COHESITY

**Version 2.1**

**June 2022**

# Cohesity Custom Reporting Solution Guide

## *Boost Business Intelligence with Cohesity Custom Reporting*

**ABSTRACT**

*In today's data-driven business environment, it is more important than ever to analyze your data and identify patterns and trends that help organizations make better decisions. With custom reporting, Cohesity extends its built-in reporting features to give you much more flexibility to produce the custom reports that address your organization's specific needs most effectively.*

# Table of Contents

# Figures



# Tables

# Introduction

Cohesity custom reporting extends Cohesity's built-in reporting to provide organizations with extensive, customizable reports. You can create reports in multiple formats, across different areas like operations and planning, troubleshooting, forecasting, chargeback, and billing, along with compliance and auditing.

This guide describes the custom reporting features across the Cohesity ecosystem, their purpose, and different use cases for Cohesity customers in all sectors.

## Audience

This guide is written for database administrators (DBAs), SQL developers, and IT staff familiar with basic database operations (such as SQL querying, JDBC connectivity, and the like). The audience has basic knowledge of entity relationship diagrams (ERDs), database schemas, and experience in creating reports.

# Cohesity's Solution for Reporting

As reporting is central to your ability to make informed, data-driven decisions, an effective reporting platform is vital to any organization.

As data collection continues to grow and deepen, the information collected in your systems increases as the utilization of your products grows. But the data collected is not of any significant value until processed. Therefore, it becomes essential to develop a way for end users to consume data in a methodical fashion, in order to proactively identify behavioral trends that can assist decision makers in gaining a competitive edge.

## Challenges

Building a pipeline to collect, process, and present data to end users is a challenge in itself. What's more, most organizations have to solve specific challenges that are unique to their industry or field. For example:

- **Archival period**: Business users often seek to analyze data over long periods, but limitations in metadata retention can make such analyses difficult. For example, while trying to identify historical trends, many organizations find they have already purged older data and can only produce reports with limited insight.
- **Granularity**: Users are often unable to achieve the fine-grain level of detail that they need, such as per-object and per-database granularity. For example, when you need to analyze the history of a specific object in your system, it is often impossible or difficult to isolate just that object's events. While you might know what objects are in a Protection Group, you will often struggle to find out the Protection Groups that all include a particular data object.
- **Personalized report presentation**: Report views often need to be personalized to cater to the needs of the particular level or perspective of the report's recipient. For example, the system administrators need a detailed report on each component's performance while a director or VP wants just a high-level summary.
- **Data aggregation**: Users often need to present data in roll-ups (aggregations), but are currently unable to do so. For example, you can produce a report on system health in a given hour or a given day, but you can also roll up a whole week or month, to identify longer-term trends.

## Our Solutions for Reporting

Cohesity reporting is comprised of these offerings:

Table 1: Cohesity Reporting Solutions

| SCOPE | BUILT-IN | CUSTOM |
|---|---|---|
| Cohesity Platform (Single cluster) | Cohesity Platform Built-in Reports | Cohesity Platform Custom Reports |
| Helios™ (Multi-cluster) | Helios Built-in Reports | Helios Custom Reports * |

* - future

# Built-in Reports

In addition to custom reporting, Cohesity and Helios also provide built-in reports for quick access to basic analysis.

## Cohesity Platform Built-in Reports

Cohesity Platform (Single cluster) Built-in Reports are pre-built reports with filtering capabilities. You can generate reports on demand and view them in different formats (HTML and CSV). You have the option to filter the report on date range and many other attributes (Protection Group name, VM name, status, etc.). You can also set up report schedules and event-triggered email messages.

There are approximately two dozen built-in single cluster reports. The reports are grouped into the following five major groups:

- **Capacity & Usage**: Total capacity vs capacity in use
- **Design & Build**: Backup sources, growth, duplicates, etc.
- **Backup Operations**: Backup Jobs reports
- **Backup Operations**: Object (VMs backed up)
- **Operational**: Recovery, clone, and GDPR security operations

See Cohesity Reports for the complete list of built-in reports.

## Helios Built-in Reports

Helios is Cohesity's SaaS-based management platform that provides a single dashboard and global management of all your Cohesity clusters. Helios provides multi-cluster management to actively manage all your clusters from a single dashboard, including multi-cluster monitoring, reporting, and orchestrated upgrades.

Helios has a growing number of built-in reports with aggregation of data across various clusters, such as capacity usage across clusters.

For more, see Helios Reporting in the documentation.

# Custom Reports

Custom reporting enhances the monitoring options available for historical data. With custom reporting, Cohesity substantially expands your ability to capture and analyze information beyond the scope of the built-in reports.

What's more, database-native access to the underlying data has created the possibility of bringing in the user's choice of third-party tools to build reports that are tailored to their specific business requirements.

## Built-in vs Custom Reports

Built-in reports address many scenarios, such as a report with a fixed range of day-to-day operations, well. However, when users need to slice and dice data to address variations in data type, for example, custom reporting is a better fit, as it provides the flexibility to delve deeper into the data and analyze a specific output.

Table 2 highlights the key differences between built-in and custom reports.

Table 2: Built-in vs Custom Reports

| CATEGORIES | BUILT-IN | CUSTOM |
|---|---|---|
| Customization | Limited | Flexible |
| Tunable granularity | Limited | Flexible |
| Format | CSV, HTML | User-defined |
| Scheduled reports | Yes | Yes |
| Access to metadata | No | Yes |
| Integration with external tools | No | Yes |

## Features and Benefits of Custom Reporting

Custom reporting encompasses many features that deliver benefits across the board.

Table 3: Custom Reporting Features and Benefits

| FEATURES | BENEFITS |
|---|---|
| Database-native access (Cohesity Platform) | • Access to the metadata via industry-standard business intelligence tools.<br>• Support for any third-party tool with a compatible database driver. |

| FEATURES | BENEFITS |
|---|---|
| Up to date | **Cohesity Platform**<br>• Tunable data refresh frequency<br>• User-defined data granularity<br>**Helios**<br>• Refreshes data every 3 – 4 minutes. |
| Flexible | • Reports can be customized to align with specific business needs.<br>• No restrictions on the type of report, scope of visualization, and choice of attributes and measures. |
| Data-driven | • Meet compliance and audit requirements.<br>• Operational visibility<br>• Prioritize remedial action |

## Cohesity Platform Custom Reporting

The Cohesity Platform's custom reporting feature was designed specifically to handle advanced reporting requests. A Custom (Postgres) Database stores relevant performance metrics, statistics, and information from a Cohesity cluster and any third-party reporting tool can connect to the Custom Database to generate reports.

Learn more about the architecture, authentication mechanism, and data collection of Cohesity Platform's custom reporting in the following sections.

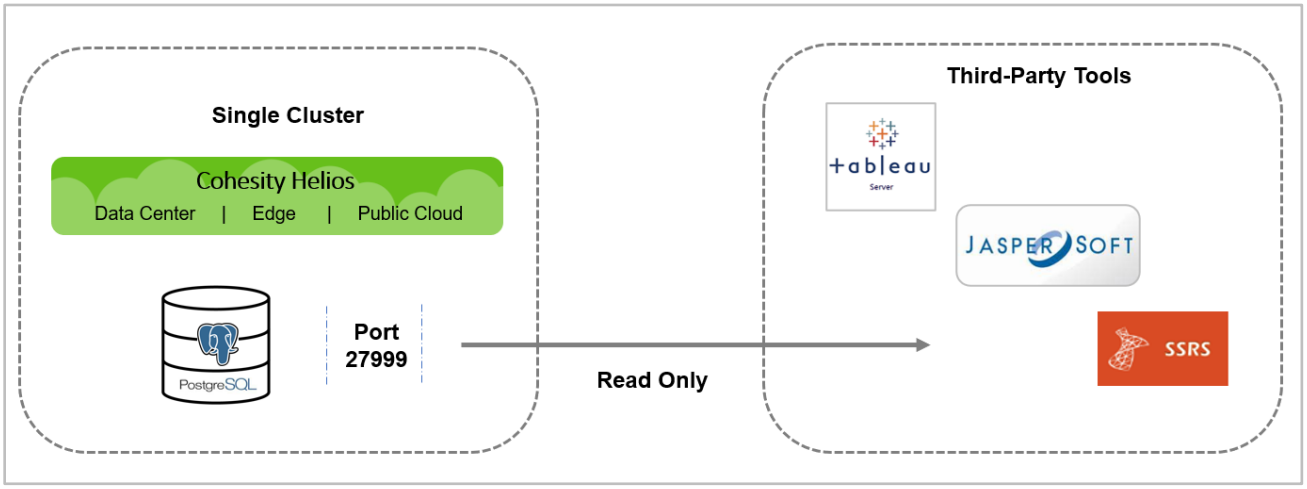
### Architecture

Cohesity stores its metadata in a distributed key value store. This metadata is fetched and loaded into the Custom Database (a Postgres database), which is stored on hard disk. Within each cluster, a Cohesity internal service manages redundant Postgres databases (set during initial cluster configuration) and the data stored in the disk is resilient to node and disk failures. As a result, there is no single point of failure.

The user has read-only access to the database schema. The database is exposed on port 27999. This port needs to be opened up on the network firewall to allow connections to the database from third-party tools.

Figure 1 below illustrates the flow of data—from your source data in Cohesity Platform to the Custom Database to third-party reporting tools—in our single-cluster reporting architecture.

Figure 1: Single-Cluster Reporting Architecture




## Custom Database Schema and ERD

The Custom Database schema is designed to be intuitive. As you will see when you explore the Custom Database's Entity Relationship Diagram and Data Dictionary in the online Help, the table names and the field names in the database are straightforward and self-explanatory.

## Data Population

A Cohesity internal service performs ETL (Extract, Transform, Load) operations at a configurable frequency. During ETL operations, the service issues API calls to get statistics from other services, and the collated metadata information is stored in the Custom Database. The bootstrap run of the ETL process pulls the entire data set to populate the Custom Database. In the subsequent runs of the ETL process, the data refresh is incremental and only the delta is stored.

Specifically, the types of data collected are:

- Details of objects and sources registered.
- Protection Groups configured
- History of Protection Group runs (archivals, replications, backup, and restore).
- Performance metrics of the cluster and the Protection Groups (IO, resource, and storage).
- Tenant and cluster (nodes and partitions) information.
- Backup schedule and Protection Policies.

The data refresh rate is a configurable parameter. Table 4 below lists the default frequency for different components.

Table 4: Data Population—Default Refresh Frequencies

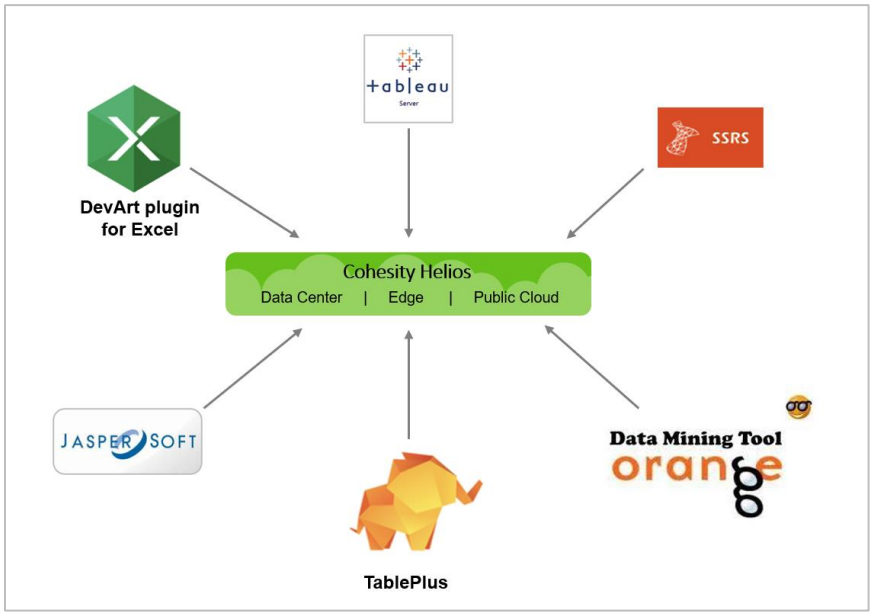
| COMPONENT | DEFAULT REFRESH INTERVAL |
|---|---|
| Object Protection status (stats on Incremental Protection Group runs) | 5 mins |
| Protection Group information (metadata on Protection Group runs) | 15 mins |
| I/O performance stats | 60 mins |
| Resource usage | 60 mins |
| Storage usage | 60 mins |

## Third-party Tools

You can use any reporting tool that supports ODBC/JDBC drivers to connect to the Postgres database, including:

- DevArt plugin for Excel
- Tableau for Desktop
- Microsoft SSRS
- Orange
- TablePlus
- Jaspersoft

Figure 2: Third-party Tools—Connect Any ODBC/JDBC-compatible Tools



# Generate a Sample Custom Report

Now, let's explore a typical scenario and generate a sample custom report. Typical use cases for custom reporting include:

- A backup administrator who needs to generate a detailed audit/compliance report on backup successes and failures over the past 365 days. (This use case applies to replication statuses and archival statuses.)
- A service provider who needs a chargeback report to charge tenant organizations based on their amount of storage consumed.

In this example, we generate a timeline view of the status of Protection Groups over a specific period of time using Cohesity Platform's Custom Database. For this example, we used Tableau for Desktop, but most ODBC/JDBC-compatible reporting tools should work similarly. First, we look at the steps involved in fetching the endpoint details, then we connect to the Custom Database, populate the data source, visualize, and finally, publish.

## Authentication

To establish a connection from a third-party reporting tool to reach the Custom Database, specify the endpoint details in Table 5:

Table 5: Database Authentication—Default Values

| PARAMETER | DESCRIPTION | DEFAULT VALUE |
|---|---|---|
| Hostname or IP address | Node IP or Hostname | n/a |
| Port number | Port number that accepts DB connections | 27999 |
| DB username | Username to connect to the database | reporting_read |
| DB password | Password for the DB user | n/a |
| Target database name | Target database to connect | postgres |

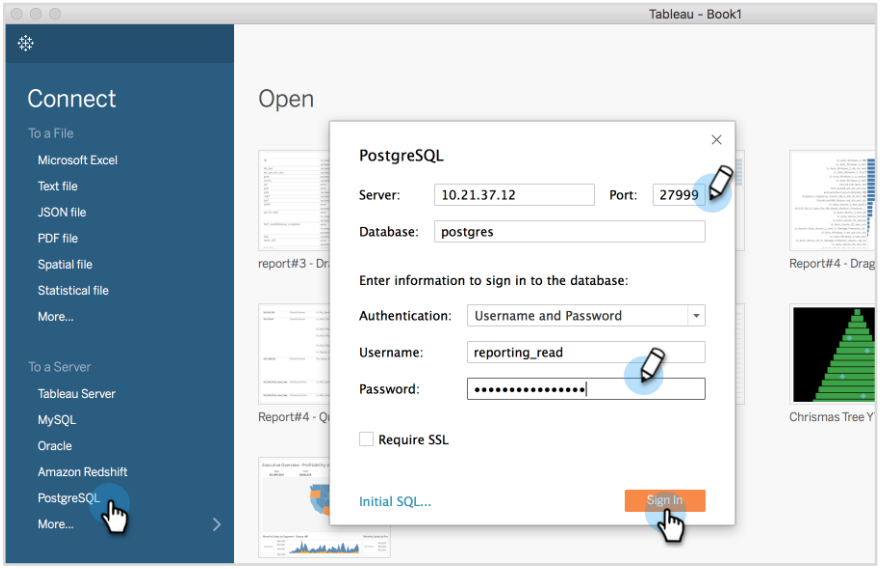
To obtain the above endpoint details, log in to Cohesity Platform as administrator and issue the following command from the Cohesity CLI (command-line interface):

```
[admin@server-1 ~]$ iris_cli custom-reporting db
```

## Connect to the Custom Database

We start by establishing a connection to the Custom Database from Tableau.

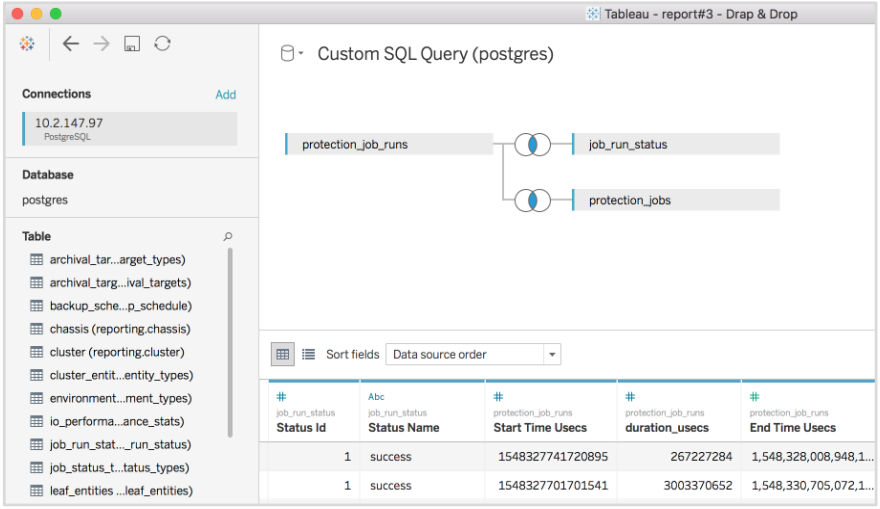
1. From the Tableau home page, select **Connect** > **To a Server** > **PostgreSQL**.
2. Enter the **Server** IP address, **Port** (27999), **Database** (postgres), and login credentials, then click **Sign In**.



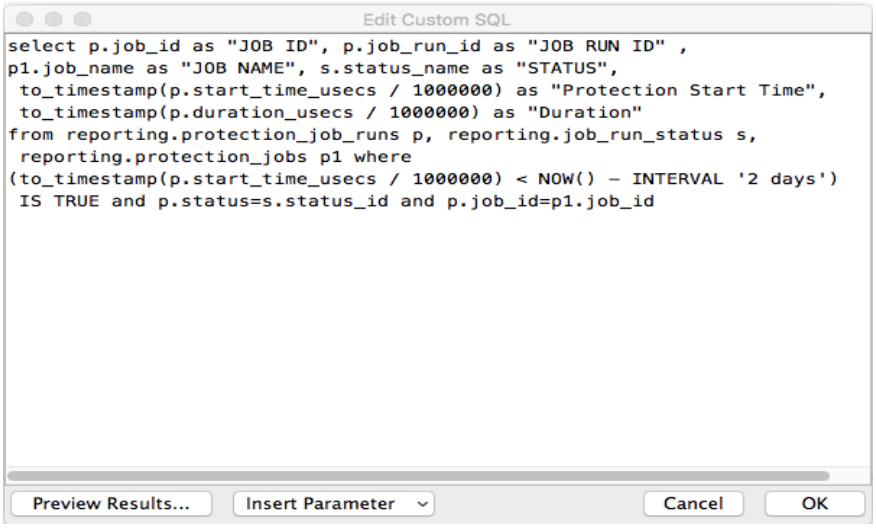

## Populate the Data Source

Now you need to populate the relevant data that is required to generate the report. This involves selecting a subset of the data upon which to build the reports. In Tableau, you can do so in two ways:

- **Drag & Drop Method**: There is an option to drag the tables into the canvas and connect the tables using joins.

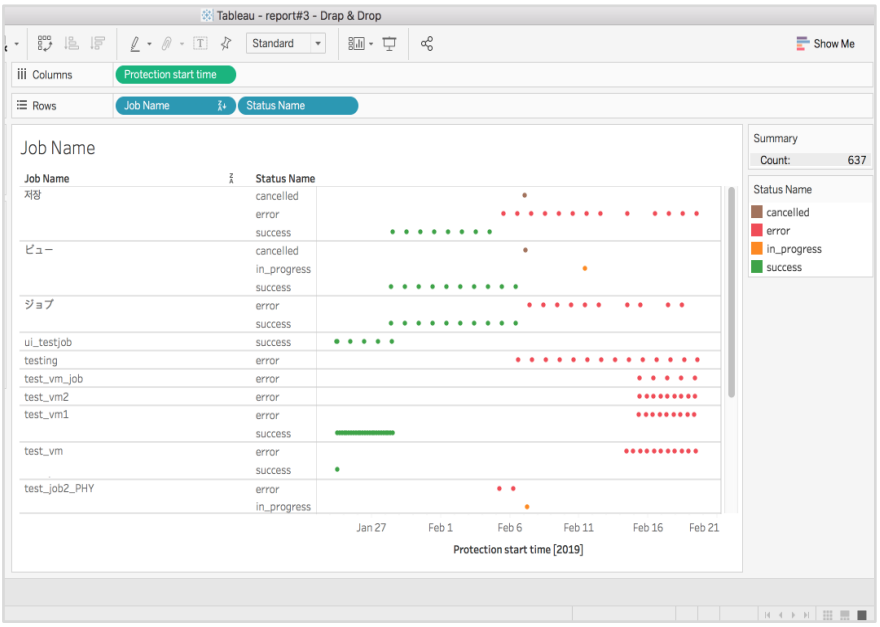


- **Custom SQL**: Tableau and many other third-party tools also allow you to write custom SQL to populate data.




## Visualize

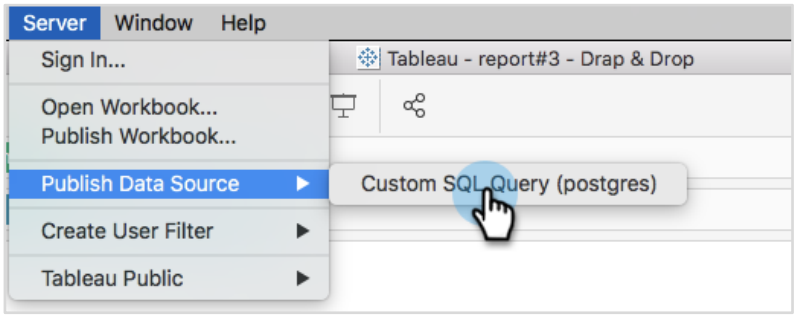
Tableau provides a wide variety of visualization options. In this example, we used a Gantt chart to generate a timeline view of data. The **Job Name** and **Status Name** are plotted against the **Protection start time**.



## Publish

Once you have generated the report you need, you can publish it into a Tableau server.

1. From the Tableau menu, select **Server** > **Publish Data Source** > **Custom SQL Query (postgres)**.




Most third-party reporting tools also offer options to schedule reports at specific times or intervals, and can be set up to distribute to a specific audience automatically.

# Summary

As organizations rely increasingly on good business analytics and intelligence to grow, they need reporting over long periods of time, fine-grain visibility and aggregated roll-ups, personalized filtering, and visualizations. To address these needs, Cohesity provides both built-in reports and custom reporting features.

Custom reporting on a single cluster is addressed via database-native access to underlying metadata, whereby you can integrate any third-party business intelligence tool to report the data you manage in Cohesity Platform.

Helios provides custom reporting across multiple clusters, as well as a one-stop, UI-based approach for all customization requirements.

With Cohesity Platform’s built-in and custom reporting capabilities, you can meet almost any reporting requirements in any context.

# Your Feedback

Was this document helpful? Send us your feedback!

# About the Authors

Surya Swaminathan is Technical Marketing Engineer at Cohesity. In his role, Surya focuses on the cloud, manageability, and disaster recovery.

Other essential contributors include:

- Adaikkappan Arumugam, Sr. Manager Technical Marketing & Solution Engineering
- Bart Abicht, Senior Technology Writer and Editor
- Yu-Shen, Sr Product Manager

# Document Version History

| VERSION | DATE | DOCUMENT HISTORY |
|---|---|---|
| 1.0 | Mar 2019 | First release |
| 2.0 | Oct 2020 | 6.5.1 updates |
| 2.1 | June 2022 | Rebranding updates |

# ABOUT COHESITY

Cohesity radically simplifies data management. We make it easy to protect, manage, and derive value from data -- across the data center, edge, and cloud. We offer a full suite of services consolidated on one multicloud data platform: backup and recovery, disaster recovery, file and object services, dev/test, and data compliance, security, and analytics -- reducing complexity and eliminating mass data fragmentation. Cohesity can be delivered as a service, self-managed, or provided by a Cohesity-powered partner.

Visit our website and blog, follow us on Twitter and LinkedIn and like us on Facebook.