COHESITY

Cohesity DataPlatform
Protecting Individual MS SQL
Databases Solution Guide

Abstract
This solution guide outlines the workflow for creating backups with Microsoft SQL Server databases and Cohesity
Data Platform.

Table of Contents

About this Guide

This paper details the steps for protecting individual MS SQL Server Databases. IT administrators are defined as individuals who have the role of managing storage and data protection of applications (virtual or physical) in a data center. Databases administrators are individuals who have the purpose of managing Microsoft SQL Servers.

Intended Audience

This paper is written for IT administrators and DBAs familiar with managing data protection of Microsoft SQL Server. Cohesity recommends having familiarity with the following:

- Cohesity DataProtection: Cohesity DataProtection is an end-to-end data protection solution that is fully converged on Cohesity DataPlatform.
- Microsoft SQL Server: Microsoft SQL Server is a relational database management system, which supports a wide variety of transaction processing, business intelligence and analytics applications.

The Microsoft SQL Server referenced in this paper has the following resources
- Windows Server 2012 R2
- 8 vCPU cores
- 8 GB RAM
- 2 x iSCSI disks
- Microsoft SQL Server 2014 (X64) Enterprise ED.

Configuration Overview

Cohesity supports backing up MS SQL Server running on VMware and Physical Servers. The Cohesity Cluster captures Full Database Server Backup and can optionally backup database transaction logs, so you can roll forward to any point in time. To learn more about the relationship between Protection Jobs and Policies, see About Policies and Protection Jobs. For instructions on how to use the Cohesity Dashboard to backup MS SQL Server, see Add or Edit Protection Job to Protect Microsoft SQL Server.

Feature Overview

There are many reasons to protect individual MS SQL Server databases. Some have a higher priority than others, like mission critical databases. These may require more frequent backups, a different schedule, or a different retention period.

The Cohesity SQL Physical agent has been enhanced to protect individual MS SQL Server databases. Cohesity DataProtect now provides the ability to select multiple individual databases, and individual databases across different servers for protection under Cohesity. It supports both Standalone and SQL cluster (FCI) environment.

Installing Cohesity Windows Agent

The Agent is the Cohesity software that you install on physical servers and certain VM hypervisors to enable Cohesity to protect them.

The Cohesity Agent has been enhanced to protect individual databases by tracking the changes at the file level. It uses the FileSystem Change Block Tracker (CBT) driver to capture changes to database files. The Cohesity Agent is not automatically upgraded. A new agent should be manually downloaded from the Cohesity Cluster.

Download Cohesity Agent

Log into the Windows host server with an Administrator level account. Use the internet browser to connect with your Cohesity Cluster. Login into the Cohesity Cluster UI. Navigate to Protection->Sources-> "Download Physical Agent" and download the agent, see figure 1. For more details please refer to Cohesity document Install and Manage the Agent on Windows Servers.
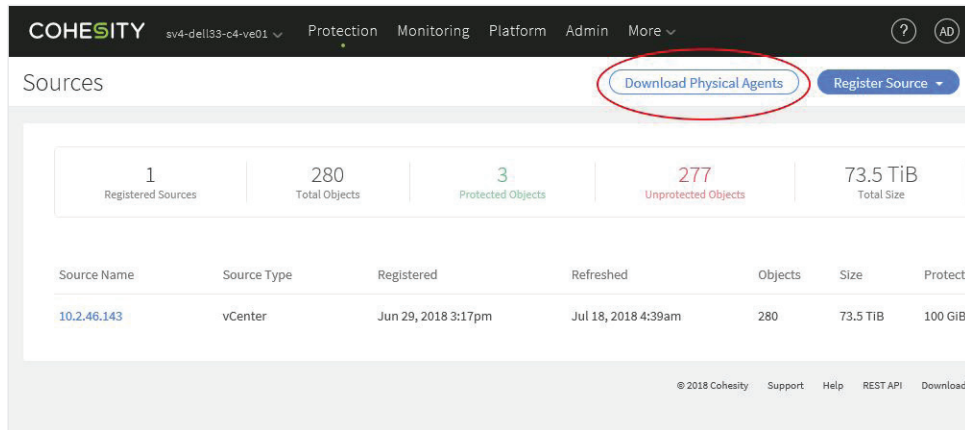
Figure 1 –Download Cohesity Physical Agent

Select Cohesity Windows Agent Type

From the "Download Agents" window, choose the Windows Agent. Ensure the Agent has been downloaded to the Server you want to protect.
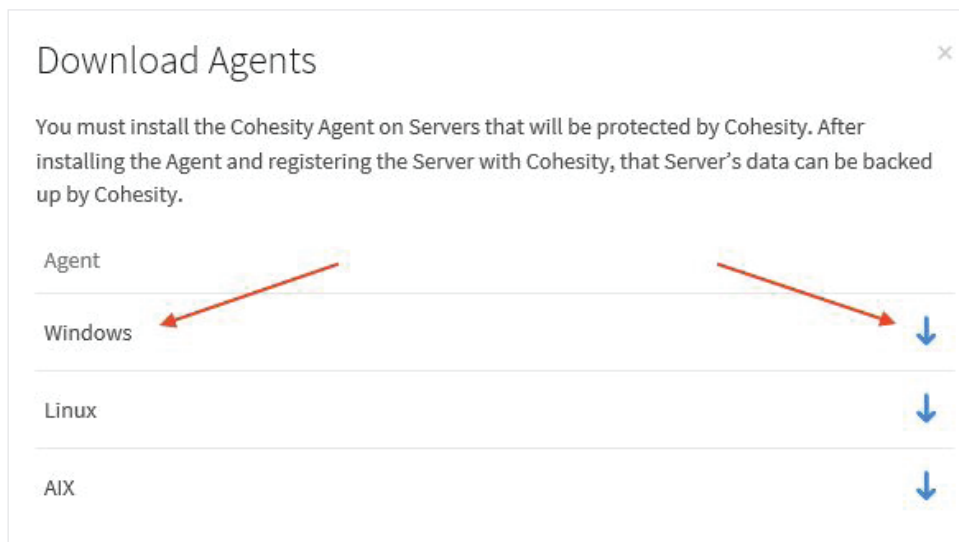

Figure 2 – Cohesity Windows Agent

It is not necessary to uninstall an existing Cohesity Agent if one exists. The installation will update the Cohesity Agent. And in the case of an existing agent, no reboot is required. For first time installs, the Cohesity Agent comes complete with all current enhancements.

Install the Cohesity Agent

As an administrator with local system privileges on the Microsoft SQL Server, run the executable and complete the installation wizard.
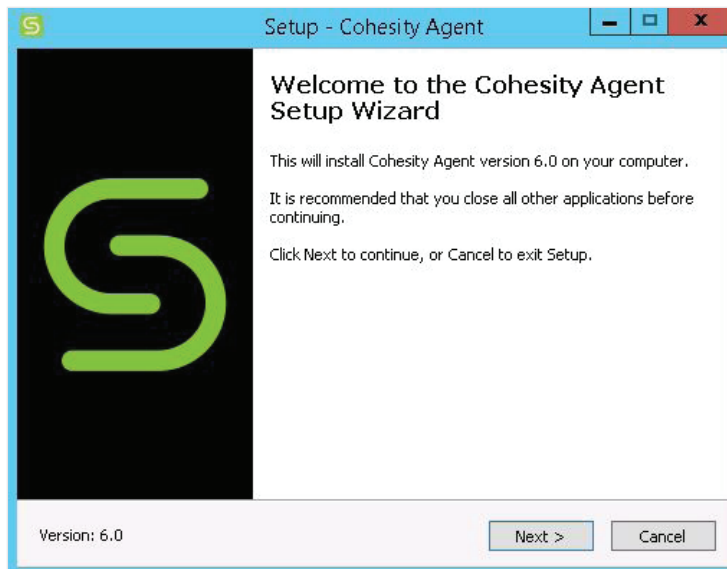
3.

Figure 3 – Cohesity Agent Setup

Cohesity Agent Setup

By default, the **Volume CBT** (Changed Block Tracker) component is selected for installation. This component is required to perform incremental backups and requires a reboot to function. Without a reboot, only Full backups can be performed on the server.

The **File System CBT** (Changed Block Tracker) component is required for MS SQL file-based protection. There is no system reboot required for File System CBT installation on MS SQL Servers that have not been registered with Cohesity DataProtect (i.e. new MS SQL Servers protected with Cohesity DataProtect.)
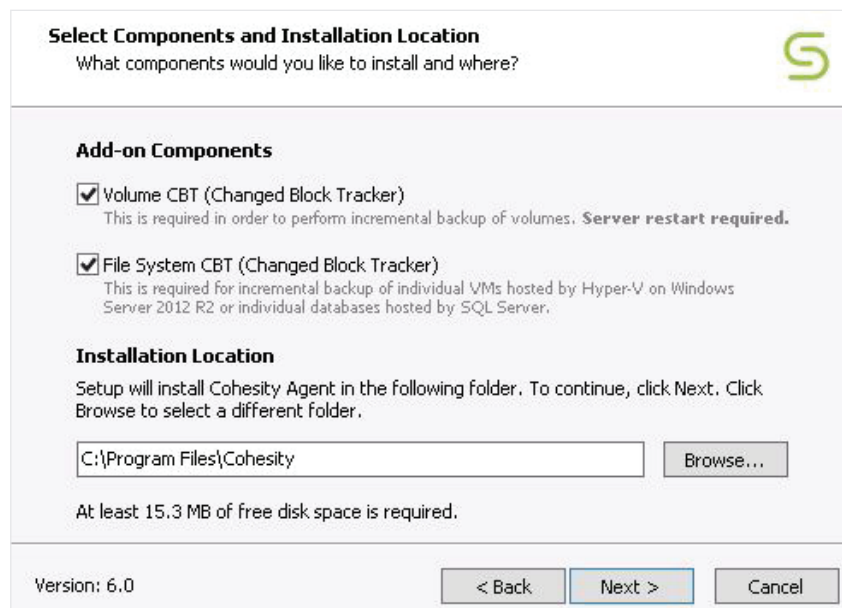

Figure 4 – Cohesity Components

4.

Specify the service account

Cohesity recommends the service account be an Active Directory domain user account and a member of the local administrative group on the MS SQL Windows Server, with logon rights to install the Cohesity Windows Agent. Be sure that the service account has been given the sysadmin server role in the Microsoft SQL Server Instance.

Figure 5 – Specify Service Account

Registration of MS SQL Server

Registration of MS SQL Server with Cohesity DataProtect is a two-step process; Register MS SQL Server host as a Physical Server on the Cohesity Cluster. And secondly, register the MS SQL Server application.

Register as a Physical Source
Login to to the Cohesity Cluster UI. Navigate to Protection(Tab)->Sources->Register Source->Physical Server.
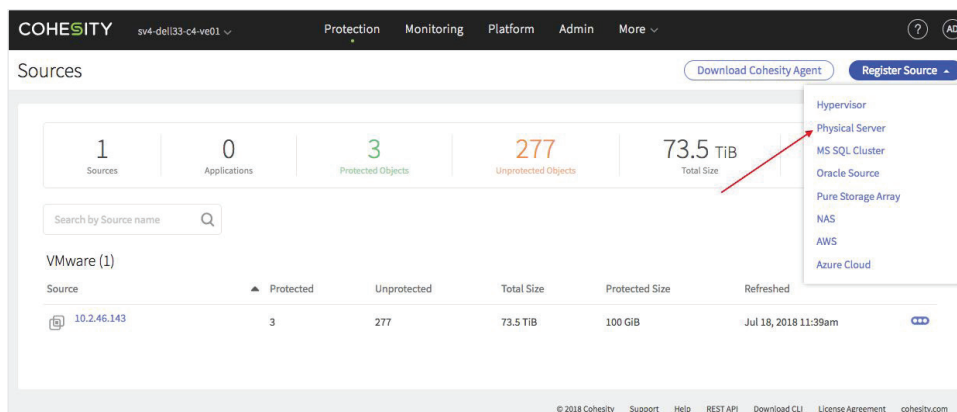
Figure 6 – Cohesity Sources

Cohesity supports registering physical servers by FQDN or IP addresses. However, it is recommended to register the servers using the fully qualified domain name (FQDN). In this example, ' *PROD-VR-Win12r2.qa01.eng.cohesity.com* ' is used to register the server as a source. When Server registration finishes, a new entry will appear in the Physical Servers section on the Sources page.
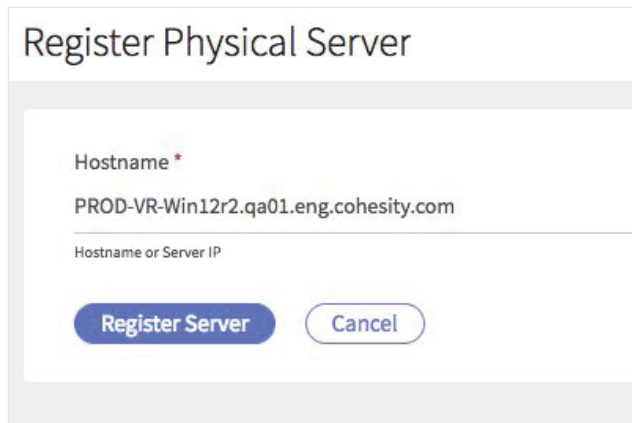
Figure 7 – Using FQDN

Register the server as a MS SQL Server

Now you can register the MS SQL Server that is running on the Physical Server with the Cohesity Cluster. Click on the icon and choose "Register as MS SQL Server". Once the server is registered as a MS SQL Server Cohesity will discover all the MS SQL Instances and databases on that server. More information on registering MS SQL Server as a Cohesity Source can be found in Register a Physical Server.
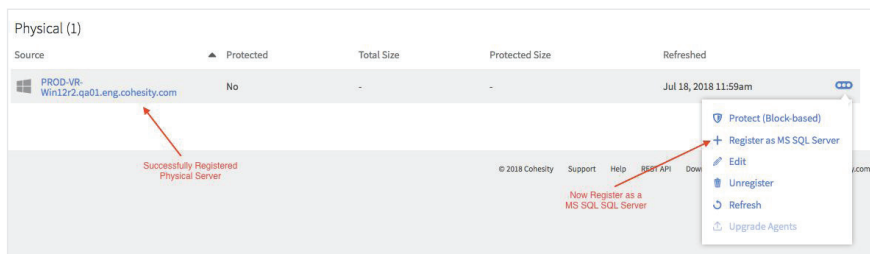

Figure 8 – Cohesity MS SQL Server Source

Successful Registration

Figure 9 shows a MS SQL Server successfully registered as a MS SQL Server Cohesity Source. The MS SQL Server is now ready to be used as a source for a Protection Job. Notice that the MS SQL Server appears in both the Physical and MS SQL sections.
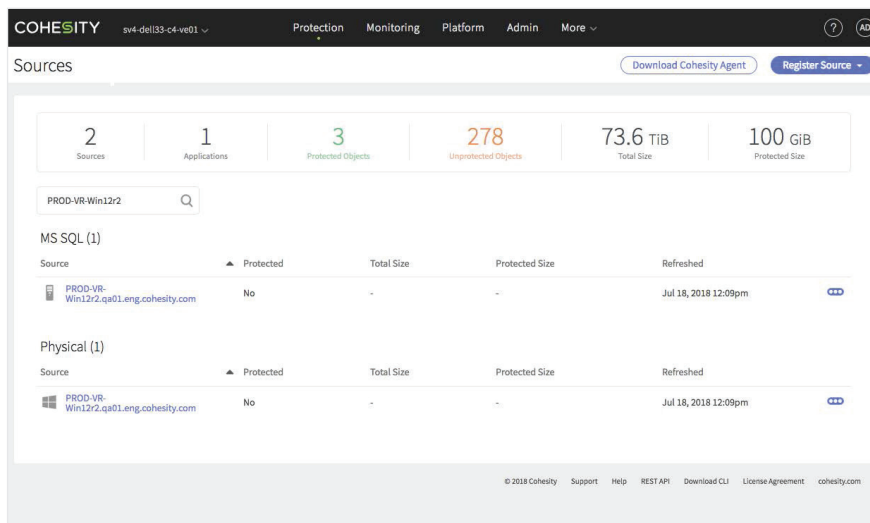

Figure 9 – Cohesity Registration of MS SQL Server.

6.

Create a Protection Job

A Protection Job is a backup job that runs repeatedly, based on an associated Policy, to backup data from a Source and store it on the Cohesity Cluster.

Existing Cohesity Volume Based SQL jobs are not automatically changed to File Based SQL jobs. A new protection job must be created to use the File Based backup method. Cohesity recommends deleting the old Volume Based SQL job (keep the snapshots), and create a new Cohesity File Based job to protect the same MS SQL Server. During the short time between the deletion of the old Volume Based SQL job, and the new File Based SQL job the server will be unprotected.

Protection Tab
Navigate to Protection(Tab)->Protection Jobs->Create Job-> **MS SQL Server**.

Figure 10 – Cohesity Protection Job Tab

New Cohesity MS SQL Job
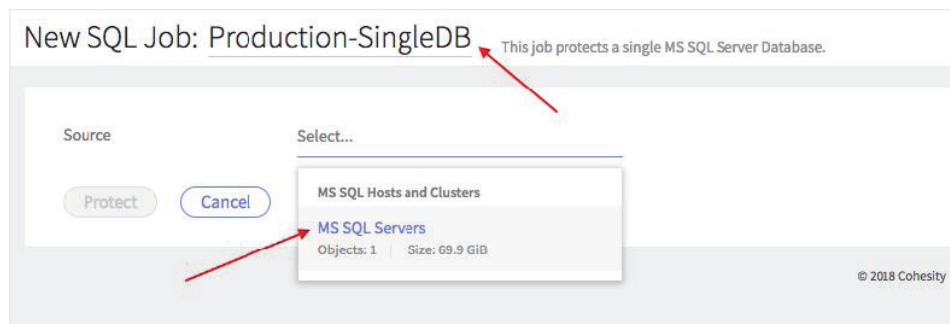Enter a name and description which will identify your Protection Job. Select the Source Type "MS SQL Servers".

Figure 11 – Job Name and Source Type

7.

Cohesity MS SQL Sources

Cohesity presents you with a hierarchical list of SQL objects. Object selection is hierarchical, for example, if you select a Host System parent, all of its child Objects (Instances) are selected, or if you select a SQL Instance, all of its child Objects (Databases) are selected. For file-based protection, select only the databases you want Cohesity to protect.

- **MS SQL Server host:** Choosing the SQL host, protects the host, the SQL Instance and all the databases under the SQL Instance.
- **MS SQL Instance:** Choosing the SQL Instance, protects the SQL Instance and all the databases under the SQL Instance.
- **MS SQL Database:** Choosing the individual SQL databases protects the databases which have been selected. One or more databases can be selected individually.
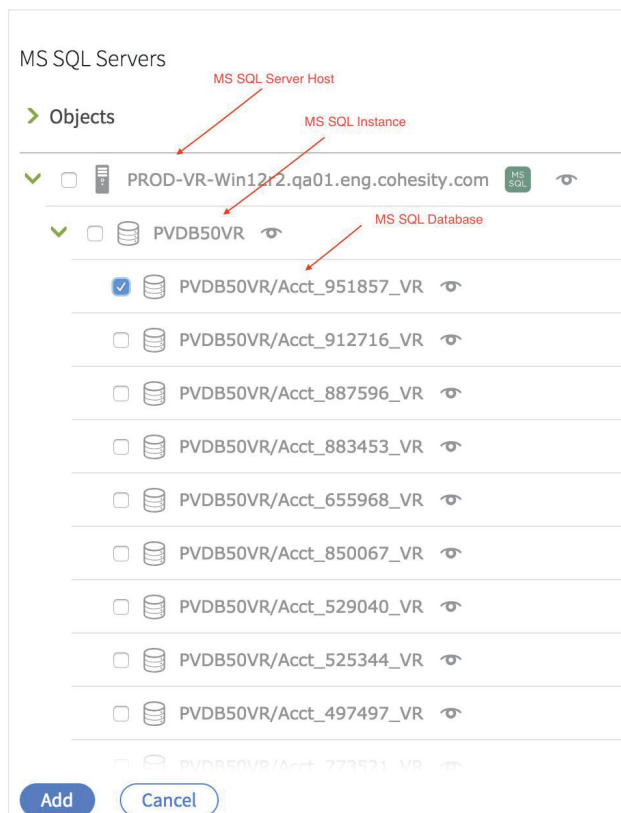

Figure 12 – Cohesity Source Selection

Select a Policy

A Protection Policy is a reusable set of settings that define how and when Virtual and Physical Servers, Views, Database Servers, are protected, replicated and archived. Select an existing protection policy or create a new protection policy and then select Next. Please refer to Cohesity document on " Policy and Protection " for creating new protection policy.
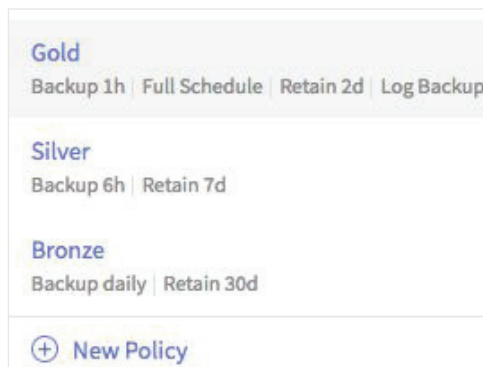

Figure 13 – Cohesity Protection Policy

8.

Select a Storage Domain

A Storage Domain is a named storage location in the Cohesity Partition. A Storage Domain defines the storage policy for deduplication. Choosing a Storage Domain determines where the Cohesity Cluster stores the snapshots. For more details about Storage Domains see Create or Edit Storage Domains.

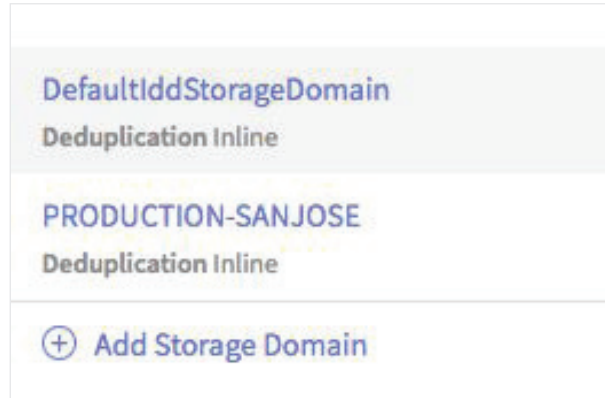Figure 14 – Cohesity Storage Domain

Specify MS SQL Backup Settings

Click on "MS SQL Setting" to expand it. The MS SQL Settings provides the following options.

- **Backup Method** - Choose **File-based** for backing up individual databases. Cohesity File-based backups are much faster because it only backups up the selected MS SQL databases and not the entire MS SQL Server Instance.

- **Copy Only** - Enabling this option will prevent the backup from breaking the log chain. It is recommended to use Cohesity DataProtect for MS SQL to ensure that your MS SQL DB backup does not break the SQL DB log chain on backup. Use Copy-Only backups with MS SQL native backup with Cohesity DataPlatform if a secondary copy via SMB share is required.

- **User Databases** - Select the **User Databases** to backup all user databases. It is recommended to leave at default.

- **Systems Databases** - Select whether to backup or skip System Databases. It is recommended to leave at default

- **AAG Preferences** – Specifies which replica to take a backup from if the database is part of an AAG group. This option applies to MS SQL Always On Availability Groups. Cohesity recommends "Use Server Preferences"

For more details see Protect MS SQL Server.

Figure 15 – Cohesity MS SQL Options

Cohesity gives you the choice of two backup methods; volume based or file based. Under the volume based method Cohesity creates a backup of the volumes for that host server. This captures a backup of the MS SQL Server data and log files and everything else located next to them on that volume. If the volume is large, it may contain a lot of other files which do not necessarily need to be protected. The file based method protects just the MS SQL Server files and will result in a significant reduction in backup and restore times. The volume based method is an option when you protect the source at the host level. The file based method is an option when you protect the source at the individual database level.

Specify Advanced Backup Settings

Advanced Settings are populated with default settings and can optionally be adjusted. For more details read the second half of Protect MS SQL Server.

- **Start Time** - Indicates when the job should run.

- **End Date** - Select the date on which the Protection Job stops capturing Snapshots.

- **QoS Policy** - Select an appropriate Quality of Service (QoS) Policy. Cohesity recommends using the default.

- **Indexing** - Indexing is enabled by default and required for file recovery. Cohesity recommends using the default.

- **Alerts & Priority** – Optional settings if you want Alerts to be created and sent for different events. Alert notification is important for monitoring the success, failure, or SLA violation of the job. Cohesity recommends using the default.

Advanced	
Start Time	10:44am │ America/New_York
End Date	Never
QoS Policy	Backup HDD (Default)
Indexing	Disabled
Alerts & Priority	Alert on Failure │ Medium Priority
SLA	Incremental: 60 minutes │ Full: 120 minutes

Figure 16 – Cohesity Advanced Options

Click the "Protect" button, and the job is created and will start according to the policy set.

Backup MS SQL Database

Protection Job Run

Once the Protection Job is created, it will run based on its Start Time. Subsequent job executions are listed under Job Runs, and will indicate duration, type, and status. MS SQL Server administrators can also use this page to quickly confirm that business-critical databases are protected with Cohesity DataProtect.

Figure 17 – Cohesity Protection Job Run

Protection Job Run Details

In the Cohesity Cluster UI, the MS SQL Database Protection Job -> Backup Run Details will show just the Microsoft SQL Server databases found and protected by Cohesity DataProtect. This is an additional level of detail to confirm that the backup is successful. As an example see figure 18.

Figure 18 – Cohesity Protection Job Run Details

Restore an MS SQL Server Database

Cohesity provides the ability to restore an individual databases. MS SQL Server or individual databases can be restored to their original location or an alternate location. There are two
general restores:

- Clone a database, see Clone an individual MS SQL Server Database
- Restore a Standalone database. This is described below. For more information see Restore an Individual MS SQL Database

Restore a Database back to a MS SQL Server Instance
Navigate to the Recovery screen. Protection->Recovery->Recover Button->MSSQL.
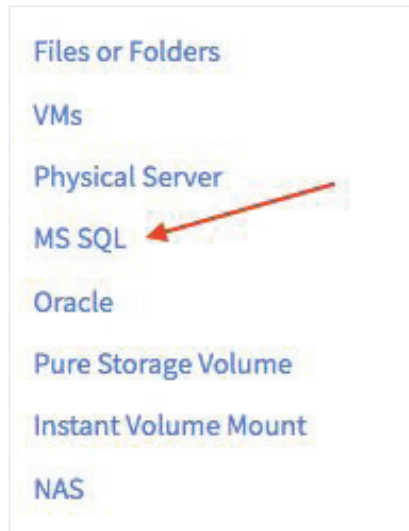
Files or Folders

VMs

Physical Server

MS SQL

Oracle

Pure Storage Volume

Instant Volume Mount

NAS

Figure 19-Cohesity Recover Menu

Choose a Database
The search and filter options provide a powerful way of finding and selecting databases from backups. Enter the characters of a SQL Server or database name and a list of items displays. Select an item from the list. You can optionally specify the wildcard character * . Search results can be narrowed by specifying filter criteria.

Search by MS SQL Server or Database name

Acct_987526_VR

PVDB50VR/Acct_987526_VR
Cohesity Cluster sv4-dell33-c4-ve01 | Protection Job Production-SingleDB
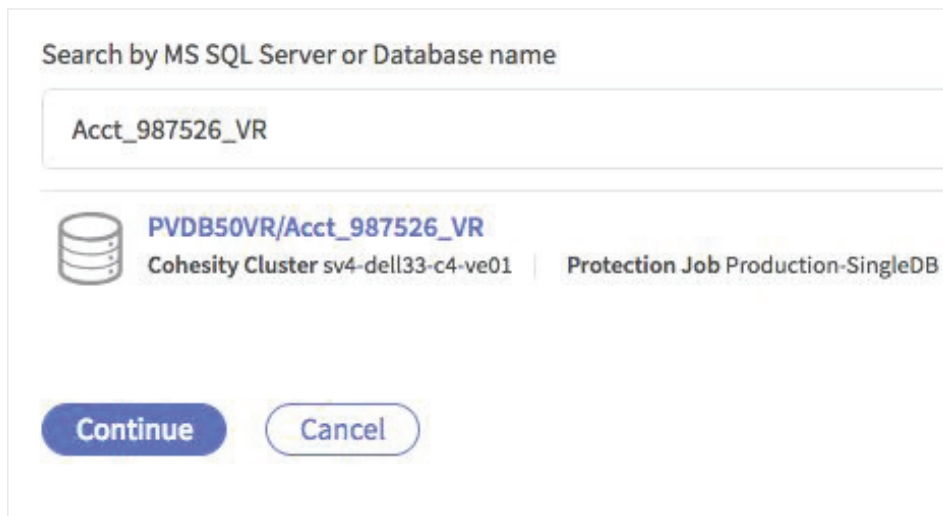
Continue   Cancel

Figure 20-Cohesity Search and Filter

Specify MS SQL Restore Options

Cohesity provides several options for databases restores:

A. If restoring to the original SQL Server instance, you can toggle the Overwrite Original Database if you want to **overwrite the original database** with the restored one.

B. To restore log files, and primary and secondary data files, to different locations, click **Add Alternative Log File and Secondary Data File Locations** and provide locations for each.

C. Provide a **Database Name** for the restored database.

D. By default, an MS SQL restore WITH RECOVERY is performed. You can optionally toggle this off to perform a restore WITH NORECOVERY.

Figure 21 – Cohesity Database Restore Options

Recovery Point

Cohesity "Recovery Point" ✏️ link provides the ability to recover the database to a particular point in time; choose the recovery point, and select the option to specify a point in time. With 6.0 and later, backup administrators can also leverage "Restore to a New Time" to a different MS SQL Server Instance that is registered with Cohesity DataProtect.
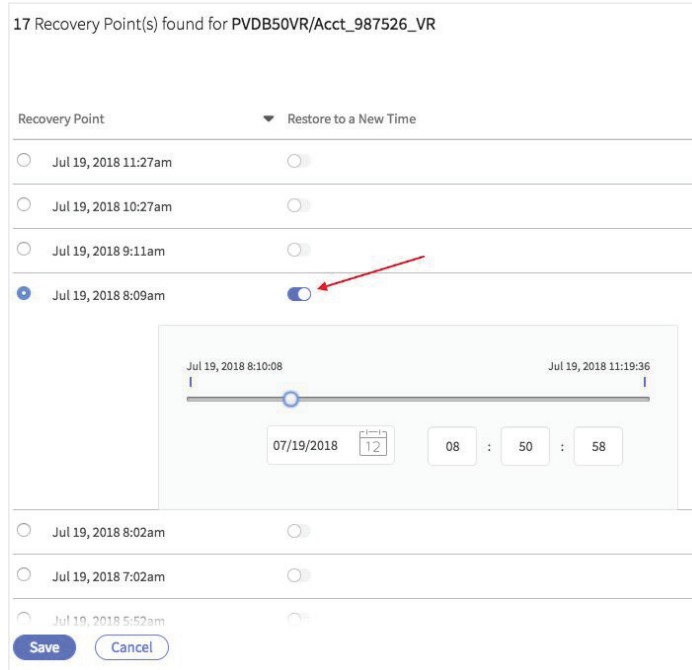

Figure 22 – Cohesity Recovery Point In Time Restore

The Recovery Task is Created. In this step, you can view a summary of the options for the current task and a list of the items to be restored.
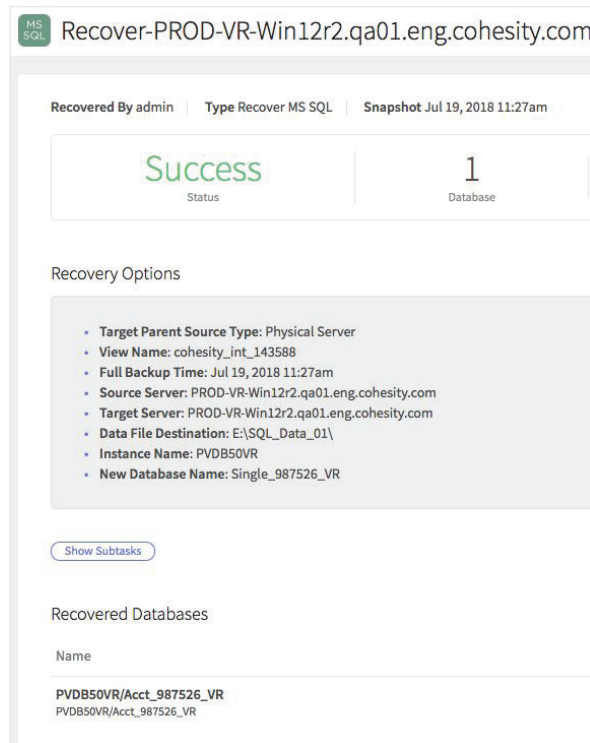

Figure 23 – Cohesity Recovery Task Detail

14.

In the Cohesity Cluster UI, the Show Subtasks button will show subtasks that Cohesity executed in order to complete the restore. It will also show that the Microsoft SQL Server database was successfully restored. See figure 24 below.

Figure 24 – Cohesity Recovery Task execution detail

Verification
A quick visual check using SQL Server Management Studio (SSMS) will verify the successful recovery of the database.
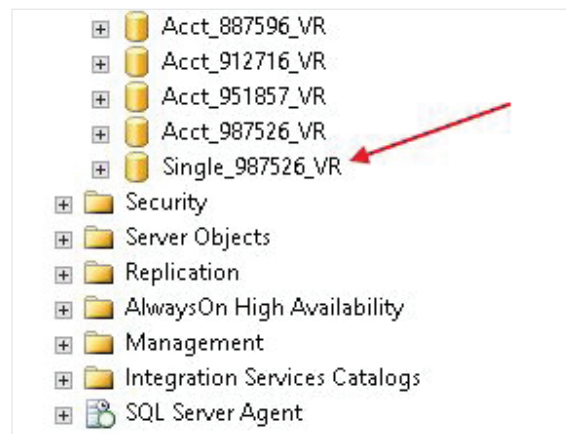

Figure 25 – Verification

15.

COHESITY

Summary

Cohesity DataProtect provides a comprehensive solution to your data center storage requirements. Cohesity DataProtect is uniquely designed to handle MS SQL Server databases; a solution now that allows for individual MS SQL Server database protection.

This means individual MS SQL Server databases are quickly identified, and a Protection Job can be created, executed and monitored from a single pane of glass UI. Individual MS SQL Server databases are effectively protected with all your other MS SQL Server databases. Tailoring your database protection needs down to the individual business-critical database is now easy to do.

About The Author

Scott Lorenz is a Solutions Engineer at Cohesity. In his role, Scott focuses on SQL Server business-critical applications, databases, and data protection with Enterprise and Cloud Storage.

Version History

Version	Date	Document Version History
Version 1.0	July 2018	Original Document