access to the external network Internet. For example, you could e-mail the registration code to an externally accessible e-mail account.
3. On a machine with access to the external Internet, please use your browser to navigate to the Delphix Registration Portal at <http://register.delphix.com>.
4. Login with your Delphix support credentials (username and password).
5. Paste the **Registration Code**.
6. Click **Register**.

Although your Delphix Engine will work without registration, we strongly recommend that you register each Delphix Engine as part of setup. Failing to register the Delphix Engine will impact its supportability and security in future versions.

To regenerate the registration code for a Delphix Engine please refer to, [Regenerating the Delphix Engine Registration Code](#). Delphix strongly recommend that you regenerate this code and re-register the engine regularly to maximize the Support Security of the Delphix Engine. Delphix recommends doing this every six months.

Summary

The final summary tab will enable you to review your configurations for System Time, Network, Storage, Serviceability, and Authentication.

1. Click the **Back** button to go back and to change the configuration for any of these server settings.
2. If you are ready to proceed, then click **Submit**.
3. Click **Yes** to confirm that you want to save the configuration.
4. Click **Setup** to acknowledge the successful configuration.
5. There will be a wait of several minutes as the Delphix Engine completes configuration.

Source Environment Requirements

Each SQL Server source host must meet these requirements:

- Either the source host must belong to the same Windows domain as the target environments, or the domain that the source environment uses must have appropriate cross-domain trust relationships established with the target environments' domains
- Source hosts can be running any supported Windows operating system version. For more information, see the topic [Supported Operating Systems, Server Versions, and Backup Software for SQL Server](#).
- The SQL Server instance on the source host should run as either domain users or local service accounts. Delphix **does not** support running SQL Server instances as local user accounts or Managed Service Accounts (MSA).
- The validated sync environment that the Delphix Engine uses must have access to an existing full backup of the source database in order to create the first full copy. Alternatively, the Windows Database User described below must have permissions to initiate a copy-only full backup of the source database.
- Backup images of the source database, including full, differential, and/or transaction logs, must be available over an SMB share to a staging environment. For more information, see the topic [Setting Up SQL Server Environments: An Overview](#).
- You must enable TCP/IP access for each SQL Server instance that contains a database to which the Delphix Engine will link
 - To enable TCP/IP access, access the **SQL Server Config Manager** and select **Network Configuration > Protocols > TCP/IP**
- If the source database is backed up with third-party backup software like LiteSpeed or Red Gate SQL Backup Pro, you must also install the backup software on the validated sync environment. For backup software compatibility requirements, see the topic [Supported Operating Systems, Server Versions, and Backup Software for SQL Server](#).
- Delphix regularly queries "msdb.dbo.backupset" to find out if a new backup image has been created and needs to be synchronized with Delphix. Microsoft recommends maintaining this table with "sp_delete_backuphistory". Periodically deleting rows from this table improves the performance of queries running against it and reduces the load on the source database.

Linking to Databases on Windows Server Failover Clusters

You can use databases on Windows Server Failover Clusters (WSFC) as data sources. Add the environment as described below, based on which WSFC feature the source databases use:

- Failover Cluster Instances
Add the environment as a **standalone** source using the cluster name or address.
- AlwaysOn Availability Groups
Add the environment as a **cluster** source using the cluster name or address.

If using a Failover Cluster or AlwaysOn data source, the cluster configuration must also meet the following requirements:

- The fully qualified Cluster Name (part of the cluster's Core Resources) must be known to the DNS server used by the Delphix Engine
- The validated sync environment must be able to resolve the Cluster Name using the Cluster IP address (perform a reverse DNS lookup)

Windows Domain User Requirements

The source environment must have a Windows Domain user that the Delphix Engine can use – for example, delphix_src. This is the user that you provide when adding the source environment to the Delphix Engine. The user provides remote read-only access to the Windows Registry for discovering SQL Server instances and databases. This user must meet the following requirements:

- Be a member of the Backup Operators or Administrators group on the source host to allow Windows remote registry access
- If Delphix-initiated copy-only backups are used for the database, the user must be a member of the Administrators group on the source host
- If the source host belongs to a cluster, the user must have these privileges on all hosts that comprise the cluster
- Have access to any SQL Server instances the Delphix Engine needs to discover and link databases from (described in [Tasks for the SQL Server Database Administrator](#))

Database User Requirements

Source SQL Server Database User Requirements

There must be a database user (for example, **delphix_db**) for each source database that will link to the Delphix Engine. Note that this user will not perform any action that could affect production operations; the Delphix Engine uses this user only to issue queries for database names, database sizes, recovery mode and backup information. This user must meet the following requirements:

- Be able to login to the SQL Server instance hosting the source database using SQL Server Authentication over a JDBC connection to the instance.
- Cannot use Windows Authentication to login to the SQL Server instance for the source database.
- For the **master** database in the source instance: have the database role `db_datareader`
 - To set this role, access the **SQL Server Management Studio** and select **Security > Logins > delphix_db > User Mappings**.
- For the **msdb** database in the source instance: have the database role `db_datareader` for reading backup history
 - To set this role, access the **SQL Server Management Studio** and select **Security > Logins > delphix_db > User Mappings**.
- If the Delphix Engine will initiate copy-only full backups of the source database, the database user must have the database role `db_backupoperator` for the source database
 - To set this role, access the **SQL Server Management Studio** and select **Security > Logins > delphix_db > User Mappings**
- Be able to "use" the desired database. For example, the Delphix Engine will periodically connect to the source database user to determine its size using this query: `(SELECT SUM(size) FROM sys.database_files;)`. The results of the query is displayed in the **Configuration** tab of the dSource
- If the source host belongs to an AlwaysOn Availability Group, the database user must be granted access to the following views:
 - VIEW ANY DEFINITION
 - VIEW SERVER STATE
 - Exclusively give privilege to AG as "GRANT view Definition on AVAILABILITY GROUP::[agname] TO [OS_user];"

Target Environment Requirements

Each SQL Server Target Host must meet these requirements:

- If the target host is used as a Connector host, it must have appropriate two-way cross-domain trust relationships established with the source hosts' domains. Please view [Delphix in Multi-domain Windows Environments](#) for further explanations on cross-domain trust requirements.
- The SQL Server instance on the target host should run as either domain users or local service accounts. Delphix does NOT support running SQL Server instances as local user accounts.
- The SQL Server instance on the target host should be the same version or higher than the instance hosting the source database, with one exception. If a source database comes from a SQL Server 2005 instance, then the target hosts that will host VDBs from that source must be running either a SQL Server 2005 instance or a SQL Server 2012 instance or higher.

Upgrading VDBs from SQL Server 2005

You can first provision a VDB to SQL Server 2005 and then upgrade it to a higher version by following the steps described in [Upgrading SQL Server VDBs](#). For more information about compatibility between different versions of SQL Server, see [Windows Server OS and SQL Server DBMS Compatibility Matrices](#).

- The target host must have 64-bit Windows as the operating system. Delphix does not support 32-bit target systems.
- To add a Windows cluster as a Target Server, see [Adding a SQL Server Failover Cluster Target Server](#).

AlwaysOn Targets Not Supported

Delphix supports the use of Failover Cluster Instances (FCI) as Targets, but not Staging Targets.

AlwaysOn Availability Groups are not supported as Targets or Staging Targets.

- If the target host is a VMWare virtual machine, then the Windows Server operating system must be configured to use the VMXNET3 network driver. Refer to [VMware KB 210992](#).
- The operating system version on a target host that will be used for provisioning VDBs should be equal to, or higher than, the operating system on the target that is hosting the staging databases for the dSource from which the VDB is being provisioned. There is no OS compatibility requirement between source and target hosts. For more information, see [Windows Server OS and SQL Server DBMS Compatibility Matrices](#).
- Windows PowerShell 2.0 or higher must be installed. To check the current version, type `$PSVersionTable.PSVersion` in a PowerShell Window.
- Execution of Windows PowerShell scripts must be enabled on the target host. To enable script execution, enter this command while running Windows PowerShell as an Administrator: `Set-ExecutionPolicy Unrestricted`.
- For Windows 2003 target hosts, the following should be installed:
 - Windows Server iSCSI initiator (available for [download](#)).
 - Hotfix documented in Microsoft Knowledge Base article [KB 943043](#).
- The Windows iSCSI Initiator Service should have its Startup Type set to Automatic, and the service should be running. For configuring the Windows iSCSI Initiator Service, see [Requirements for Windows iSCSI Configuration](#).

- Receive Side Scaling (RSS) must be enabled on each network interface to which the Delphix Engine connects. To configure this, see [Receive Side Scaling \(RSS\) for Windows Staging Target and Targets](#).
- The Delphix Connector must be installed, as described in the topics [Overview of Setting Up SQL Server Environments](#) and [Adding a SQL Server Standalone Target Environment](#).

Flash Player Required for Connector Download

A Flash player must be available on the Target Host to download the Delphix Connector when using the Delphix Management application. If the target host does not have a Flash player installed, you can download the connector directly from the Delphix Engine by navigating to this URL: `http://<name of your Delphix Engine>/connector/DelphixConnectorInstaller.msi`

- Shared Memory must be enabled as a Network Protocol for the SQL instances on the target. To enable this: In SQL Server Config Manager, navigate to Client Protocols > Shared Memory.
- TCP/IP access must be enabled for each SQL Server instance on the target host to allow remote connections to instances. To enable TCP/IP access: In SQL Server Config Manager, navigate to Network Configuration > Protocols > TCP/IP.

Windows User Requirements

There must be a Windows user for the target host that the Delphix Engine can use – for example, `delphix_trgt`. This user can be a Windows domain user or a local user. However, using a local user account will prevent the target host from being used as a Staging Target. This user must meet these requirements:

- Be a member of the local Administrators group for access to discovery operations on source hosts, and for mounting iSCSI LUNs that the Delphix Engine presents to the target host.
- Have the server role sysadmin for each SQL Server instance with which the Delphix Engine will communicate. To set this role for the user: In SQL Server Management Studio, navigate to Security > Logins > `delphix_trgt` > Server Roles.
- Have Log on as a batch job rights so the Delphix Engine can remotely execute commands via Powershell. To set this: Using the `secpol.msc` security policy editor, navigate to Local Policies > User Rights Assignment > Log on as a batch job.

Validated Sync Environment Requirement

This topic describes additional requirements for SQL Server environments that will be used as targets for validated sync. You must configure a staging (Validated Sync) environment as a target, with a few additional requirements.

Requirements for SQL Server Validated Sync Target Environments

Each SQL Server target environment used for validated sync must meet these requirements:

- Only standalone target environments can be used as validated sync target environments. Windows Failover Cluster target environments and SQL Server Failover Cluster instances cannot be used.
- The SQL Server instance must be the same version as the instance hosting the source database. For more information about compatibility between different versions of SQL Server, see [SQL Server Operating System Compatibility Matrices](#).
- The owner of the SQL Server instances on the target environment that are used for the staging databases must have SMB read access to the location containing the backup images of the source databases
- If the source database is backed up with third-party backup software like LiteSpeed or Red Gate SQL Backup Pro, you must install the backup software on both the source and the validated sync environment. For backup software compatibility requirements, see [Supported Operating Systems, Server Versions, and Backup Software for SQL Server](#).

Windows Domain User Requirements

- The Windows Domain user that the Delphix Engine uses on a source environment (for example, `delphix_src`) must also be enabled on the Staging Target host
- This user must be a member of the **Backup Operators** or **Administrators** group on the staging environment. This user must have **Log on as a batch job** rights on the staging server so that the Delphix Engine can remotely execute commands via Powershell. To set this:
 - a. Using the `secpol.msc` security policy editor, navigate to **Local Policies**.
 - b. Select **User Rights Assignment**.
 - c. Select **Log on as a batch job**.

The Windows Domain user (for example, `delphix_trgt`) that the Delphix Engine uses on a Staging Target must:

- Meet all the requirements for the Windows user on a target host as outlined in [Requirements for SQL Server Target Hosts and](#)

Databases

- Have SMB read access to the location holding the backup files of the source database

Add the Validated Sync Environment

The order is important. Add the validated sync environment as the first step in setting up the SQL Server topology.

1. From the machine that you want to use as a target, start a browser session and connect to the **Delphix Management** application.
2. Click **Manage**.
3. Select **Environments**.
4. Next to **Environments**, click the **Actions** menu and select **Add Environment**.
5. In the **Add Environment** wizard Host and Server tab, select:
 - a. Host OS: **Windows**
 - b. Host Type: **Target**.
 - c. Server Type: **Standalone**.
6. Click **Next**.
7. In the Environment Settings tab click the download link for the **Delphix Connector Installer**. The Delphix Connector will download to your local machine.
8. On the Windows machine that you want to use as a target, run the Delphix Connector installer. Click **Next** to advance through each of the installation wizard screens.

The installer will only run on 64-bit Windows systems. 32-bit systems are not supported.

- a. For **Connector Configuration**, make sure there is no firewall in your environment blocking traffic to the port on the target environment that the Delphix Connector service will listen to.
 - b. For **Select Installation Folder**, either accept the default folder, or click **Browse** to select another.
 - c. Click **Next** on the installer final 'Confirm Installation' dialog to complete the installation process and then **Close** to exit the Delphix Connector Install Program.
9. Return to the Delphix Management application.
 10. Enter the **Environment Name**, **Host Address**, **Delphix Connector Port**, **OS Username**, and **OS Password** for the target environment.
 11. Click **Submit**.

As the new environment is added, you will see two jobs running in the **Delphix Admin Job History**, one to **Create and Discover** an environment, and another to **Create** an environment. When the jobs are complete, you will see the new environment added to the list in the **Environments** panel.

Add the Target Environment

Repeat the procedure for your target environment.

1. From the machine that you want to use as a target, start a browser session and connect to the **Delphix Management** application.
2. Click **Manage**.
3. Select **Environments**.
4. Next to **Environments**, click the **Actions** menu and select **Add Environment**.
5. In the **Add Environment** wizard Host and Server tab, select:
 - a. Host OS: **Windows**
 - b. Host Type: **Target**.
 - c. Server Type: **Standalone**.
6. Click **Next**.
7. In the Environment Settings tab click the download link for the **Delphix Connector Installer**. The Delphix Connector will download to your local machine.
8. On the Windows machine that you want to use as a target, run the Delphix Connector installer. Click **Next** to advance through each of the installation wizard screens.

The installer will only run on 64-bit Windows systems. 32-bit systems are not supported.

- a. For **Connector Configuration**, make sure there is no firewall in your environment blocking traffic to the port on the target environment that the Delphix Connector service will listen to.
- b. For **Select Installation Folder**, either accept the default folder, or click **Browse** to select another.

- c. Click **Next** on the installer final 'Confirm Installation' dialog to complete the installation process and then **Close** to exit the Delphix Connector Install Program.
9. Return to the Delphix Management application.
10. Enter the **Environment Name**, **Host Address**, **Delphix Connector Port**, **OS Username**, and **OS Password** for the target environment.
11. Click **Submit**.

As the new environment is added, you will see two jobs running in the **Delphix Admin Job History**, one to **Create and Discover** an environment, and another to **Create** an environment. When the jobs are complete, you will see the new environment added to the list in the **Environments** panel.

Add the Source Environment

Delphix does not require running the Connector on your source. Instead, you'll use the Validated Sync environment as a connector environment to discover the source by proxy.

1. Login to the **Delphix Management** application.
2. Click **Manage**.
3. Select **Environments**.
4. Next to **Environments**, click the **Actions** menu and select **Add Environment**.
5. In the **Add Environment** wizard, Host and Server tab select:
 - a. Host OS: **Windows**
 - b. Host Type: **Source**
 - c. Server Type:
 - If you are adding a Windows Server Failover Cluster (WSFC), add the environment based on which WSFC feature the source databases use:
 - Failover Cluster Instances
Add the environment as a **standalone** source using the **cluster name** or **address**.
 - AlwaysOn Availability Groups
Add the environment as a **cluster** source using the **cluster name** or **address**.
 - Otherwise, add the environment as a **standalone** source.
6. Click **Next**.
7. In the Environment Settings tab select a **Connector Environment**.
Connector environments are used as proxy for running discovery on the source. If no connector environments are available for selection, you will need to set them up as described in [Adding a SQL Server Standalone Target Environment](#). Connector environments must:
 - have the Delphix Connector installed
 - be registered with the Delphix Engine from the host machine where they are located.
8. Enter the **Environment Name**, **Node Address**, **OS Username**, and **OS Password** for the source environment.
9. Click **Submit**.

As the new environment is added, you will see multiple jobs running in the Delphix Admin Job History to Create and Discover an environment. In addition, if you are adding a cluster environment, you will see jobs to Create and Discover each node in the cluster and their corresponding hosts. When the jobs are complete, you will see the new environment added to the list in the **Environments** panel. If you don't see it, click the **Actions** menu and select **Refresh All**.

Linking a SQL Server Data Source (dSource)

1. Login to the **Delphix Management** application.
2. Click **Manage datasets** and click the  icon. Then select **Add dSource** from the drop-down menu.
3. In the first page of the **Add dSource** wizard, select the **source database** by highlighting it.
4. Verify the **version**, **source environment** name, **environment user** account name, **SQL instance** name, and **database name** of the source database.

Changing the Environment User

If you need to change or add an **environment user** for the source database, see [Managing Environment Users on Windows](#) .

5. Enter the **database login credentials** for the source database.
6. Click **Verify Credentials**.
7. Click **Next** to go to the second page of the **Add dSource** wizard, entitled **dSource Configuration**.
8. By default, the name given to the dSource will be the same as the name of the **source database**. You can change it by typing a new name in the **dSource Name** text box.

If the name of the **source database** contains any non-ASCII characters, then please ensure that the **dSource Name** contains **only** ASCII characters.

9. Specify a **Dataset Group** (shown as **Target Group**) in which the new dSource will reside. This target group can either be an existing dataset group (one of the groups suggested in the selection box) or you can create a new dataset group by clicking on the **Add Dataset Group** link.

Dataset Group assignments are permanent

Once a dSource has been assigned to a target Dataset Group, it cannot be changed afterward. The only way to move a dSource to a different Dataset Group is to drop it and recreate it.

Dataset Groups and Permissions

By default, a dSource and its objects (like snapshots) inherit the Delphix Domain user permissions and policies associated with the **Dataset Group** to which the dSource belongs. You can override those policies and permissions on a case-by-case basis. For more information, see the topics under [Users, Permissions, and Policies](#).

10. Click **Next**.
11. You must first choose how the Delphix Engine will capture the **initial load image** of the source database within the dSource, and then keep the image in sync with the source database. If you choose to enable **Delphix Managed Backups**, then the Delphix Engine becomes the backup solution for your SQL Server source database. If selected, then Delphix will periodically take a copy-only full backups of your source database based on the schedule specified by the selected **SnapSync Policy**. For more information, see [Delphix as a Backup Solution to SQL Server](#).

This selection is permanent

It is only possible to enable Delphix Managed Backups on this page. Once a dSource has been linked, you cannot modify the use of this feature. If you enable this feature, the dSource can only use Delphix-initiated copy-only full backups to stay in sync with its source; the Delphix Engine will prohibit linking from existing backups.

- a. First, click the checkbox to enable **Delphix Managed Backups**.
 - b. Next, select the **Staging Environment** and **SQL Instance** that will be used to manage the staging database used for the **dSource**.
 - c. Then select the **SnapSync Policy** that will determine when and how frequently the Delphix Engine takes copy-only backups of the source database.
 - i. If you want to create a new SnapSync policy, refer to the instructions for [Creating Custom Policies](#).
 - ii. If you want to view or modify one of the existing policies, refer to the instructions for [managing policies](#).
12. If you do not enable **Delphix Managed Backups**, then you must specify how the Delphix Engine will capture the initial image of the source database and how it will keep the **dSource** synchronized with the source database.
- a. Select one of the following **Initial Load** options:
 - Delphix will take a **copy-only full backup** of the source database.
 - Delphix will use the **most recent full or differential backup**.
 - Delphix will use a **specific full or differential backup**. This option requires that a **Backup Set UUID** value be provided to identify the backup.

For more details on the three **Initial Load** options, see [Linking a dSource from a SQL Server: An Overview](#).
 - b. Next, specify **Backup Paths** in which the backups of the source database are available for the Delphix Engine to restore.
 - Select **Autodiscover** to have the Delphix Engine automatically locate the backups by querying the MSDB database in the SQL instance.
 - Otherwise, you can add multiple paths where backup files are available. To add a path, click on the (pencil icon). To add more paths, click on the + icon. For each path, please specify the top level of a particular backup path using UNC naming conventions, such as "\\hostName\shareName\subDirectory". Be aware that only **hostName** and **shareName** are required since the Delphix Engine will recursively search for the backup file in all sub-directories beneath the specified path. Specifying optional **subDirectory** names will limit the scope of the recursive search.
 - c. Select the **Staging environment** and **SQL Instance** that will be used to manage the staging database used for validated sync of the **dSource**.
 - d. Select the **Validated Sync Mode** that will be used to keep the dSource synchronized with the source database. Options are:
 - Transaction log backups with LogSync disabled, see [Advanced Data Management Settings for SQL Server dSources](#)
 - Transaction log backups with LogSync enabled, see [Advanced Data Management Settings for SQL Server dSources](#)
 - Full or differential backups
 - Full backups
 - None

Note that the Delphix **LogSync** service is **disabled** by default. For more information about how **LogSync** functions and its advantages and disadvantages, see [LogSync for SQL Server dSources](#).

13. Click the **Advanced** link at the bottom of the wizard page to specify the following additional options for linking the dSource to the source

database.

- a. If the source database is backed up using LiteSpeed or RedGate with password protected encryption, you can supply the **encryption key** that the Delphix Engine should use to restore those backups.
 - b. You can specify **Pre-scripts** and/or **Post-scripts** that the Delphix Engine should run before and/or after each SnapSync operation. For details on Pre- and Post-scripts, refer to [Customizing SQL Server Management with Pre- and Post-Scripts](#). Please be aware that Pre- and Post-Scripts are an earlier functionality than **Hooks**, and are provided mainly for backward compatibility. If you are creating a new dSource and want to add functional customizations, it is better to specify that functionality using Hooks on the next page of the Add dSource wizard.
 - c. You can also specify a **Retention Policy** that will determine how long the Delphix Engine should retain snapshots and transaction log files for the dSource.
14. If you wish to add custom functionality to run prior to, or after, a sync operation, then please follow the instructions to create hooks as shown in [Hooks for SQL Server](#).
 15. Click **Next**.
 16. Review the **dSource Configuration** and **Data Management** information. If anything is incorrect, go back to a prior page and correct it.
 17. Click **Submit** to start the creation of the **dSource**.

For initial load, the Delphix Engine will start two internal jobs to create and load the dSource, named **DB_Link** and **DB_Sync**. You can monitor these jobs by clicking **Actions** in the top menu bar, or by selecting **Event Viewer** from the **System** item in the menu bar. When the jobs have completed successfully, the icon for the source database in the **Environments > environment > Databases** window will change to a **dSource** icon, and the dSource will appear in the left pane of the **Datasets** window under its assigned group.

You can view the current state of **Validated Sync** for the dSource on the **dSource pane**.

The dSource Pane

After you have created a dSource, the **Configuration** tab of the **dSource** management panel allows you to view information about it and make modifications to its policies and permissions. In the left pane of the **Datasets** window, click the dSource's name to view the dSource management panel. Then click the **Configuration** tab. For more information, see the topic [Advanced Data Management Settings for SQL Server dSources](#).

Provisioning a SQL Server Virtual Database (VDB)

1. Login to the **Delphix Management** application.
2. Click **Manage**.
3. Select **Datasets**.
4. Select a **dSource**.
5. Click **TimeFlow** tab.
6. Next to a snapshot select the



Provision VDB icon.

The **Provision VDB** panel will open, and the **Database Name** and **Recovery Model** will auto-populate with information from the dSource.

7. Select a **target environment**.
8. Select an **Instance** to use.
9. If the selected target environment is a Windows Failover Cluster environment, select a drive letter from **Available Drives**. This drive will contain volume mount points to Delphix storage.

Windows Cluster Volume Management Software Requirements

Only cluster volumes managed by the native Windows Volume Manager are supported. For example, cluster volumes managed by Veritas VxVM are not supported.

If you use third-party volume management software, create a new LU (recommended to be 10GB in size) and add this LU as a clustered resource to the SQL Server instance using native Windows volume management tools. Assign a drive letter for this LU. You can then use this LU as the volume mount point location for Delphix VDB provisioning.

10. Enter a **VDB Name** and select a **Target Group** for the VDB.
11. Enable **Auto VDB Restart** to allow the Delphix Engine to automatically restart the VDB when it detects target host reboot.
12. Click **Next**.
13. Select a **Snapshot Policy** for the VDB. Click **Next**.
14. Specify any **Pre- or Post-Scripts** that should be used during the provisioning process.
15. For more information, see [Customizing SQL Server VDB Management with Hook Operations](#).
16. Click **Next**.
17. The final summary tab will enable you to review your configurations.
18. Click **Submit**.

When provisioning starts, the VDB will appear in the **Datasets** panel. Select the VDB and navigate to the **Status** tab to see the progress of the job. When provisioning is complete, you can see more information on the **Configuration** tab.


You can select a SQL Server instance that has a higher version than the source database and the VDB will be automatically upgraded. For more information about compatibility between different versions of SQL Server, see [SQL Server Operating System Compatibility Matrices](#).

Provisioning by Snapshot or LogSync

When provisioning by snapshot, you can provision to the start of any particular snapshot, either by time or LSN.

You can take a new snapshot of the dSource and provision from it by clicking the **Camera** icon.



Provisioning By Snapshot	Description
Provision by Time	You can provision to the start of any snapshot by selecting that snapshot card from the TimeFlow tab, or by selecting  and entering a value in the time entry fields. The values you enter will snap to the beginning of the nearest snapshot.
Provision by LSN	You can use Provision by LSN control to open the LSN entry field. Here, you can type or paste in the LSN to which you want to provision. After entering a value, it will "snap" to the start of the closest appropriate snapshot.

Next Steps

Congratulations! You have provisioned your first virtual database!

Now, perform some simple functional tests with your application. You can connect your app to the VDB using standard TNS/JDBC techniques. Delphix has already registered the VDB for you on the target listener.

We suggest the following next steps:

1. Drop a table and use the VDB Rewind feature to test recovery of your VDB.
2. Take a snapshot of your dSource and refresh your VDB to quickly get fresh production data.
3. Provision a new VDB from your VDB to test sharing data quickly with other teams.
4. Mask your new VDB to protect sensitive data. Provision new VDBs from that masked VDB to quickly provide safe data to development and QA teams.