



Sisense V6.6 Documentation

Table of Contents

Sisense V6.6 Documentation.....	1
Introduction	11
<i>Basic Concepts and Terminology.....</i>	12
<i>Sisense Architecture Internals.....</i>	21
<i>Sisense Architecture.....</i>	23
Managing Data.....	23
Interacting with Data	25
Communication Ports	27
<i>Minimum Requirements and Supported Platforms</i>	31
<i>Installing Sisense.....</i>	34
Downloading Sisense	34
Installing Sisense	35
The Installation Process	35
Setting Up Your Production Environment.....	38
Customizing the Installation	40
<i>Performing a Silent Installation</i>	43
Backing Up and Migrating a Sisense Installation	45
Backing Up a Sisense Installation.....	45
Migrating a Sisense Installation	46
<i>Upgrading Sisense</i>	50
Before Your Upgrade	50
Installing the Latest Version.....	50
Downgrading Sisense	52
<i>Launching Sisense.....</i>	54
Managing Data.....	56
<i>Introduction</i>	56
ElasticCube Advantages.....	56
ElasticCube Manager Environment	57
<i>ElasticCubes - Technical Overview.....</i>	58

Introduction	58
ElastiCube Columnar Database	58
Just-In-Time, In-Memory Processing	60
Designed with Standard Hardware in Mind.....	61
Unified Analytics Engine	61
Compliant with Industry Standards	62
<i>Working with the ElastiCube Manager</i>	63
<i>Sisense Live Connect</i>	65
Adding Live Connections.....	66
Working with Data Sets	68
<i>Connecting to Data Sources</i>	73
Introduction to Data Sources.....	73
Connecting to Amazon RDS	74
Connecting to Amazon RedShift	74
Connecting to Access	77
Connecting to Bing.....	79
Connecting to Box.....	86
Connecting to CSV.....	91
Connecting to DynamoDB.....	95
Connecting to Exact	101
Connecting to Microsoft Excel	111
Connecting to Facebook	115
Authenticating through Facebook	120
Connecting via ODBC Drivers	140
Connecting via OLE DB Drivers	143
Connecting to Gmail	147
Connecting to Google Adwords	153
Connecting to Google Analytics	156
Connecting to Google BigQuery.....	160
Connecting to Google Spreadsheets.....	174
Connecting to Hadoop via Hive	177
Connecting to HubSpot.....	195
Connecting to MailChimp	201
Connecting to Microsoft Active Directory	210

Connecting to Microsoft Dynamics CRM	218
Connecting to MongoDB.....	244
Connecting to MySQL	256
Connecting to Oracle	259
Connecting to PostgreSQL	262
Connecting to Quickbooks Desktop.....	265
Connecting to QuickBooks Online	280
Connecting to SalesForce.....	287
Connecting to ServiceNow.....	291
Connecting to Sisense ElastiCubes.....	297
Connecting to SnowFlake	307
Connecting to Splunk.....	312
Connecting to Stripe	315
Connecting to SQL Server	320
Connecting to Twitter	323
Connecting to Xero Accounting	330
Connecting to YouTube.....	338
Connecting to YouTube Analytics	344
Connecting to Zendesk	349
Connection String Parameters	354
 <i>Working with Data</i>	364
Searching for Tables and Fields.....	364
Creating and Removing a Relationship between Tables.....	364
Previewing Data from a Table.....	366
Managing Tables and Fields	367
Handling Relationship Cycles	370
Relationship Cycles	370
Many-to-Many Relationships.....	373
Chasm and Fan Traps	386
 <i>Transforming and Enriching Data</i>	390
Adding a New Custom Field	390
Defining and Editing Field Formulas	391
Adding a New Custom Table	392
Defining and Editing SQL Statements	393
Previewing Results	394

SQL Reference.....	395
Function Reference.....	398
Building ElastiCubes for Advanced Business Scenarios.....	411
<i>Building the ElastiCube</i>	436
<i>Making Changes to an Existing ElastiCube</i>	438
<i>Build Settings and Data Accumulation Behavior</i>	439
Build Options	440
Field Build Behavior	442
Table Build Behavior	445
Build Options Working Together with Table and Field Behavior	446
Accumulative Build Support.....	447
<i>Managing ElastiCubes</i>	450
Sisense Server Console	450
Scheduling Data Loads	452
Importing and Exporting ElastiCube Data	453
Creating Dashboards	455
<i>Creating a New Dashboard.....</i>	455
<i>Adding Widgets to a Dashboard.....</i>	458
Using the Widget Wizard	458
Data Browser	462
Using the Widget Designer	465
<i>Adding Text Widgets.....</i>	479
Examples	479
<i>Additional Widget Design Options.....</i>	480
Selecting Colors in Widgets.....	480
Selecting a Specific Color – Color Tab	481
Selecting a Range of Colors – Range Tab	481
Defining Conditional Coloring – Conditional Tab	482
Renaming Fields in Widgets	483
Adding a Description to a Widget	484
Formatting Numbers in Widgets.....	485
Changing a Widget’s Visualization	487

Combining Two Types of Visualizations in a Widget.....	487
Adding Drill Hierarchies to Widgets.....	488
Creating a Continuous Chart with Missing Date Values	489
<i>Managing Widgets</i>	491
Editing a Widget.....	491
Copying a Widget.....	491
Deleting a Widget	493
<i>Customizing the Dashboard Layout</i>	494
Layout Mode and View Mode.....	494
Adding Dashboard Columns.....	495
Moving a Widget on the dashboard	495
<i>Placing Widgets Side-by-Side</i>	497
Resizing Widgets	497
Adding Widget Titles.....	498
Working with Ticker Widgets.....	498
Changing the Dashboard's Color Palette	499
<i>Filtering Dashboards and Widgets</i>	502
Creating Dashboard Filters	502
Creating a Dashboard Filter	503
Editing and Deleting a Dashboard Filter	516
Creating Widget Filters	518
Creating Dependent Filters	521
Configuring How Filters Affect the Dashboard and Widgets	523
Limiting Filters to Specific Values with Background Filters.....	527
Locking Filters	529
Restoring Filters	530
Switching Filters On and Off	531
Saving Your Default Filters View	531
<i>Using Formulas</i>	533
Using the Formula Editor	534
Creating and Editing a Formula.....	536
Reusing Formulas.....	539
Using Quick Functions.....	540
Using R in Formulas	543



<i>Using R functions in Formulas</i>	545
<i>Example 2: R Logistic Regression via Sisense</i>	550
Creating Formulas Based on Criteria and Conditions (Filters)	552
Building Formulas with Functions.....	556
Function Reference	561
Changing a Dashboard's ElastiCube	581
Embedding Dashboards and Widgets	583
 Using Dashboards	 589
<i>Viewing Dashboards</i>	590
<i>Interacting with Filters as a Viewer</i>	591
<i>Drilling Down in a Widget</i>	594
<i>Making Selections in a Widget</i>	596
<i>Zooming In/Out in Charts</i>	597
<i>Resizing the X-AXIS to Expose Labels</i>	598
<i>Creating PDF Reports</i>	599
<i>Customizing PDF Reports</i>	600
<i>Downloading a Dashboard as an Image</i>	603
<i>Exporting Widgets to CSV/Excel</i>	605
 Downloading a Widget as an Image	 606
 Managing and Sharing Dashboards	 608
<i>Organizing Dashboards</i>	608
Organizing Dashboards in Folders	608
<i>Sharing Dashboards</i>	613
Working with Shared Dashboards	614
Publishing Dashboards to Recipients	616
Updating a Shared Dashboard	616
Changing Ownership of a Dashboard	617
Sending Dashboard Email Reports.....	619



Sisense Pulse and Alerting	622
<i>Sisense Alerts</i>	623
Creating Data Alerts	624
Creating Build Alerts	627
Advanced Settings	630
<i>Sisense Pulse</i>	644
Adding Tiles to Sisense Pulse	645
Managing Tiles in Sisense Pulse	646
Duplicating Tiles in Sisense Pulse	646
Deactivating Your Notifications	647
Deactivating Notifications for All Users	648
Deleting Tiles in Sisense Pulse	649
Removing Sisense Pulse	649
Administration	652
<i>Managing Dashboard Users</i>	652
Adding a New User	652
Adding Multiple Users	654
Editing an Existing User	655
Deleting a User	655
<i>Managing User Groups</i>	656
Accessing User Group Management	656
<i>Adding a New User Group</i>	658
<i>Adding Users to an Existing User Group</i>	660
Editing an Existing User Group	661
Deleting a User Group	661
Managing User Sessions	662
Customizing User Roles	663
Integrating Active Directory	674
Introduction to SSO	682
<i>SSO via JWT</i>	683
SSO Authentication Flow	683
Configuring SSO in Sisense	685



Enabling SSO in Sisense.....	685
Creating a JWT	687
return_to URL	688
Configuring Sisense as a Sub-Domain with SSO.....	689
Logging Users Out	692
<i>SSO via SAML 2.0</i>	694
Enabling SSO via SAML in Sisense	697
Sisense Default Role Set-Up.....	698
<i>ElastiCube Management</i>	701
Adding and Removing ElastiCube Servers	701
Querying ElastiCubes during Builds	704
Overcoming Build Failures in ElastiCube Sets	720
Updating ElastiCubes within an ElastiCube Set	722
<i>High Availability in Sisense</i>	725
Overview	725
Creating a High Availability Environment	730
Distributing ElastiCube Builds to Query Nodes.....	730
Replicating MongoDB	744
Creating a Replica Set	748
Connecting Sisense to the Replica Set	750
<i>Routing Queries in ElastiCube Sets</i>	752
<i>Security</i>	755
Sisense Security Architecture	755
System Level Security	755
Object Level Security	757
Data Level Security.....	757
Security Levels	758
ElastiCube Server and ElastiCube Security.....	758
Data Security.....	762
<i>White Labeling Sisense</i>	771
Rebranding Sisense	771
White Labeling Configuration Options.....	773
Rebranding Sisense Automated Emails	779



Replacing Sisense Automated Emails	783
Applying Sisense Tokens	787
<i>Displaying Custom URLs</i>	791
<i>Internationalization and Localization</i>	791
Changing the Sisense Web Application's Language.....	793
What has been Translated?	793
Changing your Default Language	794
Embedding Customized Languages	795
Translating Sisense Metadata.....	797
<i>System Configuration</i>	805
<i>Setting Up a Custom Email Server</i>	806
<i>Managing Drill Hierarchies</i>	810
<i>Migrating Sisense across Environments</i>	813
Copying Dashboards between Sisense Servers.....	815
Viewing Dashboards on Mobile Devices.....	819
<i>Using the Sisense Mobile App</i>	819
Supported Phones and Operating Systems	819
Get Sisense Mobile	819
Working with the Mobile App.....	821
<i>Using Sisense in a Mobile Phone or Tablet Browser</i>	823

Introduction

This guide contains the complete product documentation for Sisense.

Every Sisense user is assigned one of the following roles, Administrator, Designer, or Viewer. You should view the documentation that is relevant for your role.

Administrators are responsible for creating and managing users and user groups, managing connected ElastiCubes, defining the web configuration, connecting to Active Directory, and embedding Sisense analytics (OEM).

If you are an Administrator, you should review the following chapters:

Administration

Designers are responsible for building ElastiCubes and creating dashboards.

Your company may have a lone Designer or multiple Designers responsible for different aspects of Depending on your company's structure, Designers may split their responsibilities according to two types of Designers. The first type, Data Modelers, is responsible for managing your data, for example, by connecting data sources to Sisense and designing your ElastiCube schemas. The second type of Designer, Dashboard Designers, is responsible for creating dashboards and publishing them to other users.

If you are a Designer, you should review the following chapters:

Managing Data (Data Modelers)

Creating Dashboards (Dashboard Designers)

Sharing Dashboards (Dashboard Designers)

Data Connectors (Data Modelers)

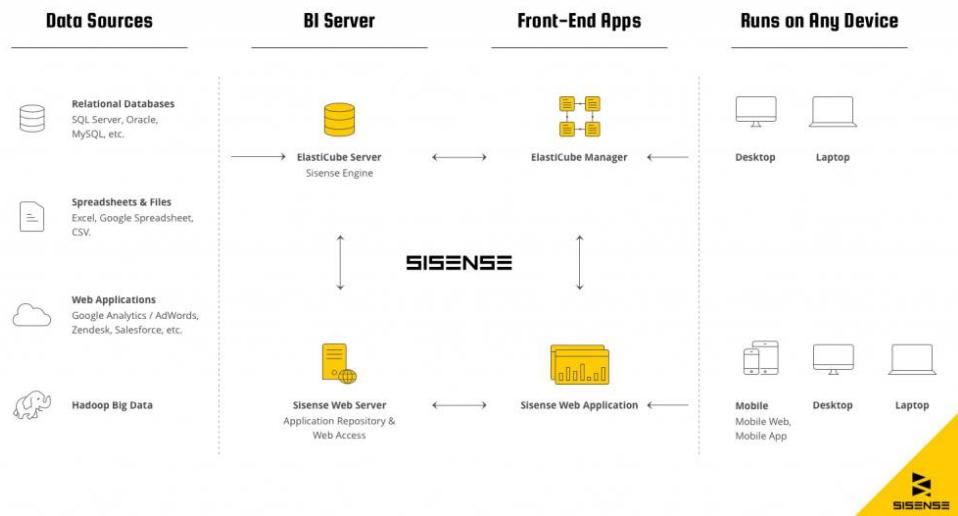
Viewers can view, explore, drill down, make selections and filter the dashboards that Designers share with them.

If you are a Viewer, you should review chapter Using Dashboards.

Basic Concepts and Terminology

Sisense Architecture

The Sisense system is comprised of the following components.



ElastiCube Manager

The ElastiCube Manager is a visual environment in which you create ElastiCubes. The ElastiCube Manager enables you to structure and import multiple data sources, create relationships between data, and perform ETL (Extract, Transform, and Load) processes to prepare data for analysis and visualization. For more information, see ElastiCube Manager.

ElastiCube Server

The ElastiCube Server is installed locally on your computer and enables access to ElastiCubes. Both the Sisense Web Application and the ElastiCube Manager query the ElastiCube Server and receive results.

Sisense Server Console

The Sisense Server Console provides administration functions for managing ElastiCubes on the ElastiCube Server.

Sisense Web Application

The Sisense Web application is an interactive web application that provides the user interface in which users can design, share, view and explore dashboards. The Sisense Web application runs in popular web browsers and enables access to the Sisense server. Dashboards can also be viewed on mobile devices.

Sisense Web Server

The Sisense Web Server is installed locally on your computer and hosts the Sisense Web application.

Designers and Viewers

Note: In previous versions, Designers and Viewers were known as Contributors and Consumers.

Sisense Administrators can access the management settings. Administrators can create and manage users and user groups, manage connected ElastiCubes, define the web configuration, connect to Active Directory, and more.

Sisense Designers can create, design, edit and share dashboards.

Sisense designers determine whether the user with who they share a dashboard has editing rights (is a designer) or only viewing rights (viewer).

Sisense Viewers can view, explore, drill down, make selections and filter the dashboards that designers share with them. Viewers only require a standard web browser; no plugins or downloads. They cannot create new dashboards or edit existing ones.

Feature	Admin	Designer	Viewer
Server and User Management	Yes	No	No
Create New Dashboard	Yes	Yes	No
Share Dashboard	Yes	Yes	No
Create New Widget	Yes	Yes	No
Edit, Modify, Export Widget	Yes	Yes	No
Add Dashboard Filters	Yes	Yes	No
Create and Manager Folders	Yes	Yes	No
Drill in Dashboards	Yes	Yes	Yes
Access Dashboards on Mobile	Yes	Yes	Yes
Receive Email Reports	Yes	Yes	Yes
Export Dashboard to PDF	Yes	Yes	Yes

ElastiCubes

ElastiCubes are Sisense's proprietary super-fast data stores.

ElastiCubes are specifically designed to withstand the extensive querying typically required by business intelligence applications.

ElastiCubes enables you to mash up terabytes of data from a variety of sources, for example:

Traditional relational databases, such as SQL Server, MySQL or Oracle.

File-based data sources, such as spreadsheets (Excel) and CSV files.

Online web services, such as Salesforce.com, Google AdWords, Google Analytics, Zendesk and more.

ElastiCubes are based on Sisense's proprietary In-Chip technology, which leverages Columnar and In-Memory database technologies in contrast to relational databases. Read more about the technology behind ElastiCube in Sisense's [technology section](#).

ElastiCubes are created and managed in the Sisense ElastiCube Manager.

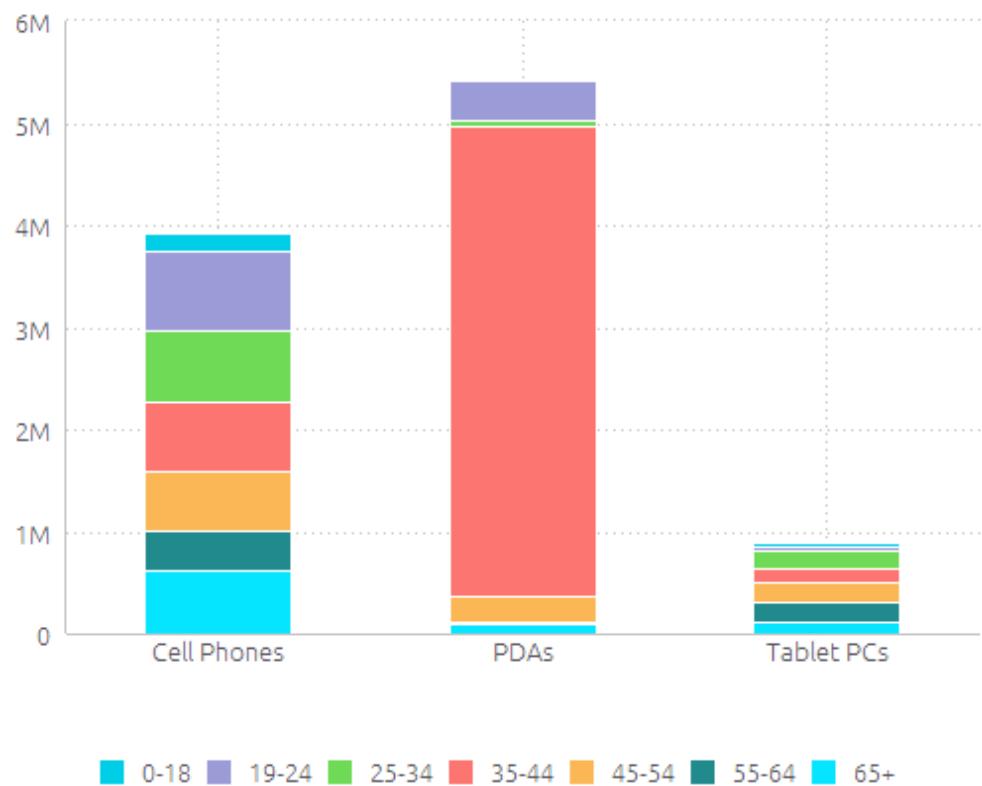
ElastiCube Sets

Sisense ElastiCube Sets are collections of ElastiCubes with identical schemas that support high availability by allowing you to query running ElastiCubes within the ElastiCube Set while other ElastiCubes are in a building state.

Widgets

Each widget is a dynamic visualization of data. You pick the type of data to appear in a widget and you pick the type of visualization (chart type). A few examples of widgets are displayed below:

Column Chart Widget



Indicator Widget

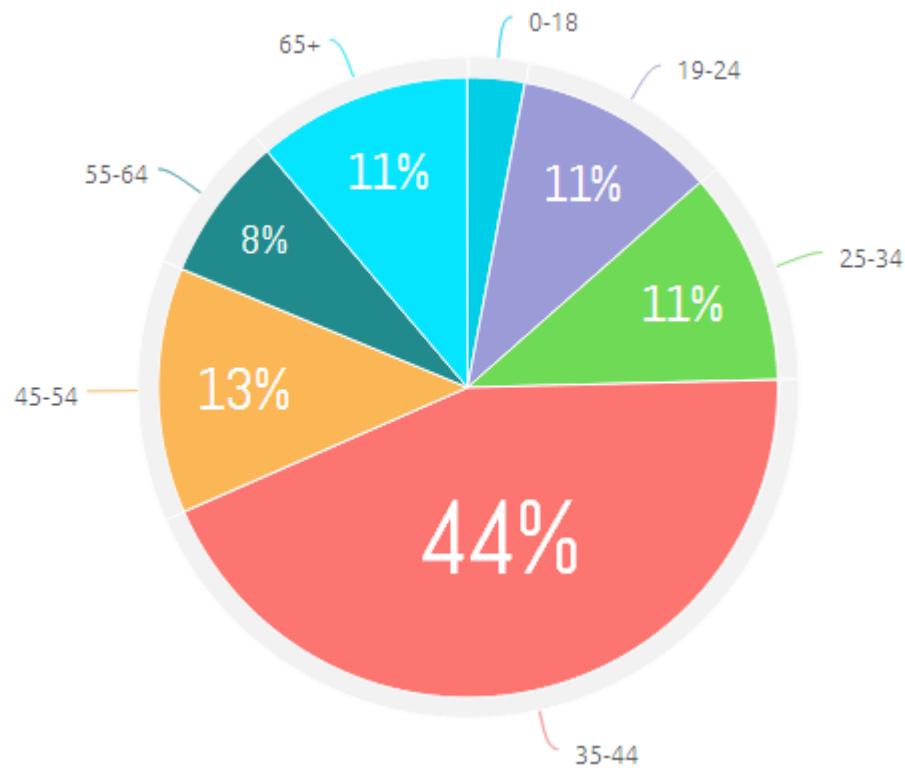
GROWTH OF VISITORS

13.2%

of Visitors 220K



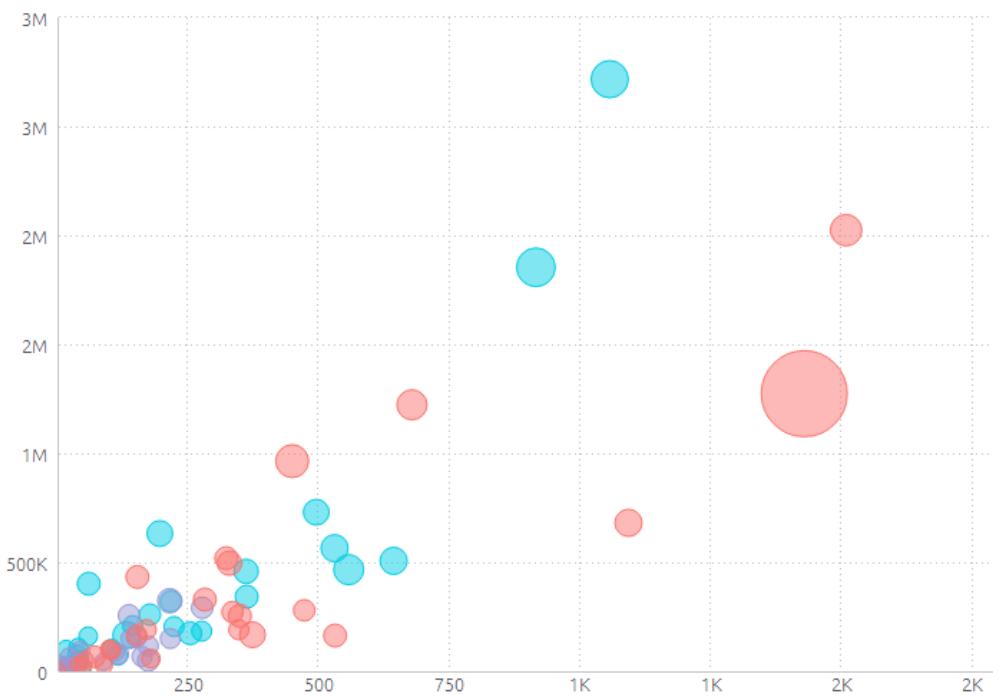
Pie Widget



Pivot Widget

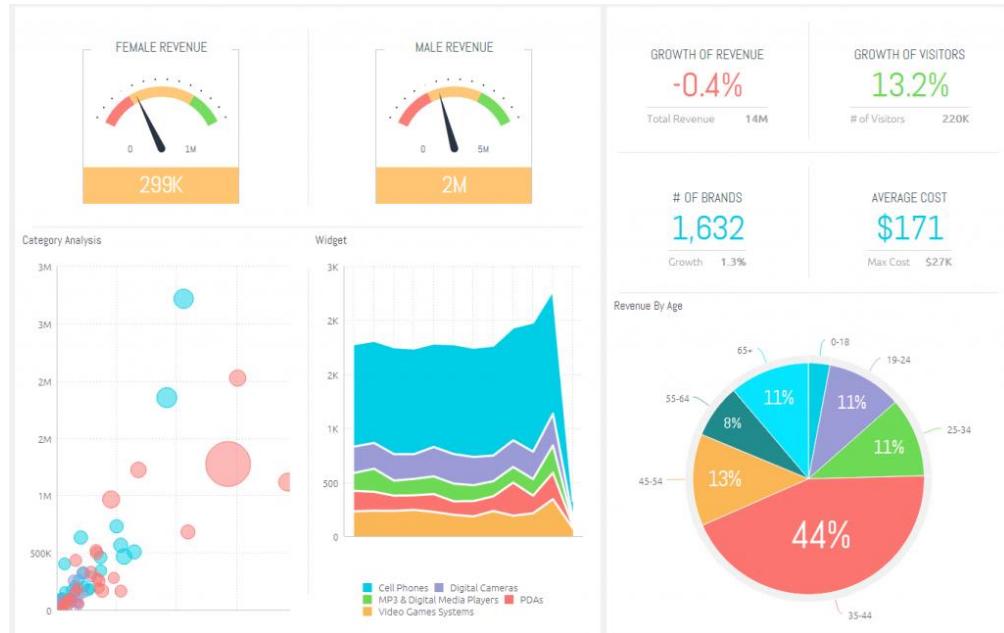
Category	Female		Male	
	Total Revenue	Total Quantity	Total Revenue	Total Quantity
Apple Mac Desktops	1,151	4	26,071	47
Apple Mac Laptops	5,310	18	50,717	110
Calculators	609	15	3,147	38
Camcorders	2,790	23	26,890	124
Camera Flashes	1,168	12	16,140	115
Car Amplifiers	734	6	4,619	32
Car Speakers and Subwoofers	612	9	6,968	90
Cell Phones	94,323	638	605,011	2,122
Digital Cameras	34,045	153	216,707	669
DVD Players	1,488	23	5,309	85

Scatter Chart Widget



Dashboards

A dashboard is a collection of one or more widgets that visualize the data that you select and design.



You define which widgets appear in the dashboard, their design, how they are organized and the filtering of the data that appears.

You can access your own dashboards and dashboards others shared with you.

Sisense allows you to create as many dashboards as you need. No limitations! When you create a dashboard, you are the dashboard's owner. Owners can share dashboards that they have created with other Sisense users (Viewers and Designers).

Sisense provides a variety of built-in automatic dashboard sharing features. You can easily share a dashboard with others in the Sisense environment or have a dashboard automatically delivered to your chosen recipients by email (on a scheduled basis or upon each data update).

Fields

Widgets are composed of fields that represent the data in your data sources. You create widgets by simply selecting from the fields displayed in a Data Browser, which appears in various places across the product. Each field represents a column of data in the ElastiCube.

ECOMMERCE X

fx Type to search for fields

Commerce

- [A] Age Range
- [#] Brand ID
- [#] Category ID
- [A] Condition
- [#] Cost
- [#] Country ID
- [#] Date
- [A] Gender
- [#] Quantity
- [#] Revenue Sum More...
- [#] Visit ID

Brand

- [A] Brand
- [#] Brand ID

When designing widgets, fields can be categorized into two groups:

Numeric Fields: Numbers (quantified data), such as salaries, sales, scores, number of clicks and so on. This is data that you may want to aggregate or calculate. For example, the sum of sales or the average of costs.

Descriptive Fields: Items used to label and categorize, such as Products, Locations, Categories and so on.

Generally, widgets combine both these types of fields.

For example, to show the sales of a product over time, **Sales** is the numeric data and both **Products** and **Time** are descriptive data.

Sisense Architecture Internals

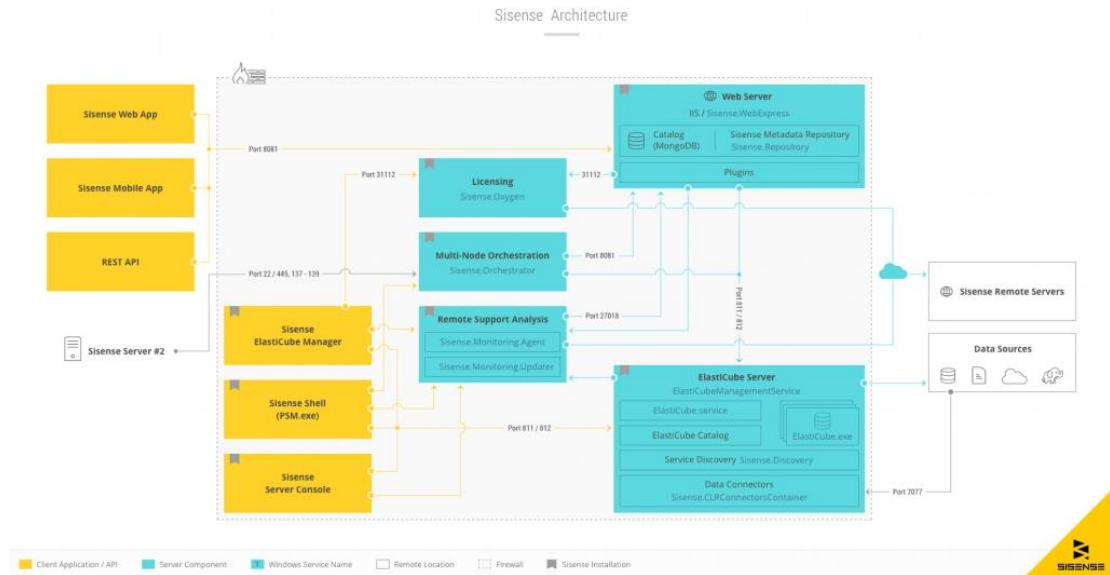
This page provides an advanced description of Sisense's architecture for IT managers and Administrators responsible for supporting Sisense in their organization.

When working with Sisense, there are two possible types of deployments. The first type is a **Single deployment** in which Sisense is installed on a single server. Single deployments can be leveraged to support single integrations or as proof of concept integrations for larger deployments.

The second type is a **Clustered deployment** where Sisense is installed on multiple machines, but only part of the components are active on each one. This type of configuration is used to support heavier traffic loads and high availability.

The Sisense full-stack solution is provided in a single installation process. After installing Sisense on a server, you will have access to all of the Sisense functionality.

The diagram below describes the main components of a Sisense Installation and how Sisense supports a full-stack solution:



In this diagram, the yellow components specify Sisense client applications or APIs. The blue components specify components of the Sisense Server.

The Windows service and application names that you can see in the Windows Task Manager under the **Services** or **Processes** tabs appear in a dark blue font.

When Sisense is installed on a single server, the dotted line indicates the boundaries of the server firewall. The lines between the components indicate communication between them, and its direction. When the Sisense Server listens on a specific port for inbound traffic, the relevant ports must be open in the firewall for the communication to succeed.

Sisense Architecture

Sisense supports a full stack solution from data, such as transaction details, user information, and machine-generated data, to dashboards accessible by viewers. There are two sides to this solution, the management of data and the analytic side where users can interact with the data.

This section provides a description of the Sisense architecture as part of the full stack solution, from data sources to dashboards.

Managing Data

The Sisense Server is responsible for managing data. Before you can begin to manage data however, Sisense communicates with remote servers for importing data sources, licensing information, tracking and similar activities.

Sisense Remote Servers

Sisense remote servers are used for license management, tracking and software updates.

Data Sources

Data sources may be files or databases located on servers on a local network, or remote locations and web services such as SalesForce and Google AdWords. Sisense supports a variety of data sources through native connectors, customer REST connectors, and ODBC drivers. These connectors are used to import data into the ElastiCube Server.

Sisense Server

The Sisense Server manages data and supports the Sisense Web Server, which hosts the Sisense Web Application. In a Single deployment, the Sisense Server resides locally on your machine. In a Clustered deployment, the Sisense Server can reside on one of your machines or across many machines depending on your needs. For example, if you want to support a high availability environment, you may install the ElastiCube Server on one machine, the web server on another, and the Multi-Node Orchestration server on another machine.

The Sisense Server is comprised of the following components:

Licensing: This service checks and validates licensing information.

Multi-Node Orchestration: The Sisense Orchestrator Service is an automated service that can be configured to synchronize builds across the ElastiCube Set.

Remote Support Analysis: This service is responsible for controlling logging, monitoring and automatic software updates.

Web Server: The Sisense Web Server is installed locally on your computer and hosts the Sisense Web application.

ElastiCube Server: The ElastiCube Server manages the process of importing data from various data sources and ETL processes. For more information, see ElastiCube Server below.

ElastiCube Server

The ElastiCube Server supports the management of ElastiCubes and the data they contain. The Sisense Server also supports the Web

Server, which queries the ElastiCube Server where the ElastiCubes are hosted.

Within the ElastiCube Server are the following components:

ElastiCube Catalog: The catalog maintains a list of ElastiCubes contained within the ElastiCube server.

Data Connectors: The Data Connectors service is responsible for managing both native and 3rd party connectors.

The ElastiCube Server supports the following applications:

Sisense ElastiCube Manager: A Windows client application for managing data sources. The ElastiCube Manager provides the GUI for creating ElastiCubes, including importing data sources, preparing them, and building the ElastiCubes.

Sisense Shell (PSM.exe): A command line interface for managing ElastiCubes, such as building an ElastiCube, attaching and detaching ElastiCubes from a server, updating data sources, etc. The Sisense Shell can also be controlled programmatically.

Sisense Server Console: A Windows application that controls Sisense servers, including attaching ElastiCube folders to Sisense servers and starting and stopping ElastiCubes.

Interacting with Data

After you have created your ElastiCubes, your users can interact with data through Sisense dashboards.

Dashboards are hosted on the Sisense Web Server.

Web Server

The Web Server provides access to Sisense dashboards and Sisense Admin screens for user management.

The Sisense Web Server hosts the Sisense Web application, which provides the user interface from which users can design, share, view and explore dashboards on the Sisense Web Application and the Sisense Mobile App.

Each dashboard created is maintained within the Web Server catalog, which contains metadata regarding users, groups, and dashboards.

There are three main client applications that communicate with the Web Server:

Sisense Web App: This Sisense Web Application is the client application for dashboard creation and access, and system administration.

Sisense Mobile App: This Sisense Mobile Application is an Android and Apple-device compatible application for viewing viewing dashboards across devices.

REST API: The Sisense REST API provides programmatic access to the Sisense web server functionality including managing ElastiCubes, user and group management, white labeling, user roles, reporting, and security.

Communication Ports

Sisense utilizes multiple communication ports. Some of the ports are used for communication between Sisense modules, and others are used for communication with remote locations.

When Sisense is installed as part of a Single machine deployment, ports used for communication between Sisense modules are considered internal ports, and do not need to be open in the firewall. When Sisense is installed in a Clustered deployment, where Sisense modules installed on different computers communicate with each other, ports used for communication between Sisense modules may need to be opened in the firewall. This depends on the specific deployment implemented in your environment.

The tables below lists the ports on which Sisense listens for communication.

Ports for External Communication

The following table list ports used for communication with external locations. These ports should be open on your firewall to allow for this communication.

Module	Port #	Protocol	Description/Purpose
Web Server	8081(*)	HTTP	Sisense Web Server
Sisense.Orchestrator	445, 137-139	SMB	Used in high availability deployments. Otherwise, not used.

Module	Port #	Protocol	Description/Purpose
			The Sisense Orchestrator Service generates inbound (query nodes) and outbound (build node) traffic.
Sisense.Orchestrator	22(*)	SSH	Used in high availability deployments. Otherwise, not used.
			In high-availability, it copies DB farm to query nodes.
			The Sisense Orchestrator Service generates inbound (query nodes) and outbound (build node) traffic.
OAuth2 server	7077	HTTP	Used by GenericREST connectors that use oAuth authentication. Otherwise, not used.

(*) Indicates that the port is configurable.

Ports for Internal and External Communication

The following table lists all ports in which the type of communication, internal or external, depends on your deployment. In a single deployment, Sisense communicates internally to other services. These ports do not need to be open on your firewall. In a clustered deployment, Sisense communicates across your deployment to other Sisense services. These ports should be open on your firewall to allow for this communication.

Module	Port #	Protocol	Description/Purpose
ElastiCube Management Service, QueryServiceTcpPort	812	TCP	In high-availability, it's the Elasticube query end point
ElastiCube Management Service, QueryServiceHttpPort	16162	HTTP	In high-availability, it's the Elasticube query end point
ElastiCube Management Service, ManagementServiceTcpPort	811	TCP	Elasticube Management Service endpoint
ElastiCube Management Service, ManagementServiceHttpPort	16161	HTTP	Elasticube Management Service endpoint
Oxygen	31112	TCP	
MongoDB	27018(*)	TCP	

(*) Indicates that the port is configurable.

Ports for Internal Communication

The following table lists all ports on which Sisense communicates internally. These ports do not need to be open on your firewall.

Module	Port #	Protocol	Description/Purpose
MontoriongMongoDB	27019	TCP	
Sisense.CLRCConnectorsContainer Server	8090	TCP	CLR container server: Data transport between EC and data sources
Sisense.JVMConnectorsContainer Server	Dynamically allocated	TCP	JVM container server: Data transport between EC and data sources

Module	Port #	Protocol	Description/Purpose
Sisense.JVMConnectorsContainer Management	8095	TCP	Container management - e.g. Adding/Removing/Updating connectors
Sisense.Discovery	2181	TCP	Discovery service for connectors (Zookeeper)
CustomREST MongoDB	27077(*)	TCP	Intermediate and cache db for CustomREST: connector results

(*) Indicates that the port is configurable.

Minimum Requirements and Supported Platforms

The following prerequisites and supported platforms are required for working with Sisense.

Supported Web Browsers

The Sisense Web Application runs in the following HTML5 supported browsers:

The Sisense Web Application also works in mobile phone and tablet browsers that support HTML5.

Supported Operating Systems

ElastiCube Server and ElastiCube Manager can be installed on the following 64-bit operating systems:

- ▶ Windows 7 and up
- ▶ Windows Server 2008 R2 and up

Capacity and Hardware Requirements

Sisense easily scales up to billions of records with typical query response times of split seconds.

This section suggests system requirements for various performance capacities of the ElastiCube Server. Actual capacity requirements are provided after consultation with a Sisense technical representative at support@sisense.com. Extreme scenarios may require additional resources.

Sisense Dashboard viewers only require an HTML5 compliant Web browser.

# of rows	< 10 users		10s of users		100s of users	
	RAM (GB)	Logical Cores	RAM (GB)	Logical Cores	RAM (GB)	Logical Cores
Up to 100K	8	4	8	4	8	8
up to 500K	8	4	16	4	16	8
up to 1M	8	4	16	4	16	8
up to 10M	16	4	16	8	32	8
up to 50M	16	8	32	8	32	16
up to 100M	25	8	32	8	32	16
up to 200M	32	8	32	8	64	16
up to 300M	32	8	64	16	64	16
up to 500M	64	16	128	16	256	32
up to 600M	128	16	128	32	256	32
up to 700M	128	16	128	32	512	32
up to 800M	128	16	256	32	512	32
over 800M	256	16	256	32	512	32

Supported Locales

The formats for dates, times and numbers in your dashboards are based on your computer's operating system or browser's locale settings (depending on your browser).

Locales from the following countries are supported:

- ▶ United States
- ▶ United Kingdom
- ▶ Israel
- ▶ Canada
- ▶ South Africa
- ▶ Australia
- ▶ Netherlands
- ▶ Germany
- ▶ Ireland
- ▶ Mexico
- ▶ France
- ▶ China
- ▶ Brazil

Notes

You can override the default locale settings for all users by updating a parameter via the Rest API. Click [here](#) to learn how to override default locale settings.

To use a locale file that is not included in the above list of countries, you can manually add a locale file in the following location:

|ProgramFiles|Sisense|PrismWeb|client|resources|base|localization
[Click here](#) to access locale files and view their codes.

Installing Sisense

The following sections list the prerequisites for installing Sisense, and explain how to install and customize your Sisense installation.

- ▶ Downloading and installing Sisense
- ▶ Setting Up your Production Environment
- ▶ Customizing the Sisense installation
- ▶ Performing a Silent Installation
- ▶ Migrating a Sisense installation
- ▶ Launching Sisense

Downloading Sisense

If you want to try Sisense for the first time, sign up for the free trial at Sisense.com. You will be prompted to download the installer.

Installing Sisense behind a Firewall

The standard installation requires internet access to download the necessary components. For machines behind a firewall, or without internet access, you can [download](#) and install the full self-contained installation file.

For detailed instructions, see [Installing Sisense Offline](#).

Installing Sisense

The entire Sisense solution can be installed from one file, either locally or in a central place in your organization within a couple of minutes.

All Sisense components are installed with a default configuration, as follows:

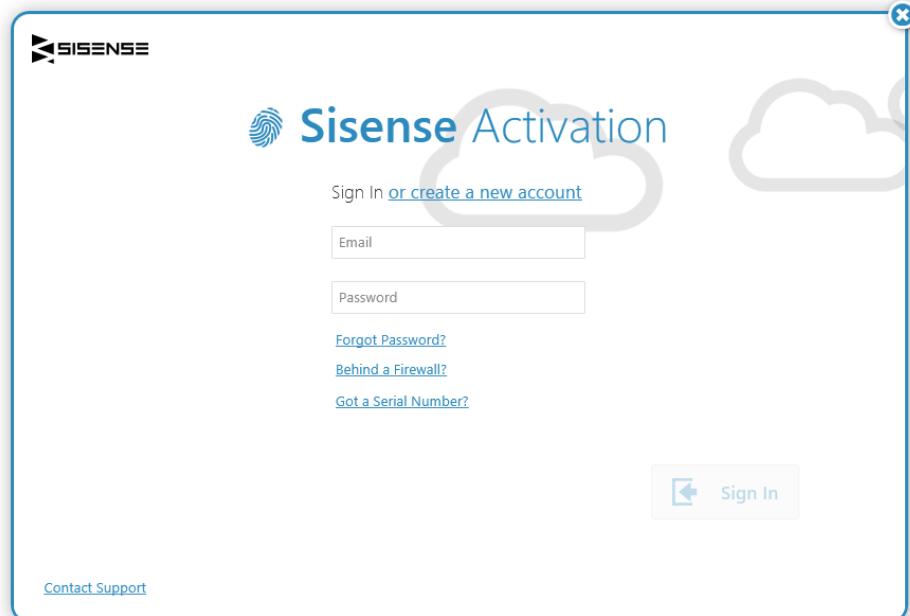
- ▶ Sisense Web Application
- ▶ ElastiCube project manager
- ▶ Sisense Server Console
- ▶ ElastiCube Server
- ▶ IIS Express as the web server hosting Sisense Web Application platform

Note: To install Sisense, you must have administrator privileges on the machine where you are installing Sisense and your environment must meet the minimum requirements.

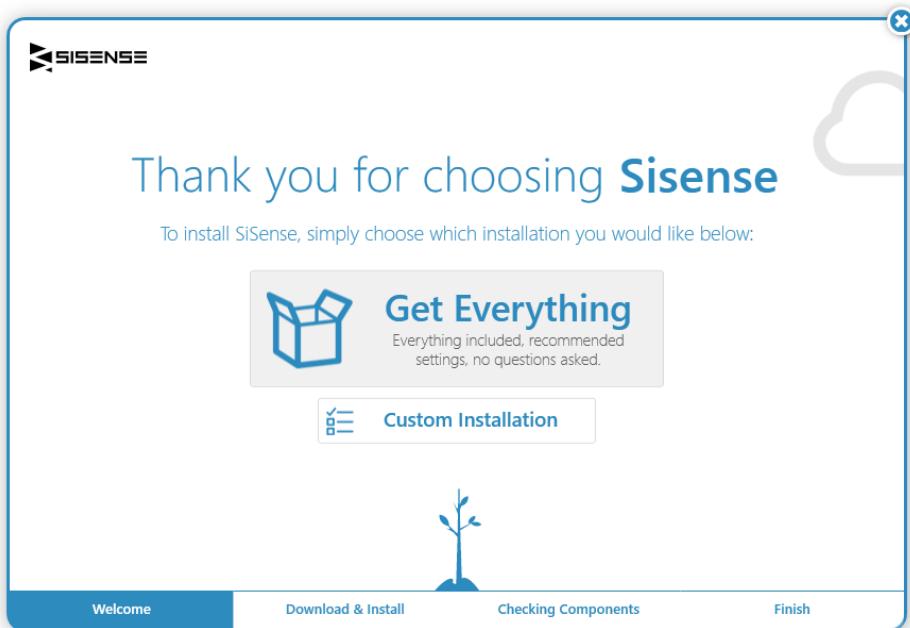
The Installation Process

To install Sisense:

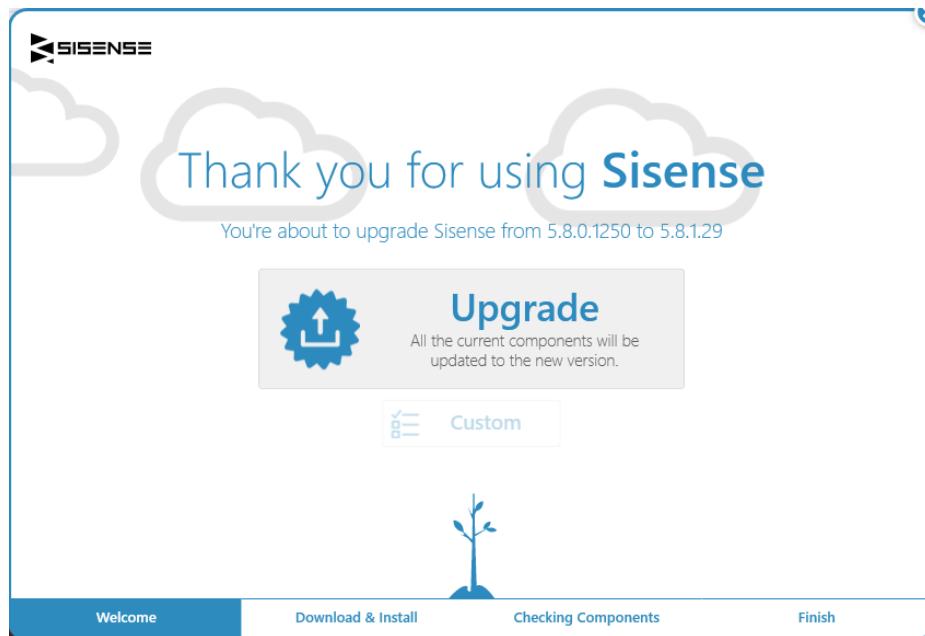
1. Open the downloaded executable file to run the installation.
2. If you are not logged into Sisense, enter your username and password to confirm licensing and begin the installation process. If you are logged in already, the activation screen is not displayed.



3. After entering your login details, select **Get Everything** to run the default installation:



4. If you are upgrading your version, click **Upgrade**.



See also:

If you need to customize your installation, see [Customizing the Installation](#). You can customize your installation during the first installation or after an installation, but not during an upgrade.

If you want to install Sisense on a remote machine, see [Performing a Silent Installation](#).

If you are migrating your installation to a different machine, see [Migrating an installation](#).

Setting Up Your Production Environment

Sisense recommends installing Sisense in a staging environment, where you can check all of your ElastiCubes and dashboards, as well as test different configurations and customization options.

When everything is working as expected, deploy Sisense in your production environment.

Providing Remote Access to the Sisense Web Environment

To maximize the Sisense experience, you will want to make your dashboards accessible to external users outside the organization's network. This requires the following configurations:

- ▶ Open TCP port (default is 8081) to access the Sisense Web Application internally.
- ▶ Ensure that your Sisense machine has a public IP address or domain name associated with a public IP for external access.
- ▶ Open port 443 if you are using SSL. Sisense recommends using SSL for secure connections or a VPN solution. For more about implementing SSL, see Setting Up SSL.

External users who were granted access can view dashboards by entering the machine's IP and the port number, for example: *xx.xxx.xx.xx:8081* or *my.company.com:8081*

Click [here](#) to learn more and see troubleshooting tips.

Providing Remote Access to the ElastiCube Manager

In some cases, you may want to provide external access to the ElastiCube Manager for designing and modeling your ElastiCubes. In

such cases it is necessary to consult your Sisense Success Manager to make sure your licensing terms allow this option.

In addition, make sure that ports 811, 812 on your Sisense machine are open.

Note: To avoid exposing your data, do not open ports 811/812 without the proper security measures.

Customizing the Installation

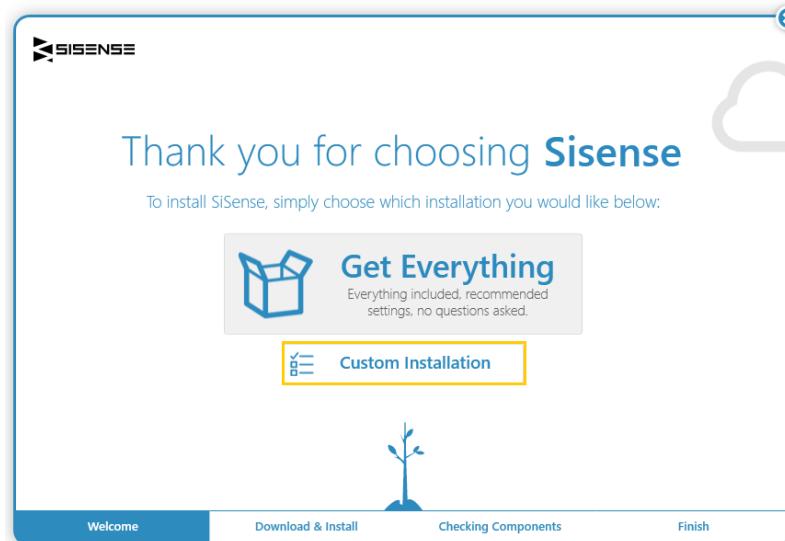
You can customize the settings for your web server application during the installation process or after Sisense is installed.

For example, you can configure your server to use the [Long index edition](#) of Sisense if your ElastiCubes are expected to contain a table with over 300 million rows.

Note: You cannot customize the settings during an upgrade process. If you upgraded Sisense and want to customize settings, refer to the post-installation option described below.

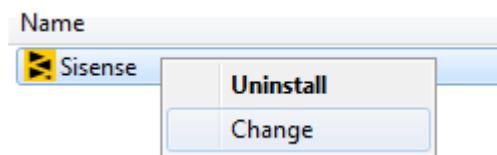
To access the customization options during the installation process:

- ▶ During the installation wizard process, click **Customize Installation**, and define the settings as described below.

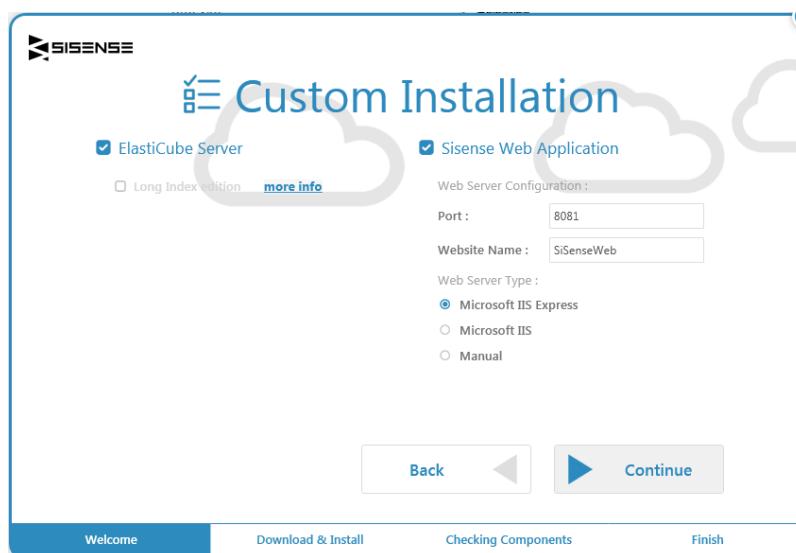


To access the customization options after you have already installed Sisense:

1. Open the Control Panel, and go to **Programs > Programs and Features**. Right-click **Sisense**, and click **Change**.



2. Click **Continue > Change Features**, and define the settings as described below.



Settings

ElastiCube Server

Select the **Long index edition** if your ElastiCubes are expected to contain a table with over 300 million rows.

It is not recommended to use this configuration setting for ElastiCubes in which the largest table does not include at least 300 million rows.

Sisense Web Application

You must specify the Sisense Web Application **Port** and **Website Name**. Usually, if no other sites are running on the machine, you can use port *8081*.

There are three options for the server type:

- ▶ **Microsoft IIS Express:** This is the default web server used to host the web application.
- ▶ **Microsoft IIS:** This is recommended for production environments. In case there is an existing site with the same name, it will be overridden.
- ▶ **Manual:** This is recommended when you have to manually set up your IIS. If this is selected during upgrades, the installation will keep the existing site details and configuration in the IIS. When using this setting, make sure to define the physical path of the IIS site to the new installation folder (C:\Program Files\Sisense\PrismWeb).

Performing a Silent Installation

You can install Sisense using the command-line interpreter, for example, when you need to run an installation on a remote machine.

You can also customize the installation using variables, for example, install Sisense without the sample dashboards.

To perform a silent installation:

1. Download the [latest Sisense version](#).
2. Open the command line interpreter and type in the installation commands as follows:

Full installation (without previous user activation)

```
<Location of downloaded executable  
file>\SisenseLatestFull.exe -q -username=<> -  
password=<>
```

Full installation (for user already activated by Sisense)

```
<Location of downloaded executable  
file>\SisenseLatestFull.exe -q
```

Full offline installation (requires a license key from Sisense)

Note: To get an offline license, go to [My Account](#) and click **GET LICENSE KEY**, or contact your Sisense Success Manager.

After attaining your key, copy and paste it after -

offlinelicense= (replacing <offline_license_key> in the example below).

```
<Location of downloaded executable  
file>\SisenseLatestFull.exe -q -  
offlinelicense=<offline_license_key>
```

Installation without sample ElastiCubes and dashboards:

```
<Location of downloaded executable  
file>\SisenseLatestFull.exe -nosamples -q
```

Choosing Sisense Website name and port:

```
<Location of downloaded executable  
file>\SisenseLatestFull.exe -q -webname=<website_name>  
-webport=<website_port>
```

Silent Customized Installations

To define which Sisense components are installed, you can edit the PrismFeature.xml and copy it to "%programData%\sisense".

If you are using IIS express, download [PrismFeature.xml](#).

If you are using IIS, download [PrismFeature.xml](#).

Note: Create "%programData%\sisense" if it does not exist.

The PrismFeature.xml file contains an XML node named "Feature". Changing its 'Selected' child node value (true/false) determines if the feature is installed or not. For example, in this configuration, the Prism Server Console is installed:

```
<Features>  
<Feature>  
<Id>PrismServerConsole_Feature</Id>  
<Name>Prism Server Console</Name>  
<Selected>true</Selected>  
<Manual>false</Manual>  
<Selector>CheckBox</Selector>  
<Tests />  
</Feature>  
</Features>
```

After editing the XML file, run the installation according to the instructions above.

To uninstall Sisense using the command-line interpreter:

- ▶ In the command-line interpreter, enter:

```
<Location of downloaded executable  
file>\SisenseLatestFull.exe /q /uninstall
```

Backing Up and Migrating a Sisense Installation

This section describes two separate, but related procedures, backing up a Sisense installation and migrating a Sisense installation from one machine to another.

Backing up your installation is recommended before a migration process, but also on a regular basis for securing your data and Sisense configurations.

You may need to migrate your entire setup when moving from a staging/development environment to a production environment, when upgrading hardware, etc.

Note: If you have advanced and/or customized web configurations, such as SSL, SSO, redirects, etc., please consult us through our [Help Center](#).

Backing Up a Sisense Installation

The backup process requires backing up data from the two main Sisense components, the ElastiCube server and the Sisense Web Application.

In addition, if you have any custom plugins located in the directory C:\Program Files\Sisense\PrismWeb\plugins, this folder should be backed up as well.

The following procedure describes how to back up the data from the two components.

To back up Sisense data:

1. To back up ElastiCube data, enter the following path:

... \ProgramData\Sisense\PrismServer\ElastiCubeData

Note: This is the default path. To check which path is being used, open the Sisense Server Console and click on the server preferences icon  to verify the correct path.

Each folder in this location represents an ElastiCube, and contains both the ElastiCube schema, and the data that was imported when the ElastiCube was built. Copy the data to your backup location.

2. To back up Sisense Web data, go to:

... ProgramData%\Sisense\PrismWeb\DB\Prod

Copy the data to your backup location.

3. (Optional) Back up your custom plugins by copying the folder C:\Program Files\Sisense\PrismWeb\plugins and saving it in the appropriate folder in the new server.

Migrating a Sisense Installation

The migration process involves migrating data for the two main Sisense components, the ElastiCube server and the Sisense Web Application. If you are using plugins, you must back up the Plugins folder as well located in C:\Program Files\Sisense\PrismWeb\plugins. The following procedures explain the migration procedures for each component.

ElastiCube Server

ElastiCube data is stored in folders per ElastiCube. Each folder contains both the ElastiCube schema, and the data that was imported when the ElastiCube was last built. You can either copy the folder

where the data is stored, or export the data.

Note: If you had build failures/server restarts, you may encounter multiple copies of the same ElastiCube. You can identify the latest ElastiCube by the path shown in the Server Console.

1. Before you begin, Sisense recommends declaring downtime to avoid the loss of work during the process.
2. Open the Sisense Server Console. Do one of the following:

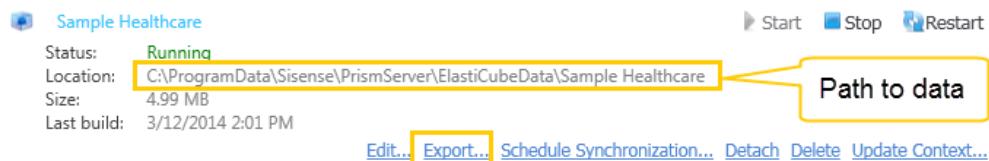
Copy the data folder (see above in the migration section)

or

Export the the data to an .ecdata file. See Importing and Exporting ElastiCubes for instructions on how to export your data.

Note: You must stop the server before exporting the data.

Click **Stop** to stop the server.



3. In the new server, attach the copied data, or import the .ecdata file you exported in the previous step.
4. Start the server.
5. If the database location was changed as well, modify the connectivity settings. See Changing Connectivity Settings to learn how.

Note: If you change the location, first stop the the IIS as follows:

- o If you are using IIS: Open the command line interpreter and type `iisreset /stop`.

- If you are using IIS Express:

Open **Services > sisensewebExpress**, right-click and select **Stop**.

Sisense Web Application

The Sisense web data contains user, dashboard, widget, and other information. This information is stored in a MongoDB instance on the machine where your Sisense Web Application is installed. It's stored in `%ProgramData%\Sisense\PrismWeb\DB\Prod` and can be backed up and moved to your new server.

Note: You can access the database on the Sisense web server using any MongoDB Browser by connecting to localhost on port 27018.

To transfer the Sisense Web Application data, you can use one of the following methods depending on your requirements:

- ▶ If you don't have many dashboards already and do not need too much setting up (adding new users, configuring sharing, data security and other system settings), then you can export the dashboards from the old environment and import them into the new one, set new sharing permissions and configure the environment per your required settings.

Note: If you have any plugins that are based on specific dashboards or widgets, you must modify them manually with their new IDs, as these IDs change when imported into a new environment.

- ▶ If you have created quite a few dashboards and have already configured sharing, data security and other system settings, you can transfer your existing environment as is. In this case, you must:

1. Back up your current MongoDB database and copy it to the new environment, as follows:
 - ▶ To back up the old environment, open your machine's Control Panel, and go to **System and Security > Administrative Tools > Services**, and stop the *SisenseRepository* service (this will make the website inaccessible). You can run the following command:*Stop Sisense.Repository*
 - ▶ Create a copy of the entire **Prod** folder.
 - ▶ Start the *SisenseRepository* service again to get the website up again.*Start Sisense.Repository*
 - ▶ To restore in the new machine, stop the *SisenseRepository* service.
 - ▶ Replace the existing Prod folder with the one you backed up.
 - ▶ Start the *SisenseRepository* service.
2. Migrate the eedata (schema and data):
 - ▶ Stop the Web service to ensure the ElastiCube is not triggered during the copy process:
 - ▶ If you are using IIS: Open the command line and type in *iisreset /stop*
 - ▶ If you are using IIS Express:
Open **Services > sisensewebExpress**, right-click and click **Stop**.

Upgrading Sisense

Sisense releases four major versions a year, and several minor versions in between. Each version includes new features and enhancements as well as other improvements.

Sisense allows you to decide when you want to upgrade to the latest version. There are no automatic updates for the Sisense Server.

Before upgrading, you can learn about the contents of each release in the [Sisense Release Notes](#).

Before Your Upgrade

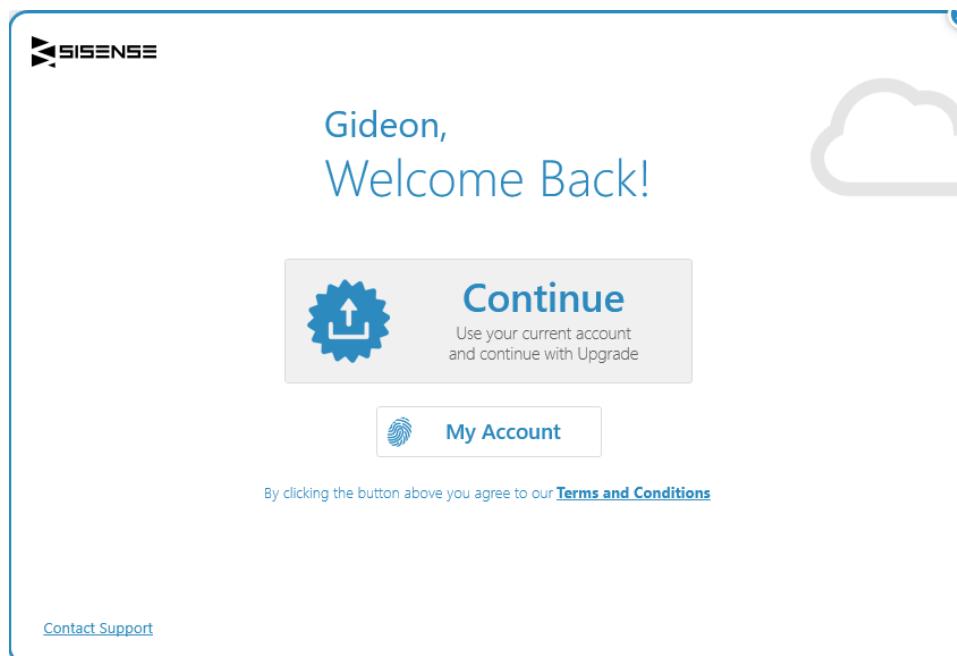
Sisense recommends you follow these guidelines before you begin the upgrade process:

- ▶ Install the new version of Sisense on a testing/staging environment and check all of your ElastiCubes and active dashboards. When you are sure that everything is working well, continue with the installation of the new version in your production environment.
- ▶ If you have a multi-server environment, make sure all of your Sisense servers are running the same version to avoid compatibility issues.
- ▶ Save all open ElastiCubes and close the ElastiCube Manager.
- ▶ Verify that no one is currently designing a dashboard.
- ▶ Declare a downtime of approximately 15 minutes and notify users.
- ▶ Install the new version with the same user who installed the original Sisense version (with the same administrator privileges).

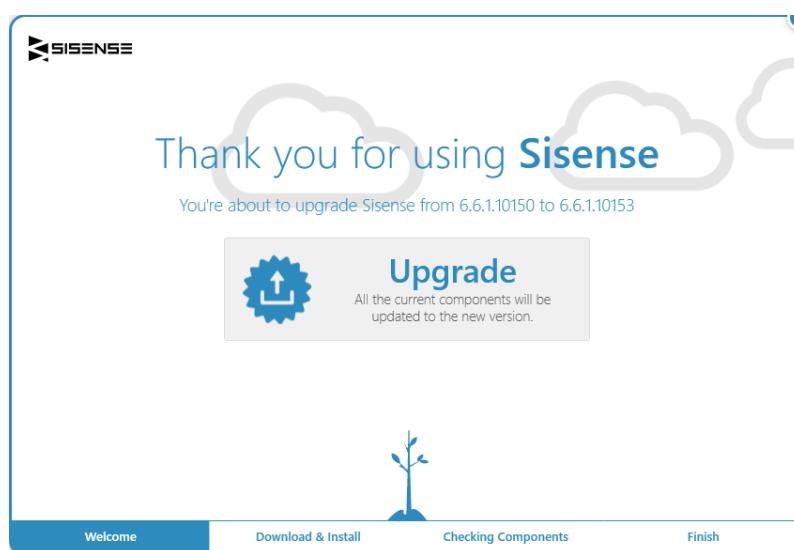
Installing the Latest Version

To install the latest version:

1. Download the latest version of Sisense. For more information, see [Downloading and Installing Sisense](#).
2. Run the installation file. Sisense automatically detects if a version is already installed, and displays the upgrade wizard.



3. Click **Continue**.
4. Click **Upgrade**.



Sisense is upgraded according to the installation settings you selected the last time you installed Sisense. To customize a

new installation, you should uninstall Sisense and then install the latest version.

Downgrading Sisense

When you upgrade Sisense, a backup copy of your MongoDB instance is automatically created. The MongoDB contains your user and dashboard information. This allows you to reinstall earlier versions of Sisense without having to set up your users or dashboards a second time.

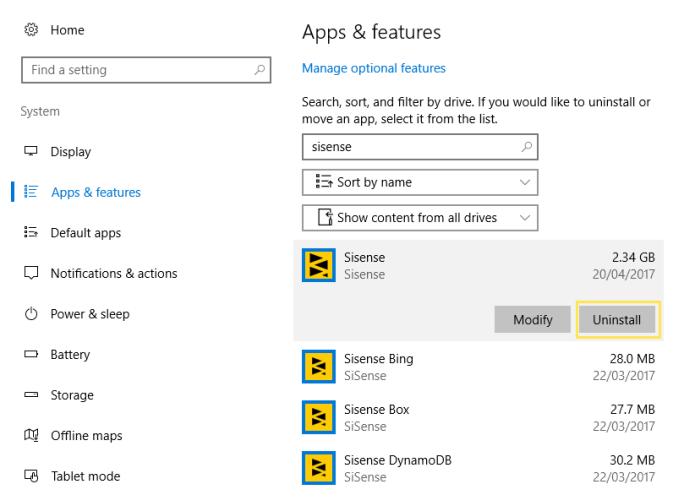
During the uninstall process, you can choose to remove all user data, which deletes the MongoDB instance. The next time you install Sisense, your dashboard, dashboard list, and user data will be completely fresh.

Sisense is improving all the time and some versions of Sisense are incompatible with earlier versions. If you upgrade to a version and need to downgrade to an incompatible version, you can restore a backup copy of the MongoDB created automatically when you installed the last version of Sisense. For help downgrading to an earlier version of Sisense that is incompatible with the last version you installed, you can contact Sisense Support who can assist you in restoring a backup copy of your MongoDB instance.

To downgrade to an earlier version of Sisense:

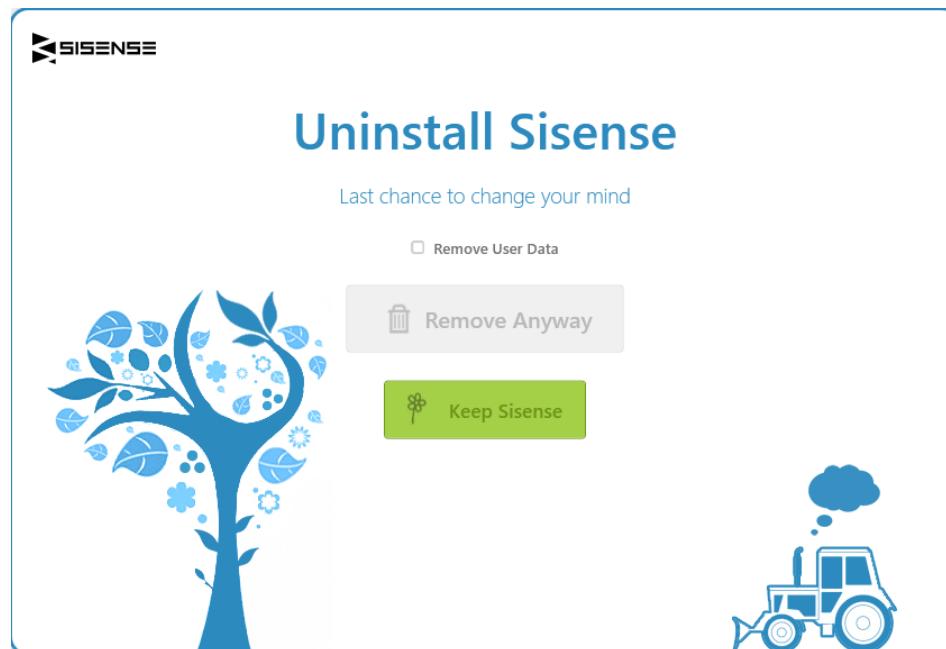
1. In Windows, goto Add or Remove Programs.

2. Locate Sisense and click **Uninstall**.



The Uninstall Wizard is displayed.

3. In the Uninstall Wizard, click **Remove Anyway**. If you select the Remove User Data checkbox, Sisense deletes all information stored in the MongoDB and this information must be configured the next time you install Sisense.



Sisense is uninstalled.

4. Install the previous version of Sisense. The downgrade process is complete.

Launching Sisense

To launch Sisense:

Use one of the following options to launch Sisense:

- ▶ If you have Sisense installed on your machine, select **Sisense** from the **Start** menu.
- ▶ From a standard browser, go to the URL of the Sisense environment provided to you by your system administrator.

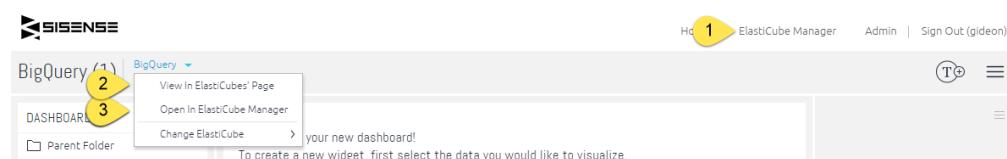
To launch the Sisense ElastiCube Manager:

Use one of the following options:

From Windows: Open the Windows start menu and select **Sisense ElastiCube Manager**.

From the Sisense Web Application: Do one of the following:

1. Click **ElastiCube Manager** in the top menu.
2. Select **View in ElastiCube's Page**, which redirects you to the ElastiCube menu within the Admin page where you can open the ElastiCube Manager through any ElastiCube.
3. Select **Open in ElastiCube Manager**, which opens the ElastiCube Manager and displays the ElastiCube of your current dashboard.



To log into Sisense:

- ▶ If you installed Sisense, you can use the same credentials to log into Sisense.

- ▶ If an account has been created for you, you should receive a password activation email. Click on the link to redirect you to choose a password and activate your account.

Managing Data

Introduction

ElastiCubes are super-fast data stores that are specifically designed to withstand extensive querying typically required by business intelligence applications.

ElastiCubes allow you to bring in data from multiple sources, and then merge, manipulate and query the data as if it was one consolidated data set. ElastiCubes perform so well, that in most cases the creation of dedicated OLAP cubes and/or optimized data marts are completely unnecessary –even when dealing with hundreds of millions of rows of raw data.

One of the biggest advantages of ElastiCubes is the ability to easily mash up multiple data sources. It is made up of fields where each value in one field has a corresponding value in another field. The data for an ElastiCube can come from one source, multiple sources or even from multiple physical locations. Once the data is inside the ElastiCube, it is all the same and every field coming from every table can be analyzed in the context of any other –quickly.

ElastiCube Advantages

ElastiCube technology make queries over hundreds of millions of rows of raw data return in seconds, with moderate hardware requirements including standard desktop-class computers with commodity hardware. More importantly, ElastiCubes can do this without having to pre-aggregate and pre-calculate the data ahead of

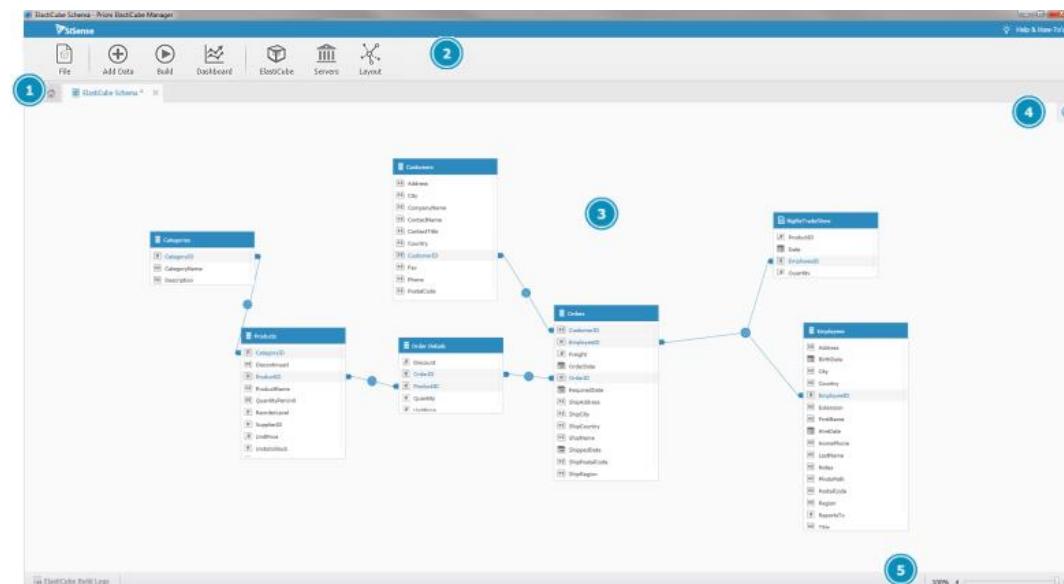
time and store it on the hard-drive, thus radically reducing required import/processing time and storage space.

ElastiCubes are most useful when one or more of the following is true:

- ▶ Large amounts of data need to be analyzed
 - ▶ Data for analysis originates from multiple disparate sources

ElastiCube Manager Environment

The ElastiCube Manager is a visual environment where you can import and structure multiple data sources, create relationships between data, and perform ETL (Extract, transform, and load) processes to prepare data for analysis and visualization.



1. Data Schema Tab
 2. Main Menu
 3. Table and relationship layout
 4. Search tables and fields
 5. Zoom control

To learn more about the ElastiCube Manager, see Working in the ElastiCube Manager.

ElastiCubes - Technical Overview

Introduction

Relational databases (RDBMS) like SQL Server, Oracle, MySQL and even Access all store tabular data row-by-row. This structure is best for transactional/operational systems that require large numbers of concurrent insertions. With indexes, it can also provide realistic query response times for row-based queries that do not frequently require aggregations or joining of many tables.

Data analysis often requires aggregation of data as well as merging of data located in multiple disparate tables. When dealing with these types of queries, relational databases reach their limits pretty quickly. The only way to extend these limits is by putting in stronger hardware and pre-aggregating data to reduce the amounts of calculations that occur in real time.

ElastiCube Columnar Database

ElastiCube data is held in a Columnar Database Management System (CDBMS) that stores data field-by-field. Each field is individually stored in a memory-mapped file, the same mechanism the Windows operating system pagefile uses for memory dumping and loading.

When a query is executed over an ElastiCube, only fields referenced in the query need to be loaded into memory. This leaves enough space for actually processing the query entirely in memory without any read/write to the hard-drive - the prime reason for poor performance

of queries. Once a field is no longer used, it is removed from memory and its consumed space is freed.

This approach has several advantages:

► **Query Response Time**

Queries over data sets containing millions of rows of data return in seconds even under modest hardware configurations such as desktop computers.

► **Materialization Time**

ElastiCubes do not require pre-aggregations and/or creation of indexes to assure fast query response, therefore the actual creation of an ElastiCube takes a fraction of the time of a data mart or an OLAP cube.

► **Storage Space**

Pre-aggregations and the creation of indexes are not needed to assure fast query response, making an ElastiCube's size significantly smaller than a datamart or an OLAP cube.

► **High Compression**

This columnar storage strategy makes the data much more suitable for high levels of compression, without loss of detail or accuracy. This means less hardware is needed; less disk space and less RAM than for an equivalent-sized, traditional Business Intelligence DB.

► **64-bit Support**

Written and designed to natively support 64-bit processing, it vastly increases the amount of memory the system can address at any given time. 64-bit architecture means you can work with virtually unlimited amounts of data.

► **True Multi-User, Multi-Application Architecture**

ElastiCubes are not tightly coupled with the application layer of the system. This frees up a single ElastiCube to handle multiple applications and users. Not having to reproduce your data model for every application saves significant time developing and maintaining your dashboards and reports.

Just-In-Time, In-Memory Processing

► **Smart Cache and Instruction Recycling**

CPU cycles and RAM space are the two most precious resources in any computer, and ElastiCube is designed to use both as efficiently and speedily as possible. Using our sophisticated caching algorithm, the data is only loaded into memory when it's needed. As part of this algorithm, compute-and time-intense calculations are also intelligently cached to further reduce I/O calls.

► **Cache-aware Algorithm**

Additional sophisticated algorithms further increase Sisense's performance. Once data is loaded into memory, the main performance bottleneck becomes CPU cache misses that naturally come with random access. The ElastiCube is specifically designed to minimize these errors by employing a unique cache-aware algorithm, further increasing Sisense's performance by an additional order of magnitude.

► **Compressed Calculations**

Every DB compresses data to save disk space and RAM. ElastiCube is designed to work directly on this compressed

data, so that the need for decompression is virtually eliminated, further increasing ElastiCube's performance.

Designed with Standard Hardware in Mind

Just about every new computer on the market—even portables like iPhones and iPads—are built with very powerful multi-core processors, putting several CPUs into one. ElastiCube was built specifically to take advantage of these powerful CPUs, further increasing Sisense's performance on standard hardware, enabling you to run multiple applications and support multiple users.

► Highly RAM-efficient

The thing we know for sure about DBs is that they grow. Fast. So, no matter how much fancy footwork is done with completely in-memory DBs, eventually you run out of RAM space and need to upgrade—at least your RAM (best case) or your entire hardware platform (worst, very expensive, case). At Sisense we know this, so we spent years designing the ElastiCube to be able to handle terabytes—billions of rows—of data efficiently and quickly, even on standard PC hardware.

Unified Analytics Engine

Sisense can execute queries against a wide variety of data sources as if they were all of the same type, essentially making the individual characteristics of each physical data source unimportant. Our Unified Analytics Engine is what makes this possible.

When Sisense imports data, the Unified Analytics Engine creates a metadata layer, or abstraction layer, which is then used to formulate queries across any number of tables from any number of data sources

in any number of formats. It even supports the combined querying of resident and external (live) database sources without first loading data into the database!

These capabilities provide the user with unparalleled flexibility and speed in creating, executing and sharing highly complex reports, dashboards, and analytic applications, with any number and variety of data sources.

Compliant with Industry Standards

► **Supports SQL-92 Standard**

Even with all this advanced technology, we knew that none of it would be any good if our users couldn't access their existing data. So, we built in an SQL layer to the system, which allows users to integrate Sisense to external applications without needing to learn new scripting languages.

► **Seamless Integration with Existing Data Sources**

Got an ODBC/OleDB compliant DB today? Great, we built in the ability to access those, too. ElastiCube will seamlessly connect to those data sources so, again, there is no need to learn a new language or write special code to connect to your existing data. With ElastiCube there's no need to start over, you just get faster, easier, and more scalable, with minimal need for IT.

Working with the ElastiCube Manager

Menu options in the ElastiCube Manager are located on the top panel. Below are descriptions of the main menu items.

File

Your data model or schema is saved in an ecube file. ecube files contain the schema and the data connection settings required for building your ElastiCube. The ecube file does not contain the raw data. After building your ElastiCube from an ecube file, your ElastiCube with the data is stored in an ecdta file.

- ▶ **Files:** Open and save ElastiCube schema diagrams (ecube files)
- ▶ **Recent Files:** Access a recently used schema diagram.

Add Data

The **Add Data** menu controls access to multiple data sources.

- ▶ **Database servers:** Includes data connectors for traditional and generic databases such as MS SQL and Oracle.
- ▶ **Files:** Includes data connectors for data files like Excel and Access.
- ▶ **Web Services:** Includes data connectors for web-based services like Google Analytics and SalesForce.
- ▶ **Custom:** Custom SQL tables can be generated based on tables and data within the schema.

Build

- ▶ Provides access to build settings to import data into ElastiCube.

Dashboard

- ▶ Opens the Sisense web app to begin designing the dashboard.

ElastiCube

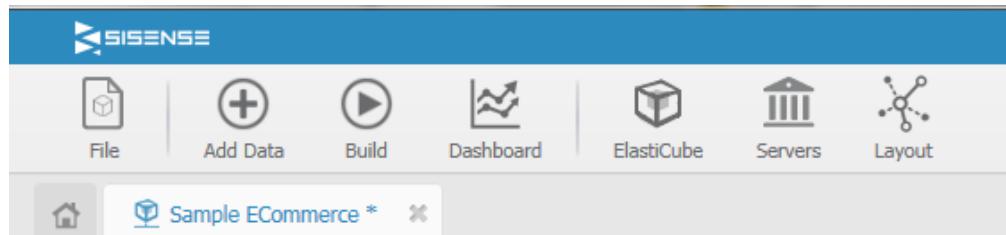
- ▶ *Build ElastiCube:* Provides access to build settings.
- ▶ *Build Settings:* Provides options to change connectivity to data sources, schedule builds and run custom plugins.

Servers

- ▶ Lists available Sisense server consoles and associated ElastiCubes that can be edited or deleted.

Layout

- ▶ Automatically rearranges the layout of tables in the schema diagram.



Sisense Live Connect

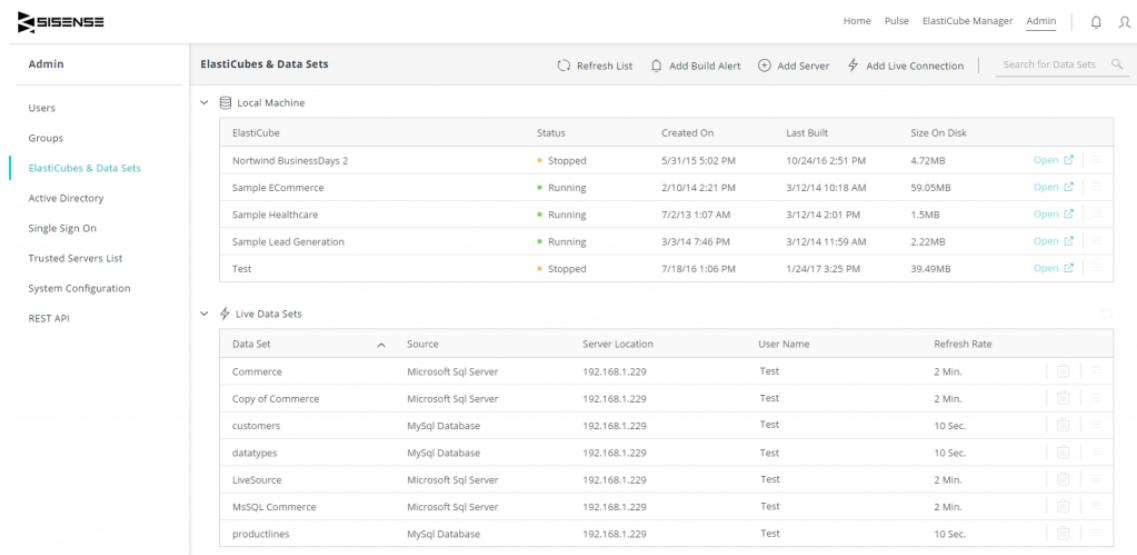
Sisense Live Connect queries are run directly against the data source.

Sisense performs no additional caching on the data.

Having the right data available at the right time is absolutely critical to making smart business decisions. To ensure that you have the latest data available, Sisense provides live connections to some of the most popular data sources.

Traditionally, your data was only as fresh as your latest ElastiCube build. With live connections, you can refresh your dashboards manually or set them to refresh automatically every few seconds.

When you add a live connection, Sisense creates a data set. A data set includes connection and credential details to the data source and defines the table where live data is pulled from. Each data set supports a single table from your data source. After creating a data set, it is displayed in the ElastiCube page of the Admin section from where you can manage all your data sets.



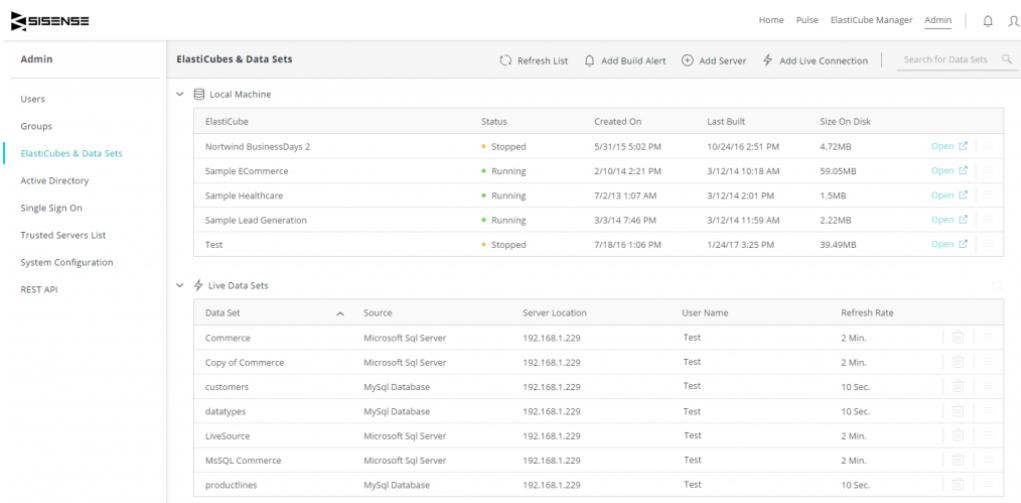
The screenshot shows the Sisense Admin interface with the 'ElastiCubes & Data Sets' page selected. The left sidebar has categories like Users, Groups, Active Directory, Single Sign On, Trusted Servers List, System Configuration, and REST API. The main area has two sections: 'Local Machine' and 'Live Data Sets'. Under 'Local Machine', there is a table with columns: Data Set, Status, Created On, Last Built, and Size On Disk. Under 'Live Data Sets', there is a table with columns: Data Set, Source, Server Location, User Name, and Refresh Rate.

Data Set	Source	Server Location	User Name	Refresh Rate
Commerce	Microsoft Sql Server	192.168.1.229	Test	2 Min.
Copy of Commerce	Microsoft Sql Server	192.168.1.229	Test	2 Min.
customers	MySQL Database	192.168.1.229	Test	10 Sec.
datatype	MySQL Database	192.168.1.229	Test	10 Sec.
LiveSource	Microsoft Sql Server	192.168.1.229	Test	2 Min.
MsSQL Commerce	Microsoft Sql Server	192.168.1.229	Test	2 Min.
productlines	MySQL Database	192.168.1.229	Test	10 Sec.

Adding Live Connections

To add a live connection:

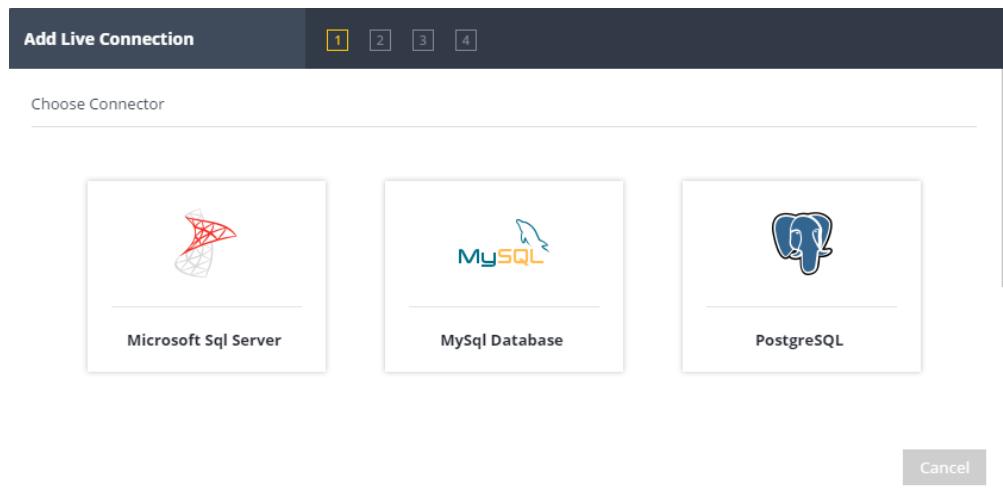
1. In the Sisense Web Application, select **Admin > ElastiCubes & Data Sets**. The ElastiCubes & Data Sets page is displayed.



Local Machine				
ElastiCube	Status	Created On	Last Built	Size On Disk
Nortwind BusinessDays 2	Stopped	5/31/15 5:02 PM	10/24/16 2:51 PM	4.72MB
Sample ECommerce	Running	2/10/14 2:21 PM	3/12/14 10:18 AM	59.05MB
Sample Healthcare	Running	7/21/13 1:07 AM	3/12/14 2:01 PM	1.5MB
Sample Lead Generation	Running	3/3/14 7:46 PM	3/12/14 11:59 AM	2.22MB
Test	Stopped	7/18/16 1:06 PM	1/24/17 3:25 PM	39.49MB

Live Data Sets				
Data Set	Source	Server Location	User Name	Refresh Rate
Commerce	Microsoft Sql Server	192.168.1.229	Test	2 Min.
Copy of Commerce	Microsoft Sql Server	192.168.1.229	Test	2 Min.
customers	MySQL Database	192.168.1.229	Test	10 Sec.
datatype	MySQL Database	192.168.1.229	Test	10 Sec.
LiveSource	Microsoft Sql Server	192.168.1.229	Test	2 Min.
MsSQL Commerce	Microsoft Sql Server	192.168.1.229	Test	2 Min.
productines	MySQL Database	192.168.1.229	Test	10 Sec.

2. In the ElastiCubes & Data Sets page, click **Add Live Connection**. The Add Live Connection dialog box is displayed.



3. In the Add Live Connection dialog box, select your live data source.
4. Under the Connect tab, enter the following credentials:
Location: The IP address of your data source.

User Name: The user name to access data source.

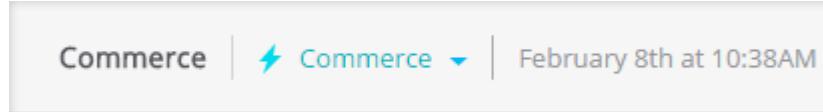
Password: The password to access your data source.

5. Click **Next**.
6. Under the Select Data tab, on the left side is a list of schemas located in your data source. Select the relevant schema. You can find the schema easily by searching for it through the search field at the top of the list to filter the schemas displayed. After selecting the schema, a list of the available tables and views are displayed.
7. Select a table or view. You can find the table easily by searching for it through the search field at the top of the list to filter the tables displayed.
8. Click **Next**.
9. Under the Connection Settings tab, set the following:
 - Connection Name:** Enter a name for your connection. This field defines the name of your data source displayed in the Sisense Web Application.
 - Refresh Rate:** Enter or select the refresh rate for your data source. The shortest refresh rate you can set is 10 seconds.
 - Time Out:** Enter or select the Time Out rate for your data source. The Time Out rate determines how long Sisense waits before cancelling a query that does not receive a response from your data source.
10. Click **Done**. The connection is added as a data set to your ElastiCube & Data Set page.

Working with Data Sets

After you have added your data set, you can begin to create visualizations like you would with any other data source. See [Creating Dashboards](#) and [Creating Filters](#) for more information.

In the Sisense Web Application, live data sources have the  icon next to their name.



Commerce |  Commerce | February 8th at 10:38AM

While you can create dashboards from a live data set like you would an ElastiCube, you can also add data sets as a data source to a dashboard that already has a data source such as an ElastiCube. When have multiple data sources working together on a single dashboard, the data source and the widgets built on it operate independently of each other. This allows you to view widgets that visualize time sensitive information in addition to widgets that are refreshed less frequently. For more information about adding multiple data sources, see [Changing a Dashboard's Data Source](#).

In terms of governance, all existing access rights settings and data security settings can be applied for live connections.

Load Management

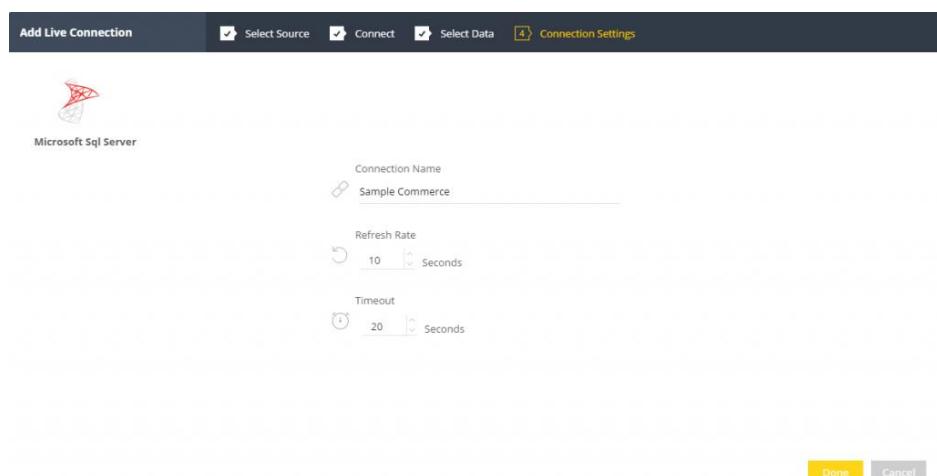
The most significant difference between dashboards built on ElastiCubes and dashboards connected to live data sources is how the load is managed. With an ElastiCube, you refresh the data when you build an ElastiCube and query the ElastiCube, not the data sources that define your ElastiCube. With live connections, queries are run

against the data source itself. This means that you don't have to wait for an ElastiCube to finish building before you see the results, however, your data source must be able to handle the potential load of multiple refreshes.

When you refresh the data for a live connection, Sisense uses the connection settings and credentials in the data set to query the live data source. Depending on your Sisense role, there are several options for refreshing the data.

Administrators create the data set. They define the refresh rate in seconds and timeout for each data set. After the refresh rate is set by the Administrator, no user can set a refresh rate shorter than the rate set by the Administrator. Administrators can manually refresh the data set from the ElastiCube & Data Set page in the Admin section.

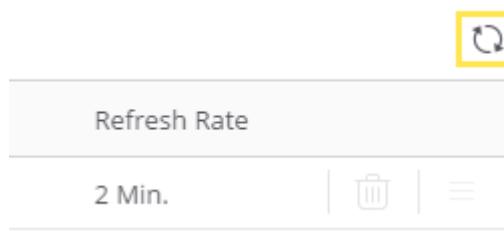
Keep in mind that each time a refresh takes place, your data source is queried, so your data source needs to handle the refresh rate you define. Designers can reduce the frequency of data refreshes or stop automatic refreshes on a per widget level, but they cannot increase the frequency of refreshes beyond the threshold set by an Administrator.



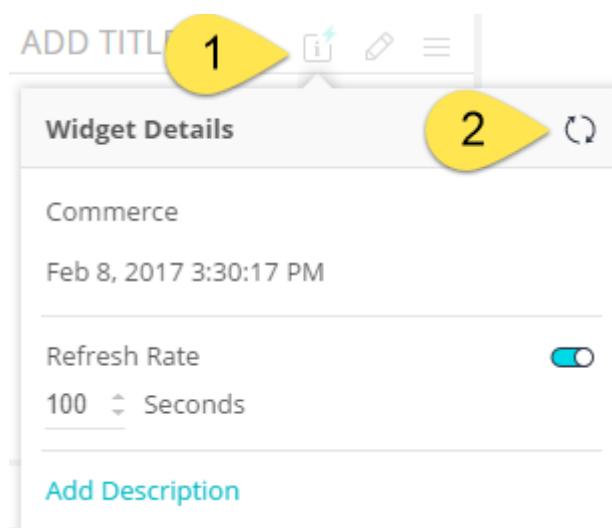
Refreshing a Data Set

There are several ways in which you can refresh a widget build on a live data set depending on your Sisense role.

Administrators set the default refresh and can refresh the data manually from the ElastiCubes & Data Sets page in the Admin section of the Sisense Web Application.

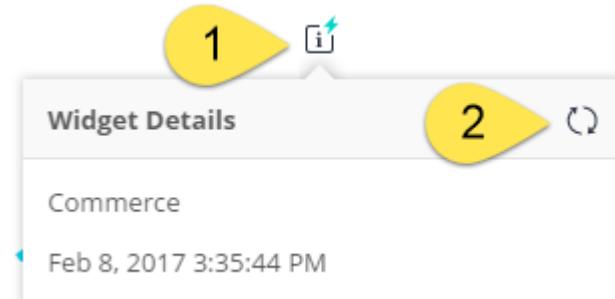


Designers can refresh a widget from the widget's information window .



In addition, Designers can reset the Refresh Rate, however, the rate cannot be lower than the default rate set by the Administrator.

Like Designers, Viewers can refresh a widget from the widget's information window , but they cannot modify the refresh rate.

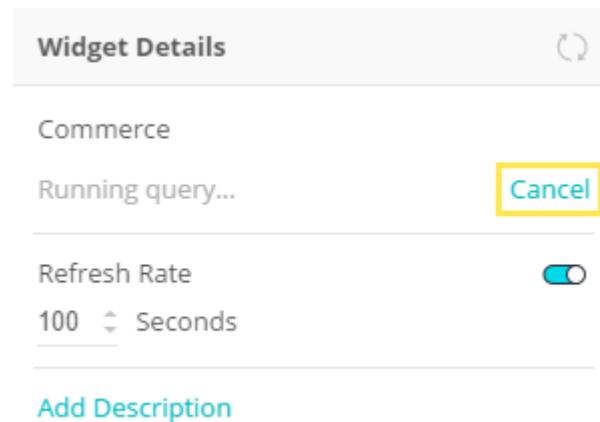


While any users can refresh a widget, Sisense has a minimum 10 second refresh limit. Any refreshes requested within 10 seconds are ignored.

Cancelling a Refresh

If you have a refresh that you want to cancel, you can cancel it by clicking Cancel in the widget's information box.

This button is displayed only when a refresh is in progress.



In addition, any time a user leaves a query, for example, to view another page in their browser, Sisense pauses the query.

Limitations

The current features are not supported by live connections:

1. Pulse alerts

2. Bow & Whisker
3. R is not currently supported
4. Some functions are not currently supported

Frequently Asked Questions

Q: I cannot connect to my data source?

A: As the connection is live, the data source must be available for Sisense to connect. Also, if your credentials changes, you must update the Administrator must update their data set so Sisense can continue to connect and refresh the data.

Q: No data sources are displayed when trying to add one?

A: In Windows Services, verify that the following are running, and if not, restart them:

Sisense.CLRConnectorsContainer

Sisense.Discovery

IIS

Q: If I use Windows Authentication with my SQL server, what do I enter in the Username and Password fields?

A: Enter the Location and leave the Username and Password fields blank.

Q: After entering a data set name, the Done button is not displayed:

A: Make sure the data set's name is unique. You cannot create data sets with the same name.

Connecting to Data Sources

Introduction to Data Sources

Data is the basis for any analytical or explorative business intelligence. In Sisense, data is generally managed via one or more data source. Data Sources hold information on metadata and connectivity settings for the data used in dashboards.

One of the biggest advantages of this approach is the ability to combine and mash up data from multiple sources quickly and easily.

A Data Source contains the following:

- ▶ *Connectivity information* such as the server address, table name, login credentials, etc. A Data Source maintains a connection for a single table, view, OLAP cube or an ElastiCube.
- ▶ *A dimensional representation* of the data, in terms of dimensions and measures.
- ▶ *User-created resources* such as custom filters or measures, widgets or questions.

This section provides a list of each connector and how to connect to its data source.

Connecting to Amazon RDS

Amazon Relational Database Service (Amazon RDS) is a web service that makes it easy to set up, operate, and scale a relational database in the cloud.

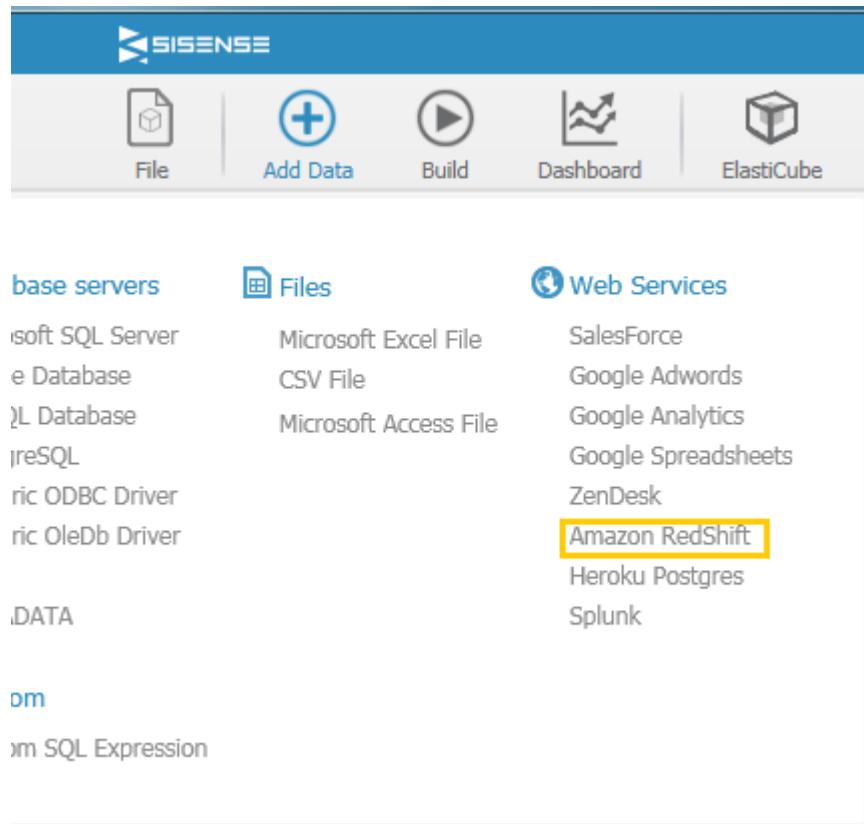
With ElastiCube Manager you can easily connect to your Amazon RDS instances.

Amazon RDS supports instances with MySQL, Oracle and SQL Server databases. You can connect to those instances in the same manner as connecting to those databases on premise, with the IP and credentials as supplied by Amazon. Follow these instructions depending on your RDS type:

Connecting to Amazon RedShift

The ElastiCube Manager enables easy and quick access to databases, tables and views contained within Amazon RedShift databases. The steps below detail how to connect to this type of data source.

1. Click **Add Data** in the top menu of the ElastiCube Manager.
2. Under the **Web Services** category select **Amazon RedShift**.

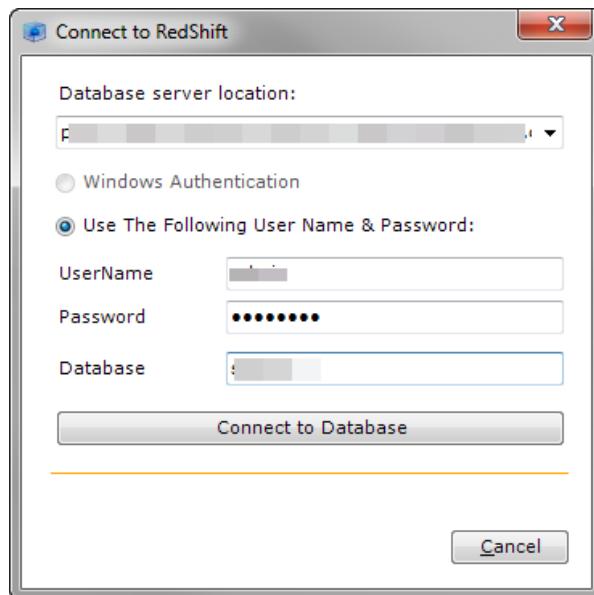


The screenshot shows the Sisense application interface. At the top, there is a navigation bar with icons for File, Add Data, Build, Dashboard, and ElastiCube. Below the navigation bar, there are two main sections: 'base servers' and 'Web Services'. The 'base servers' section lists various ODBC drivers and file types. The 'Web Services' section lists various cloud services. The 'Amazon RedShift' entry in the 'Web Services' section is highlighted with a yellow box.

base servers	Files	Web Services
Microsoft SQL Server	Microsoft Excel File	SalesForce
Oracle Database	CSV File	Google Adwords
MySQL Database	Microsoft Access File	Google Analytics
PostgreSQL		Google Spreadsheets
Microsoft ODBC Driver		ZenDesk
Microsoft OleDb Driver		Amazon RedShift
DATA		Heroku Postgres
		Splunk

Below the 'base servers' and 'Web Services' sections, there is a 'From' section containing a 'From SQL Expression' button.

3. You will be prompted to enter the following information:
 - ▶ **Database server location:** Enter the computer/server IP address that holds the database.
 - ▶ Either use your **Windows Authentication** if configured with the database, or alternatively enter the **Username** and **Password** to connect to the database.
 - ▶ **Database:** Enter the name of the database to which you want to connect.



4. Click **Connect to Database**.

All tables and views associated with the database will appear in a new window.

To view a preview of data contained in a particular table, highlight the table or view in the list and click the preview pane below. To preview the table, select the **Preview** checkbox.

Enable the checkbox next to each table or view you would like to use.

Existing relationships between tables can be automatically replicated in the ElastiCube by selecting the **Automatically create relationships from database** checkbox. Likewise fields with similar names can be linked by selecting the **Automatically create relationships for fields with the same name** checkbox.

Note: If you encounter problems connecting to Amazon RedShift, please see [this article](#) in our support forums.

5. Once all relevant tables are selected click **Add**.

Connecting to Access

The ElastiCube Manager enables easy and quick access to tables contained within Microsoft Access files. The steps below describe how to connect to this type of data source.

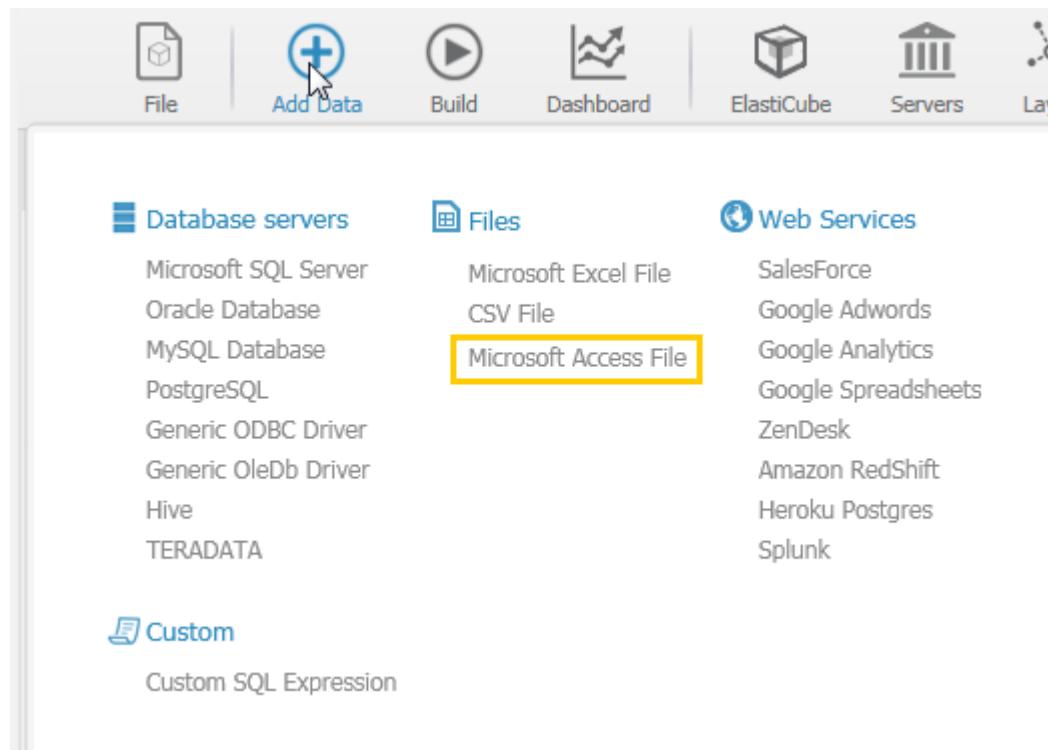
1. Click **Add data** in the top menu of the ElastiCube Manager.
2. Under the **File** category, select **Microsoft Access File**.
3. In the window that opens, locate and select the Microsoft Access file you want to use. Click **OK**.
4. A popup will open with the following options.

All tables and views associated with the database will appear in a new window. To preview the data contained in a particular table, highlight the table or view in the list and click the preview pane below. To preview the table, select the **Preview** checkbox.

5. Select the checkboxes next to each table or view you want to use.

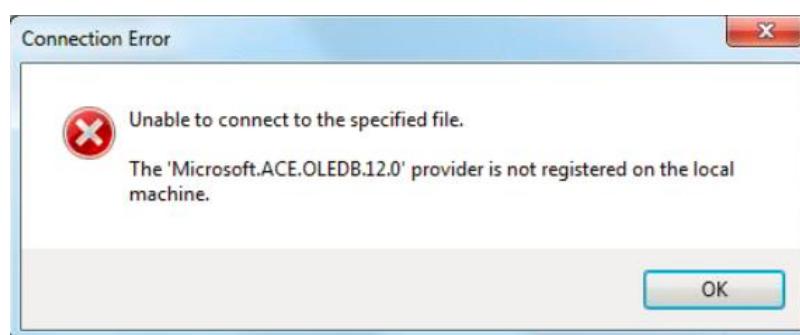
Existing relationships between tables can be automatically replicated in the ElastiCube by selecting the **Automatically create relationships from database** checkbox. Likewise fields with similar names can be linked by selecting the **Automatically create relationships for fields with the same name** checkbox.

6. Once all relevant tables are selected, click **OK**.



The screenshot shows the Sisense application's main interface. At the top, there is a navigation bar with icons for File, Add Data, Build, Dashboard, ElastiCube, Servers, and Lay. The 'Add Data' button is currently selected. Below the navigation bar, there are three main sections: 'Database servers', 'Files', and 'Web Services'. The 'Database servers' section lists various database types. The 'Files' section lists file types, with 'Microsoft Access File' highlighted by a yellow box. The 'Web Services' section lists various cloud and enterprise services. At the bottom, there is a 'Custom' section with a 'Custom SQL Expression' option.

Note: Sisense does not support Access files that were created with the 32-bit version of Access. If you try to upload an Access file created in a 32-bit version of Access, the following connection error is displayed.



To upload 32-bit Access files, you can install Microsoft's 64-bit [Access Database Engine](#) to upload your Access files or click [here](#) for more information.

Connecting to Bing

The Sisense Bing connector is a standalone connector that allows you to import data from Bing's APIs into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to Bing's API. To obtain a connection string, you will need to create a Bing app.

Once you have connected to Bing, you can import a variety of tables from the Bing API.

This section describes how to install the Bing connector, how to connect to Bing with a connection string, and what tables you can import into the ElastiCube Manager

Installing the Bing Connector

Sisense provides the Bing connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Bing connector:

1. [Download](#) the Bing installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.

The Bing connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

 Database servers	 Files	 Web Services
Microsoft SQL Server	Microsoft Excel File	ServiceNow
Oracle Database	CSV File	HubSpot
MySQL Database	Microsoft Access File	Gmail
PostgreSQL		Dynamo DB
Generic ODBC Driver		Twitter
Generic OLE DB Driver		Box
TERADATA		Microsoft Dynamics CRM
		Netsuite
		QuickBooks Online
		Stripe
		Youtube
		Facebook
		Bing
		SalesForce
		Google Analytics
		Google Spreadsheets
		Zendesk
		Amazon RedShift
		Heroku Postgres
		Splunk
	 Custom	
	Custom SQL Expression	

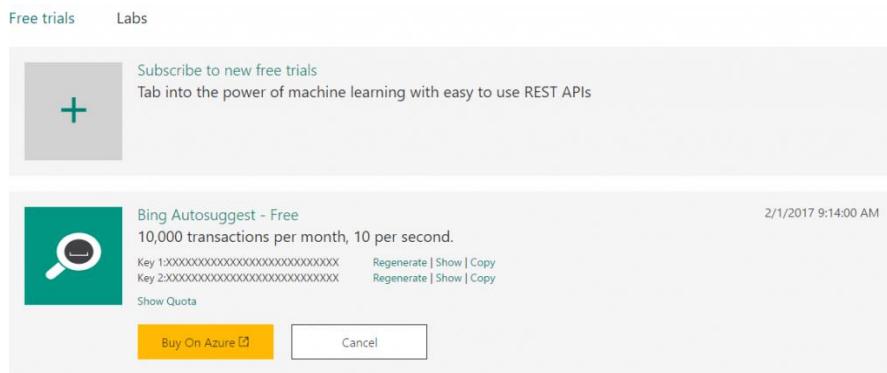
Connecting to the Bing REST API

To access Bing's REST API from Sisense, you must obtain a valid API key. The API Key is provided by Bing after you subscribe to an API.

Obtaining an API Key:

1. Log in to <https://www.bing.com/dev>.
2. From the main menu, select the API you want to retrieve data from.
3. If you have already subscribed to it, you will see two subscription types, Free Trial and Labs. Select the relevant subscription type.

The API Key is displayed.



The screenshot shows the Sisense API Keys page. At the top, there are two tabs: "Free trials" (selected) and "Labs". Below the tabs is a button with a plus sign and the text "Subscribe to new free trials" and "Tab into the power of machine learning with easy to use REST APIs". The main content area displays a single API key entry for "Bing Autosuggest - Free". The entry includes a magnifying glass icon, the key name, a quota of "10,000 transactions per month, 10 per second.", and two keys: "Key 1:XXXXXXXXXXXXXXXXXXXXXX" and "Key 2:XXXXXXXXXXXXXXXXXXXXXX". There are "Regenerate" and "Show | Copy" links for each key. Below the keys is a "Show Quota" link. At the bottom of the card are two buttons: "Buy On Azure" (yellow) and "Cancel". The timestamp "2/1/2017 9:14:00 AM" is visible at the top right of the card.

4. Click **Copy** for the relevant key.
5. Save the key to paste it into your connection string.

Adding Bing Tables to your ElastiCube

Sisense uses connection strings to connect to Bing and import data into the ElastiCube Manager.

The connection string to connect to Bing has the following structure:

```
jdbc:bing:ApiKey=73529b7a4994ytec8d4ca6c33049;InitiateOAuth=G  
ETANDREFRESH;
```

Mandatory properties are ApiKey, and InitiateOAuth.

- ▶ **ApiKey:** Application key provided by Microsoft that authenticates your account.
- ▶ **InitiateOAuth:** Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken connection property.

Note: To switch between accounts, you need to delete the file OAuthsettings.txt located at

`...\Users\XXX\AppData\Roaming\CData\Bing Data Provider`.

To add Bing data:

1. In ElastiCube Manager, click **Add Data** and then, **Bing**. The Connect to Bing window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Bing is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Bing and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**. The tables are displayed in the ElastiCube Manager.

Switching between Accounts

When you connect to the Bing data source, Sisense saves your OAuth values in the file OAuthsettings.txt file located at

.\Users\xxx\AppData\Roaming\CDATA\Bing Data Provider on your Sisense server. To connect to the Bing data source with another user on the same machine, you must delete the OAuthsettings.txt file. Sisense will then generate a new file for that user.

Another option to support multiple users is to define the location and file name of an OAuthsettings file for each unique user in your connection string through the **OAuthSettingsLocation** parameter. When each user connects to the data source, Sisense generates the OAuth file with the file name you specify in the location you define. In the examples below, two users are allowed to access the Bing data source and for each user, Sisense generates a file that contains that user's OAuth values in the location defined in the string.

```
jdbc:Bing:OAuthSettingsLocation=C:\Bing\auth\john.txt;OAuthClientID=11276856774486;OAuthClientSecret=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhost/;
```

```
jdbc:Bing:OAuthSettingsLocation=C:\Bing\auth\sally.txt;OAuthClientID=11276856774486;OAuthClientSecret=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhost/;
```

In the example above, two OAuth files are created, one for John and one for Sally in the location C:\Bing\auth\.

This is useful if you support many users who each need to access the Bing data source.

Bing Tables

The Sisense Bing connector allows you to import the following tables into the ElastiCube Manager.

Name	Description
ImageSearch	<p>Query the Bing image search engine.</p> <p>When querying this table the SearchTerms parameter must be specified. For example, to search Bing Images for the term 'bing' use the following query:</p> <pre>SELECT * FROM ImageSearch where SearchTerms = 'bing'</pre> <p>You can also specify a search safety level:</p> <pre>SELECT * FROM ImageSearch where SearchTerms = 'bing' and SafeSearch = 'Moderate'</pre>
NewsSearch	<p>Query the Bing news search engine</p> <p>When querying this table the SearchTerms parameter must be specified. For example, to search Bing Images for the term 'bing' use the following query:</p>

Name	Description
	<pre>SELECT * FROM NewsSearch where SearchTerms = 'bing'</pre> <p>You can also specify a search safety level:</p> <pre>SELECT * FROM NewsSearch where SearchTerms = 'bing' and SafeSearch = 'Moderate'</pre>
Search	<p>Query the Bing search engine.</p> <p>When querying this table the SearchTerms parameter must be specified. For example, to search Bing Images for the term 'bing' use the following query:</p> <pre>SELECT * FROM Search where SearchTerms = 'bing'</pre> <p>You can also specify a search safety level:</p> <pre>SELECT * FROM Search where SearchTerms = 'bing' and SafeSearch = 'Moderate'</pre>
VideoSearch	<p>Query the Bing video search engine.</p> <p>When querying this table the SearchTerms parameter must be specified. For example, to search Bing Images for the term 'bing' use the following query:</p> <pre>SELECT * FROM VideoSearch where SearchTerms = 'bing'</pre> <p>You can also specify a search safety level:</p> <pre>SELECT * FROM VideoSearch where SearchTerms = 'bing' and SafeSearch = 'Moderate'</pre>

Name	Description
WebSearch	<p>Query the Bing web search engine.</p> <p>When querying this table the SearchTerms parameter must be specified. For example, to search Bing Images for the term 'bing' use the following query:</p> <pre>SELECT * FROM WebSearch where SearchTerms = 'bing'</pre> <p>You can also specify a search safety level:</p> <pre>SELECT * FROM WebSearch where SearchTerms = 'bing' and SafeSearch = 'Moderate'</pre>

Connecting to Box

The Sisense Box connector is a standalone connector that allows you to import data from Box's API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the Box APIs. To obtain a connection string, you will need to register a Box app.

Once you have connected to Box, you can import a variety of tables from the Box API.

This section describes how to install the Box connector, how to connect to Box with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the Box Connector

Sisense provides the Box connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Box connector:

1. [Download](#) the Box installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The Box connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

Connecting to the Box REST API

To access Box's REST API from the ElasticCube Manager, you must provide valid Oauth Box credentials through a connection string.

These credentials are provided by Box when you create an application.

After you receive your credentials from Box, you can create the connection string and provide Sisense with it to connect to your data.

Registering an App

You can follow the procedure below to register an app and obtain the OAuth client credentials, the OAuthClientId and OAuthClientSecret:

1. Log in to your Box developers dashboard and click **Create a Box Application**.
2. If you are making a desktop application, set the Redirect URI to <http://localhost:3333> or a different port number of your choice. If you are making a Web application, set the Redirect URI to a page on your Web app you would like the user to be returned to after they have authorized your application. The OAuthClientId and OAuthClientSecret are also displayed in the same page.
3. Select the scope of user permissions your app will request.

Authenticating through Box

After setting the following properties, you are ready to connect:

- ▶ **OAuthClientId:** Set this to the Client Id in your app settings.
- ▶ **OAuthClientSecret:** Set this to the Client Secret in your app settings.
- ▶ **CallbackURL:** Set this to the Redirect URL in your app settings.

- ▶ **InitiateOAuth:** Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken connection property.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver completes the OAuth process.

1. Extracts the access token from the callback URL and authenticates requests.
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Adding Box Tables to your ElastiCube

Sisense uses connection strings to connect to Box and import data into the ElastiCube Manager. Each connection string contains a authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add Box data:

1. In ElastiCube Manager, click **Add Data** and then, **Box**. The Connect to Box window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Box is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Box and displays a list of tables available for you to import.

5. Select the relevant tables and click **Add**. The tables are displayed in the ElastiCube Manager.

Box Tables

Box's RESTful APIs expose the following Box tables that you can import into the ElastiCube Manager through the Sisense Box connector:

Available Tables

Name	Description
Collaborations	Create, update, delete, and query the available Collaborations in Box.
Comments	Create, update, delete, and query the available Comments in Box.
Files	Update, delete, and query the available Files in Box.
Folders	Create, update, delete, and query the available Folders in Box.
Groups	Create, update, delete, and query the available Groups in Box.
Memberships	Create, update, delete, and query the available Memberships in Box.
TaskAssignments	Create, update, delete, and query the available Task Assignments in Box.
Tasks	Create, update, delete, and query the available Tasks in Box.
TrashedItems	Delete and query the available TrashedItems in Box.
Users	Create, update, delete, and query the available Users in Box.

Available Views

Name	Description
SharedItems	Query the available shared items in Box.
UserEvents	Query the available Events in Box.

Limitations

1. Aggregate functions are not supported.
2. To query the Collaborations table, specify an Id, FolderId, or a GroupId.

```
SELECT * FROM Collaborations WHERE Id = '123'
```

3. To select from the Tasks table you need to specify an Id or an ItemId.

```
SELECT * FROM Tasks WHERE Id = '123'
```

4. To select from the TasksAssignments table you need to specify an Id or a TaskId.

```
SELECT * FROM TasksAssignments WHERE Id = '123'
```

5. To select from the Memberships table you need to specify an Id or a GroupId.

```
SELECT * FROM Memberships WHERE Id = '123'
```

6. To select from SharedItems specify the Shared Link that points to the item.

```
SELECT * FROM SharedItems WHERE SharedLink =
'www.url.com'
```

If the SharedItem has a password also specify the password.

```
SELECT * FROM SharedItems WHERE SharedLink =
'www.url.com' AND SharedLinkPassword = 'password'
```

7. To query the Comments table specify an Id or a FileId.

```
SELECT * FROM Comments WHERE Id = '123'
```

8. Due to these limitations, ElastiCubes cannot be built without custom SQL.

Connecting to CSV

The ElastiCube Manager enables easy and quick access to tables contained within CSV files.

You can add a single csv file or multiple csv files to your ElastiCube. A typical use case for importing multiple files is when you want to split a large csv file into multiple smaller files.

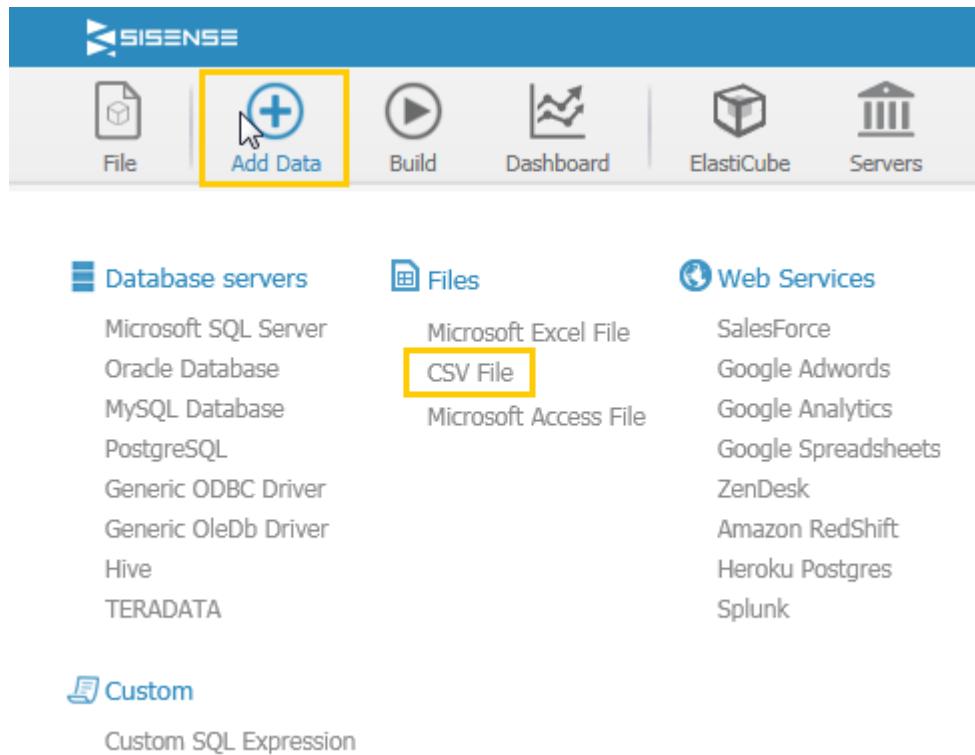
Notes:

- ▶ Make sure that all the files adhere to the same structure, including the presence or absence of headers in the first row. Files will be added to the table based on their file names in alphabetical order.
- ▶ When you import multiple files from a folder and build your ElastiCube, each consecutive build will import data from all the updated and new files in the folder. If you have any questions about data accumulation between builds, please contact our [support team](#).
- ▶ If your csv file contains special symbols or foreign characters, the import process may fail. To resolve this issue, convert the csv file to the UTF-8 format. One way to do this is to open the .csv file in an application like Notepad, and then select File > Save As. At the bottom of the dialog box, open the Encoding settings, select UTF-8 and save the file as a new file.

To connect CSV files:

1. In the Elasticube Manager, click **Add Data**.

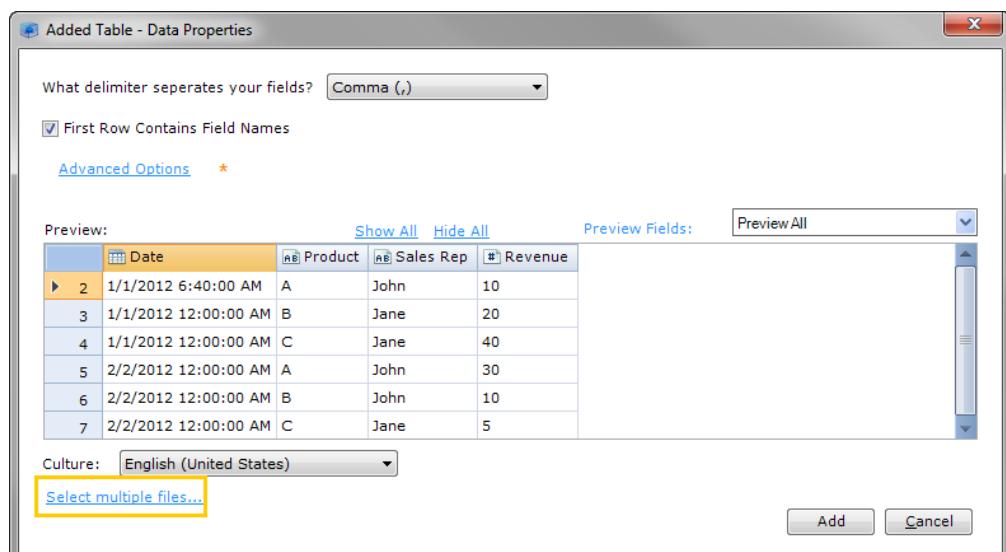
- Under the **Files** category select **CSV File**.



The screenshot shows the Sisense interface with the following components:

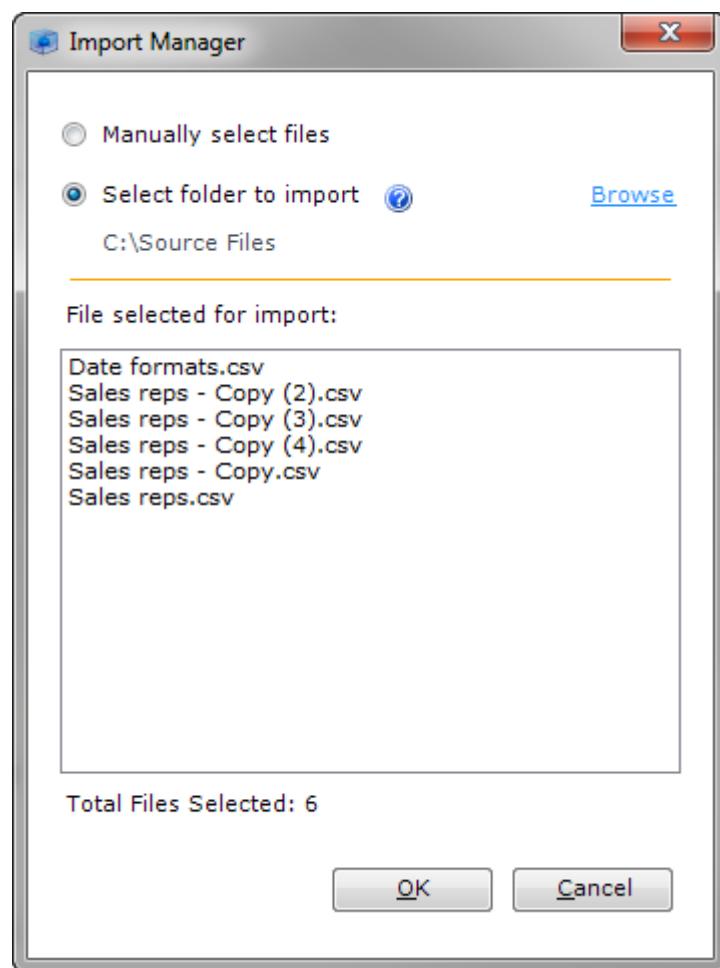
- Top Bar:** Includes icons for File, Add Data (highlighted with a yellow box), Build, Dashboard, ElastiCube, and Servers.
- Database servers:** Microsoft SQL Server, Oracle Database, MySQL Database, PostgreSQL, Generic ODBC Driver, Generic OleDb Driver, Hive, TERADATA.
- Files:** Microsoft Excel File, CSV File (highlighted with a yellow box), Microsoft Access File.
- Web Services:** SalesForce, Google Adwords, Google Analytics, Google Spreadsheets, ZenDesk, Amazon RedShift, Heroku Postgres, Splunk.
- Custom:** Custom SQL Expression.

- Select a file to import. If you want to import multiple files, select just a single file that will be used for previewing the configuration options. However, all files are treated with the same configuration.



4. Define the following settings in the **Data Properties** window:
 - ▶ **What delimiter separates your sheets:** Allows you to choose the character that separates values within the csv file.
 - ▶ **First Row Contains Field Names:** Enables you to specify table column names based on the header in the first row of the spreadsheet.
 - ▶ Click **Advanced Options** to reveal more options to filter rows:
 - **Ignore rows that start with:** Specify rows to ignore that start with a specific symbol, value or letter.
 - **Ignore rows that contain:** Specify rows to ignore that contain a specific symbol, value or letter.
 - **Text qualifier:** Change the value if necessary.
 - **Ignore first rows:** Specify a number of first rows to ignore.
 - **Ignore last rows:** Specify a number of last rows to ignore.
5. To add additional files:
 1. Click **Select multiple files...**
 2. Click **Select folder to import.** Locate and select the folder. Click **OK.**

3. All csv files in the folder will be selected for import. Click **OK**.



6. Click **Add** to complete the import configuration.

Connecting to DynamoDB

The Sisense DynamoDB connector is a standalone connector that allows you to import data from the DynamoDB API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the DynamoDB API. To obtain a connection string, you will need to create a DynamoDB developers account.

Once you have connected to DynamoDB, you can import a variety of tables from the DynamoDB API.

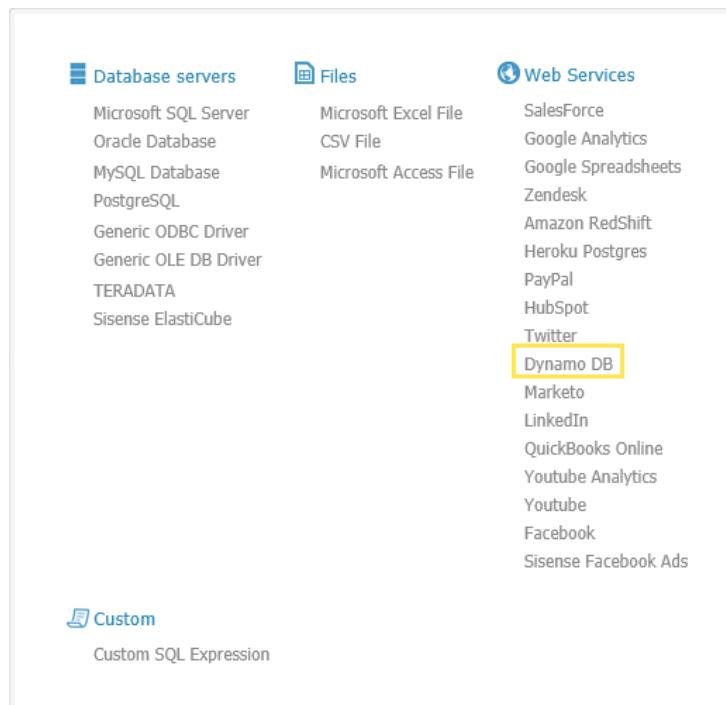
This section describes how to install the DynamoDB connector, how to connect to DynamoDB with a connection string, and what tables you can import into the ElastiCube Manager:

Installing the DynamoDB Connector

Sisense provides the DynamoDB connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the DynamoDB connector:

1. [Download](#) the DynamoDB installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The DynamoDB connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.



Connecting to the DynamoDB REST API

The connection to DynamoDB is made using your **AccessKey**, **SecretKey**, and optionally your **Domain** and **Region**. Your AccessKey and SecretKey can be obtained on the security credentials page for your Amazon Web Services account. Your Region will be displayed in the upper left-hand corner when you are logged into DynamoDB.

- ▶ **AccessKey:** Your AWS account access key. This value is accessible from your AWS security credentials page.
- ▶ **SecretKey:** Your AWS account secret key. This value is accessible from your AWS security credentials page.
- ▶ **Domain:** Your AWS domain name. You can optionally choose to associate your domain name with AWS.
- ▶ **Region:** The hosting region for your Amazon Web Services. Available values are NORTHERNVIRGINIA, OREGON,

NORTHERNCALIFORNIA, IRELAND, SINGAPORE, SYDNEY, TOKYO, and SAOPAULO. The default value is NORTHERNVIRGINIA.

Adding DynamoDB Tables to your ElastiCube

Sisense uses connection strings to connect to DynamoDB and import data into the ElastiCube Manager.

The connection string to connect to DynamoDB has the following structure:

`jdbc:DynamoDB:Property1=Value1;Property2=Value2;`

An example of a connection string:

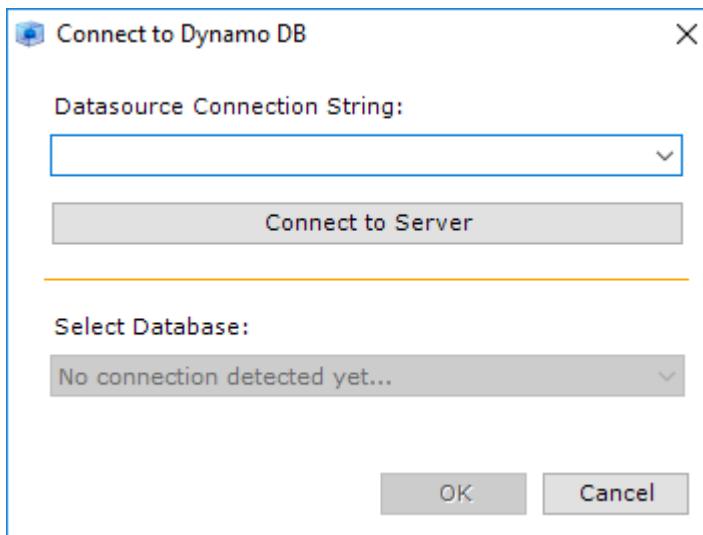
```
jdbc:dynamodb:Access Key=AKIAJH283HDH2932DQ;Secret  
Key=vG07bFgSmvfgsrdfgHsB6iQU/HVOqO9L9g;Domain=amazonaws.com;R  
egion=OREGON;
```

Note: To switch between accounts, you need to delete the file OAuthsettings.txt file located at
`...\\Users\\xxx\\AppData\\Roaming\\CData\\DynamoDB Data Provider.`

To add DynamoDB data:

1. In ElastiCube Manager, click **Add Data** and then, **DynamoDB**.

The Connect to DynamoDB window is displayed.



2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. DynamoDB is displayed in the Select Database list.
4. Click **OK**. Sisense connects to DynamoDB and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.

The tables are displayed in the ElastiCube Manager.

Switching between Accounts

When you connect to the DynamoDB data source, Sisense saves your OAuth values in the file OAuthsettings.txt file located at
 .\Users\xxx\AppData\Roaming\CDATA\DynamoDB Data Provider on your Sisense server. To connect to the DynamoDB data source with another user on the same machine, you must delete the OAuthsettings.txt file. Sisense will then generate a new file for that user.

Another option to support multiple users is to define the location and file name of an OAuthsettings file for each unique user in your connection string through the **OAuthSettingsLocation** parameter.

When each user connects to the data source, Sisense generates the OAuth file with the file name you specify in the location you define. In the examples below, two users are allowed to access the DynamoDB data source and for each user, Sisense generates a file that contains that user's OAuth values in the location defined in the string.

```
jdbc:DynamoDB:OAuthSettingsLocation=C:\DynamoDB\auth\john.txt;OAu
thClientId=11276856774486;OAuthClientSecret
=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Vers
ion=2.8;CallbackURL=http://localhost/;
```

```
jdbc:DynamoDB:OAuthSettingsLocation=C:\DynamoDB\auth\sally.txt;OA
uthClientId=11276856774486;OAuthClientSecret
=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Vers
ion=2.8;CallbackURL=http://localhost/;
```

In the example above, two OAuth files are created, one for John and one for Sally in the location C:\DynamoDB\auth\.

This is useful if you support many users who each need to access the DynamoDB data source.

DynamoDB Tables

The Sisense DynamoDB connector allows you to import the following tables into the ElastiCube Manager.

Name	Description
Table0	Test table.
Table1	Test table.
Movies	Sample data file that contains information about a few thousand movies from the Internet Movie Database (IMDb).



Accumulative Builds

Sisense support accumulative builds for all numeric and dates data types. However, the data must be sorted before building the ElastiCube.

Connecting to Exact

The Sisense Exact connector is a standalone connector that allows you to import data from Exact's API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the Exact APIs. To obtain a connection string, you will need to create a Exact app.

Once you have connected to Exact, you can import a variety of tables from the Exact API.

This section describes how to install the Exact connector, how to connect to Exact with a connection string, and what tables you can import into the ElastiCube Manager:

Installing the Exact Connector

Sisense provides the Exact connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Exact connector:

1. [Download](#) the Exact installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The Exact connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

Connecting to the Exact REST API

To access Exact's REST API from the ElastiCube Manager, you must provide valid Oauth Exact credentials through a connection string.

These credentials are provided by Exact when you register an application.

After you receive your credentials from Exact, you can create the connection string and provide Sisense with it to connect to your data.

Registering an App

Follow the steps below to obtain the OAuth client credentials, the OAuthClientId and OAuthClientSecret:

1. Create an Exact Online developer account.
2. Log into the App Center and click **Manage Apps> Add a New Application.**
3. Enter the app name to be displayed to users when they are prompted to grant permissions to your app.
4. Enter the Redirect URI.If you are making a desktop application, set the Redirect URI to <http://localhost:portnumber>.If you are making a Web application, set the Redirect URI to a page you would like the user to be returned to after they have granted your application permissions.
5. Click **Edit** for your app. The client credentials, the client Id and client secret, are displayed.

Authenticating through Exact

After setting the following connection properties, you are ready to connect:

- ▶ **OAuthClientId:** If you want to use an app you registered with Exact Online, set this value to the client Id you defined in your app settings. To use the embedded credentials of the driver, leave this value blank.

- ▶ **OAuthClientSecret:** If you want to use an app you registered with Exact Online, set this value to the client secret you defined in your app settings. To use the embedded credentials of the driver, leave this value blank.
- ▶ **OAuthCallbackURL:** If you want to use an app you registered with Exact Online, set this value to the Redirect URI in your app settings. To use the embedded credentials of the driver, leave this value blank.
- ▶ **Region:** Set this to the region of the Exact Online service you want to connect to.
- ▶ **InitiateOAuth:** Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the access token in the connection string.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver then completes the OAuth process:

1. Extracts the access token from the callback URL and authenticates requests.
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Adding Exact Tables to your ElastiCube

Sisense uses connection strings to connect to Exact and import data into the ElastiCube Manager. Each connection string contains a authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add Exact data:

1. In ElastiCube Manager, click **Add Data** and then, **Exact**. The Connect to Exact window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Exact is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Exact and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**. The tables are displayed in the ElastiCube Manager.

Exact Tables

Exact's RESTful APIs expose the following Exact tables that you can import into the ElastiCube Manager through the Sisense Exact connector:

Available Tables

- ▶ AcceptQuotation
- ▶ AccountInvolvedAccounts
- ▶ AccountOwners
- ▶ Accounts
- ▶ Addresses
- ▶ BankAccounts
- ▶ BankEntries
- ▶ BankEntryLines
- ▶ CashEntries
- ▶ CashEntryLines
- ▶ CommunicationNotes

- ▶ Complaints
- ▶ Contacts
- ▶ Costcenters
- ▶ CostTransactions
- ▶ Costunits
- ▶ DepreciationMethods
- ▶ DirectDebitMandates
- ▶ DocumentAttachments
- ▶ DocumentFolders
- ▶ DocumentTypeFolders
- ▶ Events
- ▶ ExchangeRates
- ▶ GeneralJournalEntries
- ▶ GeneralJournalEntryLines
- ▶ GLAccounts
- ▶ GoodsDeliveries
- ▶ GoodsDeliveryLines
- ▶ InvoiceSalesOrders
- ▶ InvoiceTerms
- ▶ InvolvedUserRoles
- ▶ InvolvedUsers
- ▶ Items
- ▶ ItemWarehouses
- ▶ Journals
- ▶ Mailboxes
- ▶ MailMessageAttachments
- ▶ MailMessages
- ▶ OperationResources

- ▶ Operations
- ▶ Opportunities
- ▶ PaymentConditions
- ▶ PrintedSalesInvoices
- ▶ PrintedSalesOrders
- ▶ PrintQuotation
- ▶ ProductionAreas
- ▶ ProjectHourBudgets
- ▶ ProjectRestrictionEmployees
- ▶ ProjectRestrictionItems
- ▶ ProjectRestrictionRebillings
- ▶ Projects
- ▶ PurchaseEntries
- ▶ PurchaseEntryLines
- ▶ QuotationLines
- ▶ Quotations
- ▶ RejectQuotation
- ▶ ReopenQuotation
- ▶ ReviewQuotation
- ▶ SalesEntries
- ▶ SalesEntryLines
- ▶ SalesInvoiceLines
- ▶ SalesInvoices
- ▶ SalesItemPrices
- ▶ SalesOrderID
- ▶ SalesOrderLines
- ▶ SalesOrders
- ▶ ServiceRequests

- ▶ ShopOrderMaterialPlans
- ▶ ShopOrderRoutingStepPlans
- ▶ ShopOrders
- ▶ SolutionLinks
- ▶ StockCountLines
- ▶ StockCounts
- ▶ SubscriptionLines
- ▶ SubscriptionRestrictionEmployees
- ▶ SubscriptionRestrictionItems
- ▶ Subscriptions
- ▶ Tasks
- ▶ TimeCorrections
- ▶ TimeTransactions
- ▶ VATCodes
- ▶ Warehouses
- ▶ Workcenters

Available Views

- ▶ AccountantInfo
- ▶ AccountClasses
- ▶ AccountClassificationNames
- ▶ AccountClassifications
- ▶ ActiveEmployments
- ▶ AddressStates
- ▶ AgingOverview
- ▶ AgingPayablesList
- ▶ AgingReceivablesList
- ▶ Assets
- ▶ AvailableFeatures

- ▶ Banks
- ▶ BatchNumbers
- ▶ Budgets
- ▶ Currencies
- ▶ DefaultMailbox
- ▶ Departments
- ▶ Divisions
- ▶ DocumentCategories
- ▶ Documents
- ▶ DocumentsAttachments
- ▶ DocumentTypeCategories
- ▶ DocumentTypes
- ▶ Employees
- ▶ EmploymentContractFlexPhases
- ▶ EmploymentContracts
- ▶ EmploymentEndReasons
- ▶ EmploymentOrganizations
- ▶ Employments
- ▶ EmploymentSalariesFinancialPeriods
- ▶ GLClassifications
- ▶ GLSchemes
- ▶ GLTransactionTypes
- ▶ HourCostTypes
- ▶ ItemGroups
- ▶ ItemVersions
- ▶ JobGroups
- ▶ JobTitles
- ▶ JournalStatusList

- ▶ Layouts
- ▶ OpportunityContacts
- ▶ OutstandingInvoicesOverview
- ▶ PayablesList
- ▶ PreferredMailbox
- ▶ PriceLists
- ▶ ProfitLossOverview
- ▶ ProjectBudgetTypes
- ▶ PurchaseOrderLines
- ▶ PurchaseOrders
- ▶ ReasonCodes
- ▶ ReceivablesList
- ▶ RecentCosts
- ▶ RecentHours
- ▶ ReportingBalance
- ▶ Returns
- ▶ RevenueList
- ▶ SalesPriceListDetails
- ▶ Schedules
- ▶ SerialNumbers
- ▶ ShippingMethods
- ▶ StockBatchNumbers
- ▶ StockSerialNumbers
- ▶ StorageLocations
- ▶ SubscriptionLineTypes
- ▶ SubscriptionReasonCodes
- ▶ SubscriptionTypes
- ▶ TaxComponentRates

- ▶ TaxEmploymentEndFlexCodes
- ▶ TaxScheduleComponents
- ▶ TaxSchedules
- ▶ TimeAndBillingAccountDetails
- ▶ TimeAndBillingActivitiesAndExpenses
- ▶ TimeAndBillingEntryAccounts
- ▶ TimeAndBillingEntryProjects
- ▶ TimeAndBillingEntryRecentAccounts
- ▶ TimeAndBillingEntryRecentActivitiesAndExpenses
- ▶ TimeAndBillingEntryRecentHourCostTypes
- ▶ TimeAndBillingEntryRecentProjects
- ▶ TimeAndBillingItemDetails
- ▶ TimeAndBillingProjectDetails
- ▶ TransactionLines
- ▶ Transactions
- ▶ Units
- ▶ UserRoles
- ▶ UserRolesPerDivision
- ▶ Users
- ▶ VatPercentages

Connecting to Microsoft Excel

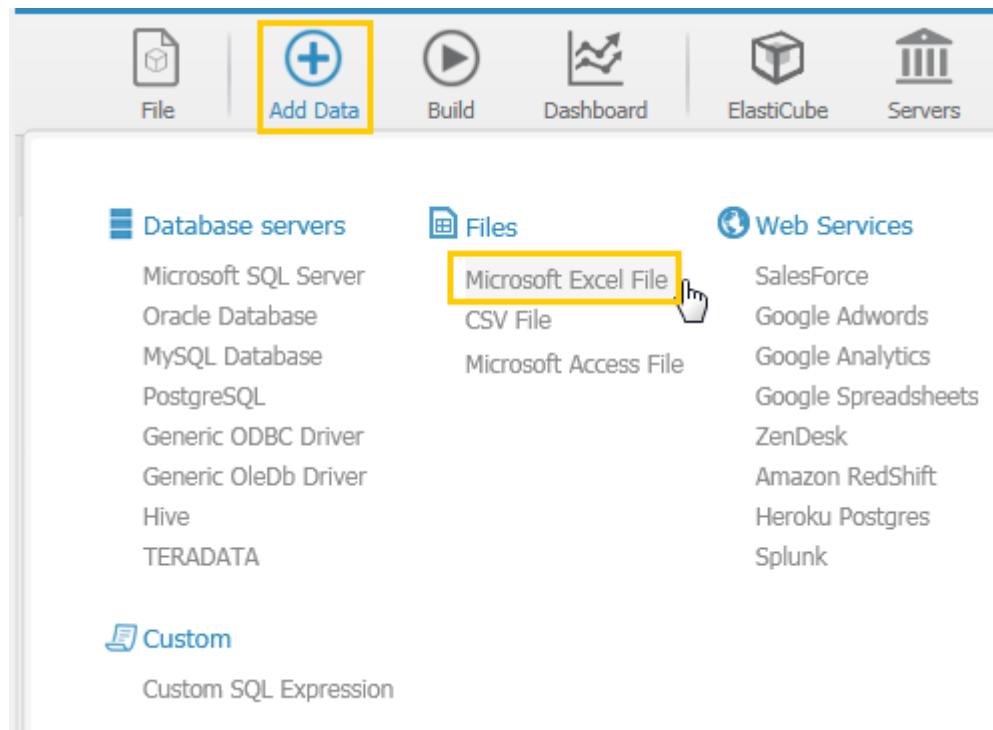
The ElastiCube Manager enables easy and quick access to tables contained within Excel spreadsheets.

You can add multiple Excel files to your ElastiCube. A typical use case for importing multiple files is when you want to split a large file into multiple smaller files.

Note: See [Preparing Your Excel](#) for see tips on how to prepare your Excel files before adding them to your schema in the ElastiCube.

The steps below detail how to connect to this type of data source.

1. Click **Add Data** in the top menu of the ElastiCube Manager.
2. Under the **Files** category select **Microsoft Excel File**.



3. In the window that opens, locate the Excel file.
4. Select the Excel file and click **OK**.
5. In the window that opens, you have the following options:

- ▶ **Fetch cells from:** You can fetch cells from either a sheet or a named range (provided you have named ranges in your file).
- ▶ **Available Sheets:** Select a sheet contained in the spreadsheet.
- ▶ **Range:** Select a **Static Range** or **Take range from whole sheet**.

For a **Static Range**, enter two cells, each with a leading \$ sign and a colon as a delimiter. Hit Enter to preview the selection. Selecting **Static Range** option enables you to select a specific range of data in the sheet. Data needs to be in a table structure, starting at the top left cell of the range, with field names as the first row.

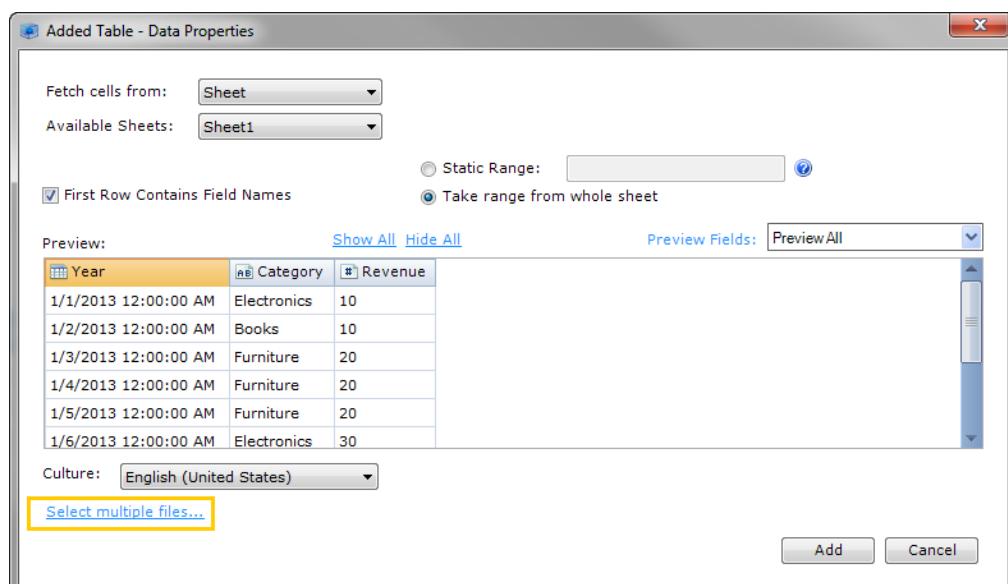
For a static range between cells A1 and E10, type in **\$A1:\$E10**

A	B	C	D	E	F	G	H
1							
2							
3							
4							
5	Date	Total	Campaign	CampaignType	Region	AdGroup	Keyword
6	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	all music	blues songs
7	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	all music	download jazz
8	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	all music	free classical music online
9	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	all music	online classical music free
10	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	bing music	famous jazz musicians
11	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	bing music	free blues radio
12	7/1/2012 Total		Band - Search	Test	Unknown	cheap music downloads	stl blues
13	7/1/2012 Total		MUSIC	Test	EMEA	classical music online	what is hip hop
14	7/1/2012 Total		MUSIC	Test	EMEA	country music	most famous classical music
15	7/1/2012 Total		MUSIC	Test	EMEA	country music videos	blues guitar chords
16	7/1/2012 Total		MUSIC	Test	EMEA	download free music	classic music
17	7/1/2012 Total		MUSIC	Test	EMEA	download free music	download classical music
18	7/1/2012 Total		MUSIC	Test	EMEA	download mp3 music	psychedelic trance music
19	7/1/2012 Total		MUSIC	Test	EMEA	bing music	famous jazz musicians
20	7/1/2012 Total		MUSIC	Test	EMEA	bing music	house music mix
21	7/1/2012 Total		MUSIC	Test	EMEA	download music	good classical music
22	7/1/2012 Total		MUSIC	Test	EMEA	downloadable music	hip hop music
23	7/1/2012 Total		MUSIC	Test	EMEA	downloading free music	hip hop new releases
24	7/1/2012 Total		MUSIC	Test	EMEA	downloading free music	hip hop news and gossip
25	7/1/2012 Total		MUSIC	Test	EMEA	free country music	hip hop songs free download

Selecting **whole sheet** will import all data within the sheet. Data within the sheet needs to be in a table structure starting at the top left corner (cell A1) with the field names as the first row.

	A	B	C	D	E	F	G
1	Date	Total	Campaign	CampaignType	Region	AdGroup	Keyword
2	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	all music	blues songs
3	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	all music	download jazz
4	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	all music	free classical music online
5	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	all music	online classical music free
6	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	bing music	famous jazz musicians
7	7/1/2012 Total		Europe LIVE - Search	Live	EMEA	bing music	free blues radio
8	7/1/2012 Total		Band - Search	Test	Unknown	cheap music downloads	stl blues
9	7/1/2012 Total	MUSIC		Test	EMEA	classical music online	what is hip hop
10	7/1/2012 Total	MUSIC		Test	EMEA	country music	most famous classical music
11	7/1/2012 Total	MUSIC		Test	EMEA	country music videos	blues guitar chords
12	7/1/2012 Total	MUSIC		Test	EMEA	download free music	classic music
13	7/1/2012 Total	MUSIC		Test	EMEA	download free music	download classical music
14	7/1/2012 Total	MUSIC		Test	EMEA	download mp3 music	psychedelic trance music
15	7/1/2012 Total	MUSIC		Test	EMEA	bing music	famous jazz musicians
16	7/1/2012 Total	MUSIC		Test	EMEA	bing music	house music mix
17	7/1/2012 Total	MUSIC		Test	EMEA	download music	good classical music
18	7/1/2012 Total	MUSIC		Test	EMEA	downloadable music	hip hop music
19	7/1/2012 Total	MUSIC		Test	EMEA	downloading free music	hip hop new releases
20	7/1/2012 Total	MUSIC		Test	EMEA	downloading free music	hip hop news and gossip

- ▶ **Preview:** Shows a preview of data contained in the sheet.
- ▶ **First Row Contains Field Names:** Enables you to specify table column names based on the header in the first row of the spreadsheet.
- ▶ Click **Select multiple files...**to import multiple files.



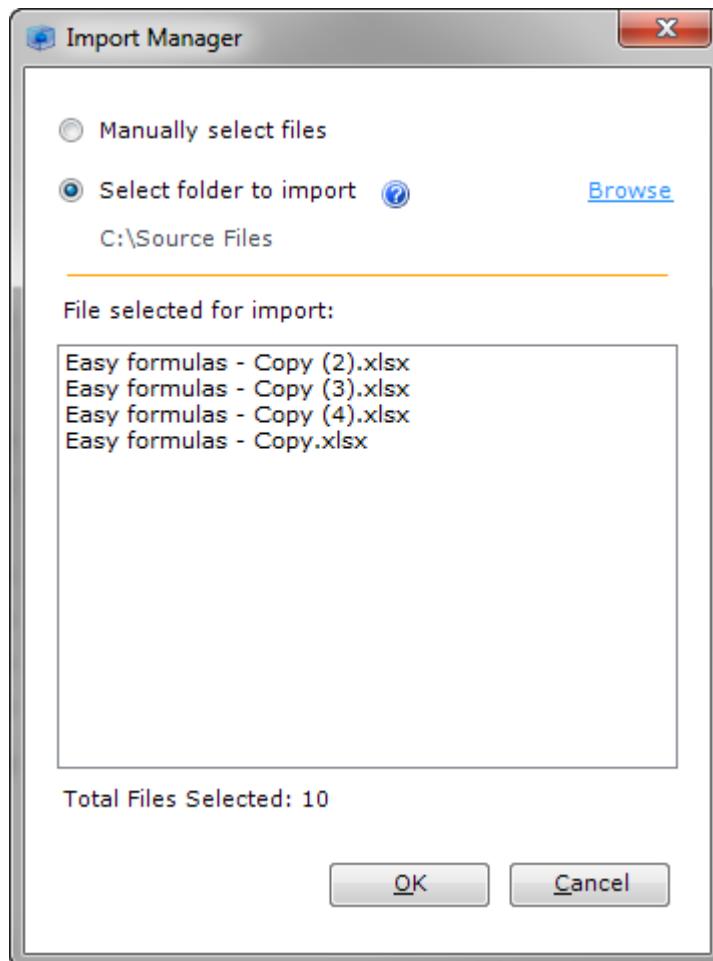
Note: Make sure that all the files adhere to the same structure, including the presence or absence of headers in the first row.

Files will be added to the table based on their file names in alphabetical order.

Select the **Select folder to import** option. All the Excel files in



the folder are added to the selected files list.



Click **OK**.

6. Click **Add**.

Connecting to Facebook

The Sisense Facebook connector is a standalone connector that allows you to import data from various Facebook APIs into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect to Facebook through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to Facebook's APIs. To obtain a connection string, you will need to create a Facebook developers account.

Once you have connected to Facebook, you can import a variety of tables from multiple Facebook APIs.

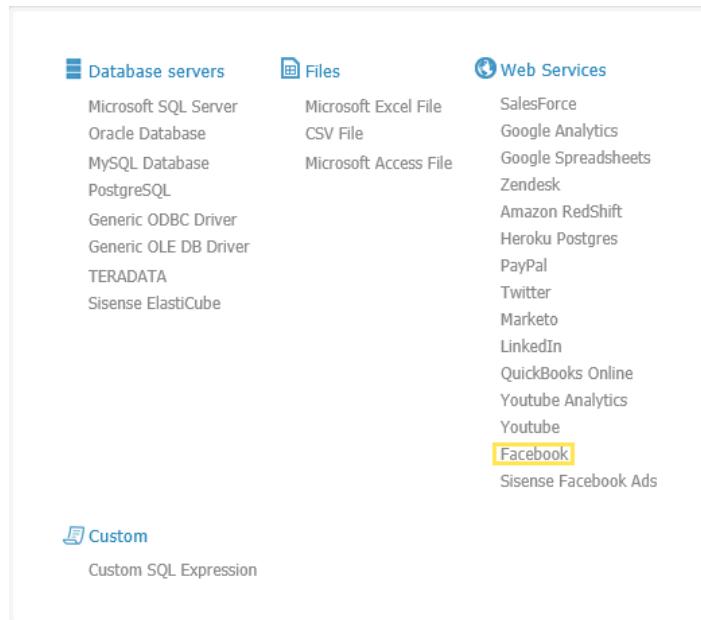
This section describes how to install the Facebook connector, how to connect to Facebook with a connection string, and what tables you can import into the ElastiCube Manager:

Installing the Facebook Connector

Sisense provides the Facebook connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Facebook connector:

1. [Download](#) the Facebook installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The Facebook connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.



Connecting to the Facebook REST API

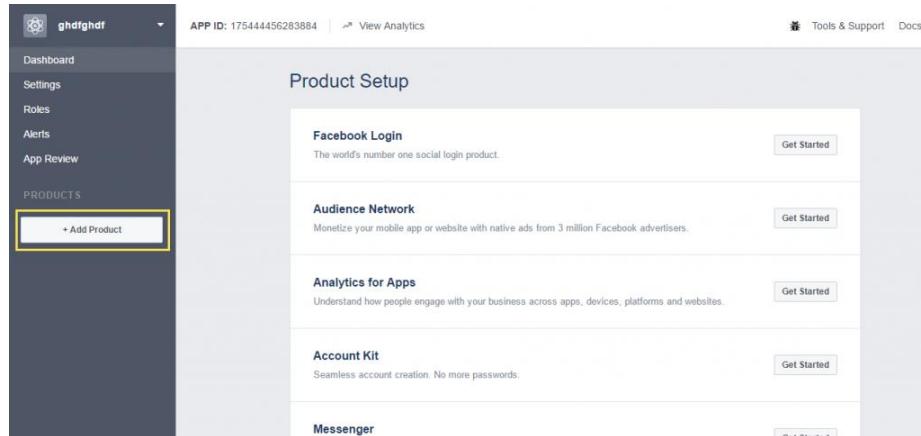
To access Facebook's REST API from Sisense, you must provide Sisense with valid OAuth Facebook credentials in a connection string. These credentials are provided by Facebook when you create an application.

Creating an App

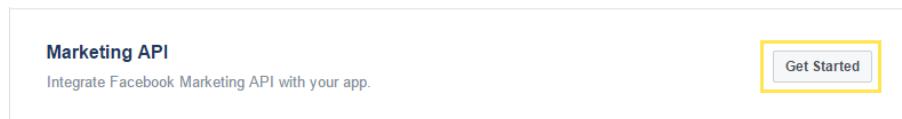
To obtain the OAuth client credentials, follow the steps below:

1. Log into Facebook and navigate to <https://developers.facebook.com/apps>.
2. Click + Add a New App.
3. Define your app's name and click **Create App ID**. The Product Setup page is displayed.

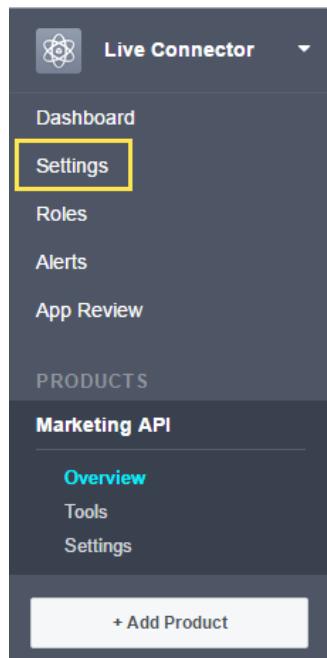
4. Click **Add Product**.



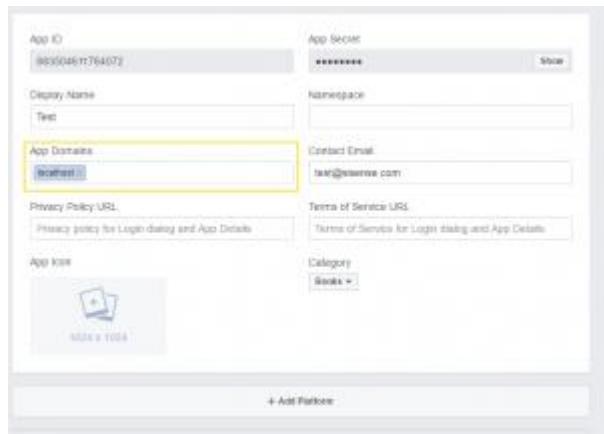
5. For the Marketing API, click **Get Started**.



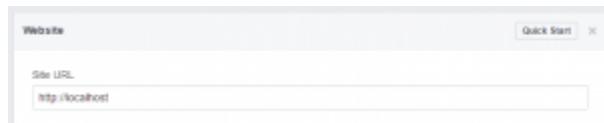
6. Open the Settings page.



7. In the Settings page, the OAuth client credentials, **App Id** and **App Secret**, are displayed.



8. In the **App Domains** field, enter localhost.
9. In the Settings page, click **+ Add Platform** and select **Website**.
The Website area is added to the Settings page.
10. In the **Site URL** field, enter http://localhost. This value is represents the callback URL in your connection string where the user will return with the token that verifies that they have granted your app access.



11. In the bottom-right corner of the Settings page, click **Save Changes**. Facebook requests that you authenticate your account.
12. Enter your password to authenticate your account and click **OK**. Your app is saved.
13. After you have authenticated your account, you must set permissions for the app to retrieve your data. Open

the [Facebook Explorer](#) to set your permissions.



14. In Facebook Explorer, click Get Token > Get User Access Token.

Select Permissions v2.8

User Data Permissions

<input checked="" type="checkbox"/> email	<input type="checkbox"/> user_hometown	<input type="checkbox"/> user_religion_politics
<input type="checkbox"/> publish_actions	<input type="checkbox"/> user_likes	<input type="checkbox"/> user_status
<input type="checkbox"/> user_about_me	<input type="checkbox"/> user_location	<input type="checkbox"/> user_tagged_places
<input checked="" type="checkbox"/> user_birthday	<input type="checkbox"/> user_photos	<input type="checkbox"/> user_videos
<input type="checkbox"/> user_education_history	<input type="checkbox"/> user_posts	<input type="checkbox"/> user_website
<input checked="" type="checkbox"/> user_friends	<input type="checkbox"/> user_relationship_details	<input type="checkbox"/> user_work_history
<input type="checkbox"/> user_games_activity	<input type="checkbox"/> user_relationshipships	

Events, Groups & Pages

<input type="checkbox"/> ads_management	<input type="checkbox"/> pages.messaging	<input type="checkbox"/> read_page_mailboxes
<input type="checkbox"/> ads_read	<input type="checkbox"/> pages.messaging_payments	<input type="checkbox"/> rsvp_event
<input type="checkbox"/> business_management	<input type="checkbox"/> pages.messaging_phone_number	<input type="checkbox"/> user_events
<input type="checkbox"/> manage_pages	<input type="checkbox"/> pages.messaging_subscriptions	<input type="checkbox"/> user_managed_groups
<input type="checkbox"/> pages_manage_cta	<input type="checkbox"/> pages.show_list	<input type="checkbox"/> pages.manage_instant_articles
<input type="checkbox"/> publish_pages		
<input type="checkbox"/> user_actions.books	<input type="checkbox"/> user_actions.music	<input type="checkbox"/> user_actions.video
<input type="checkbox"/> user_actions.fitness	<input type="checkbox"/> user_actions.news	

Other

<input type="checkbox"/> read_audience_network_insights	<input type="checkbox"/> read_custom_friendlists	<input type="checkbox"/> read_insights
---	--	--

Public profile included by default

Get Access Token **Clear** **Cancel**

15. Select the relevant permissions you need and click Get Access Token. Your permissions are now set and you can request the relevant data from Facebook.

Note: For apps to manage ads, you must be granted the app ads_management or ads_read extended permission. Use ads_read if you only need Ad Insights API access, to pull

reporting information. Use ads_management when you need to read, create and update ads. For more information about permissions, click [here](#).

Accessing the Facebook API

Facebook has three levels of access to its Marketing API, which determine how you can use and access data from your Facebook app:

- ▶ **Development:** Try and test apps with the API.
- ▶ **Basic:** Test, iterate, build apps with up to 25 external accounts.
- ▶ **Standard:** Support unlimited accounts and be nominated to Facebook Marketing Partner program.

When you create a Facebook, you are assigned a Development level of access by default. Each access level has its own account and rate limitations. You can modify your level of access by submitting a request to Facebook. For more information about these limitations and upgrading your access, click [here](#).

Authenticating through Facebook

After setting the following connection properties, you are ready to connect:

- ▶ **OAuthClientId:** Set this to the App ID in your app settings.
- ▶ **OAuthClientSecret:** Set this to the App Secret in your app settings.
- ▶ **CallbackURL:** Set this to the Return URL in your app settings.
- ▶ **InitiateOAuth:** Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken connection property.

Note: When switching accounts, you must sign out of your account and then sign it with the new account.

When you connect the Sisense connector completes the OAuth process:

1. Extracts the access token from the CallbackURL and authenticates requests.
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

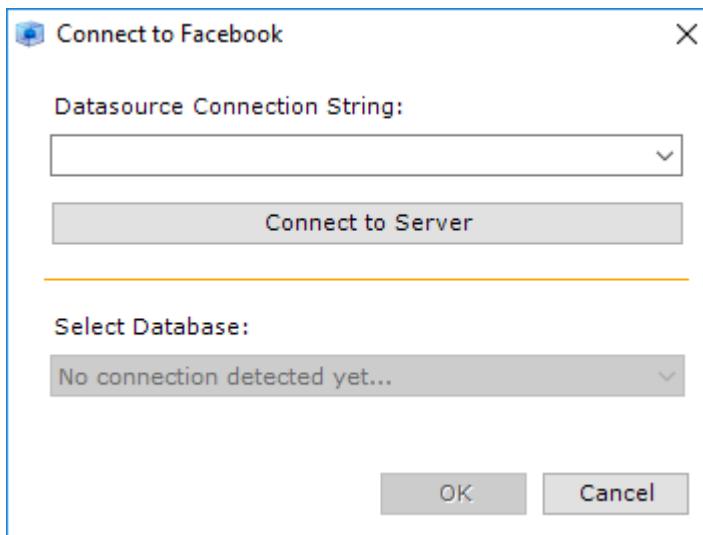
Adding Facebook Tables to Your ElastiCube

Sisense uses connection strings to connect to Facebook and import data into the ElastiCube Manager. Each connection string contains authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add Facebook data:

1. In ElastiCube Manager, click **Add Data** and then, **Facebook**.

The Connect to Facebook window is displayed.



2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Facebook is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Facebook and displays a list of tables available for you to import. For a list of tables you can import, see Facebook Tables.
5. Select the relevant tables and click **Add**. The tables are displayed in the ElastiCube Manager.

Facebook Tables

The table below describes the tables you can import from various Facebook APIs. Facebook limits the number of calls and the amount of data that you can retrieve from their APIs. These limitations depend on which Facebook API you are using and your account type. When you reach a limit, Sisense returns the error provided by Facebook. For example, if you receive any of the following error codes, this usually indicates that you have reached one of Facebook's limitations:

- ▶ error code = 4, CodedException
- ▶ error code = 32, CodedException
- ▶ error code = 17, User request limit reached

If you encounter one of these errors, refer to the Facebook API that exposed the table for more details regarding that API's limitations.

Note: Some Facebook tables require custom SQL, the table below provides examples of custom SQL statements you can use to access the relevant table.

Available Tables

Name	Description
Comments	<p>Query the Comments for a Target. Comments may also be inserted based on a Target or deleted based on Id.</p> <p>When querying comments, either the Target or the Id of the comment must be specified. For example, to retrieve all the comments about a specific post, your SELECT statement could look something like this:</p> <pre>SELECT * FROM Comments WHERE Target = '15526475270_410830705612736'</pre> <p>Alternatively, you can specify the Id to retrieve a specific comment. For example:</p> <pre>SELECT * FROM Comments WHERE Id = '15526475270_410830705612736_5193593'</pre> <p>The following is an example in which dynamic targets have been implemented:</p> <pre>SELECT * FROM Comments where target in (SELECT ID FROM Posts where target in (select ID from Pages))</pre>

Name	Description
Likes	<p>Query the Likes for a Target. Alternatively, lists Pages that the specified User or Page Likes. Authentication is required to use this table.</p> <p>The following is an example in which dynamic targets have been implemented:</p> <pre>SELECT * FROM Likes where target in (SELECT ID FROM Posts where target in (select ID from Pages))</pre>
Milestones	<p>Query a list of Milestones for the given Page. Milestones may only be inserted, updated, or deleted when authenticating as the target Page. Unless you are using the option Authenticate As Page, you need to supply a Target when retrieving milestones. For instance:</p> <pre>SELECT * FROM Milestones WHERE Target='facebook'</pre>
Posts	<p>Query the Posts for a Target based on either the Target or Id. Posts can also be inserted based on a Target, or deleted based on Id. This table requires authentication.</p> <p>The following is an example in which dynamic targets have been implemented:</p> <pre>SELECT * FROM Posts where target in (SELECT ID FROM Pages)</pre>
Users	Query Users by SearchTerms or Id.

Available Views

Name	Description
AdAccounts	<p>Query the Ad Accounts available for a User. Accessing Ad Account information requires the ads_read permission.</p>
AdCreatives	<p>Query information about an Ad Creative or the Ad Creatives on a specific Ad Account, Ad Set, or Ad. Accessing Ad Creative information requires the ads_read permission.</p> <p>The following is an example in which dynamic targets have been implemented:</p> <pre>SELECT * FROM AdCreatives where target in (select ID from AdAccounts);</pre>
AdInsights	<p>Query an Ad Report. Accessing Ad Report information requires the ads_read permission.</p> <p>When requesting AdInsights, a Target must be specified. This indicates what element to retrieve the insights from. It can be an AdAccount, Campaign, AdSet, or an Ad. For example:</p> <pre>view source SELECT * FROM AdInsights WHERE Target = 'act_123456'</pre> <p>A date range can be specified using DateStart and DateEnd, or DatePreset must also be specified. For example:</p> <pre>view source SELECT DateStart, DateEnd, AdAccountId, Spend, Impressions FROM AdInsights WHERE Target =</pre>

Name	Description
	<p>'act_123456' AND DateStart >= '01/01/2015' AND DateEnd <= '03/31/2015'</p> <p>view source SELECT DateStart, DateEnd, AdAccountId, Spend, Impressions FROM AdInsights WHERE Target = 'act_123456' AND DatePreset='last_90_days'</p> <p>The available values for DatePreset are:</p> <ul style="list-style-type: none"> • today • yesterday • this_week • last_week • last_7_days • last_14_days • last_28_days • last_30_days • last_90_days • this_month • last_month • this_quarter • last_3_months <p>The TimeIncrement can be used to specify how many days should be included in each report row. For instance:</p> <p>view source SELECT DateStart, DateEnd, AdAccountId, Age, Spend, Impressions FROM AdInsights WHERE Target = 'act_123456' AND DatePreset='last_90_days' AND TimeIncrement='7' view source SELECT DateStart, DateEnd, AdAccountId, Age, Spend, Impressions FROM AdInsights WHERE Target = 'act_123456' AND DatePreset='last_90_days' AND TimeIncrement='monthly'</p> <p>The Level column can be used to specify what level insights are retrieved at. This can be set to</p>

Name	Description
	<p>ad,adset,campaign, or account. For example: view source</p> <pre>SELECT DateStart, DateEnd, AdAccountId, Age, Spend, Impressions FROM AdInsights WHERE Target = 'act_123456' AND Level='campaign' The following is an example in which dynamic targets have been implemented:</pre> <p>SELECT Target, DateStart, DateEnd, AdAccountId, AdAccountName, CampaignId, CampaignName, AdSetId, AdSetName, AdId, AdName, Placement, Clicks, CPC, CPM, CPP, CTR, Frequency, Impressions, Reach, Spend, TotalActions, TotalUniqueActions, UniqueClicks, UniqueCTR, WebsiteClicks FROM AdInsights where target in (SELECT ID FROM AdAccounts) and level='ad' and DatePreset='last_7_days' and TimeIncrement=1</p>
AdInsightsActions	<p>Query an Ad Report. Accessing Ad Report information requires the ads_read permission.</p> <p>AdInsightsActions represents a breakdown of the Actions column from the AdInsights. See AdInsights for general information on querying AdInsights.</p> <p>When requesting AdInsightsActions, a Target must be specified. This indicates what element to retrieve the insights from. It can be an AdAccount, Campaign, AdSet, or an Ad. For instance:</p> <pre>SELECT * FROM AdInsights WHERE Target = 'act_123456'</pre> <p>In comparison to all of the available selection criteria from AdInsights, only the breakdowns listed below are available. Other than that, ActionAttributionWindows is available to input a comma separated list of attribution windows. For instance:</p> <pre>SELECT * FROM AdInsightsActions WHERE Target='act_123456' AND level='ad' AND</pre>

Name	Description
	ActionAttributionWindows='1d_view,7d_view,28d_click'
Columns	
Age:	The age range for the metrics in this row. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.
Country:	The country for the metrics in this row. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.
FrequencyValue:	The number of times an ad in your Reach and Frequency campaign was served to each person. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.
Gender:	The gender for the metrics in this row. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.
HStatsByAdvertiserTZ:	Time period over which the stats were taken for the advertiser. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.
HStatsByAudienceTZ:	Time period over which the stats were taken for the audience. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.
ImpressionDevice:	The devices used to view the Ad. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.
Placement:	The placement of the Ad on the screen. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.
PlatformPosition:	The position on the platform.

Name	Description
	<p>ProductId: The product Id advertised in the Ad. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.</p> <p>PublisherPlatform: The platforms the ads were published on.</p> <p>Region: The region someone viewed the Ad from. This is a breakdown column and selecting this column will cause results to be further broken down by this metric.</p> <p>The following is an example in which dynamic targets have been implemented:</p> <pre>select Target, DatePreset, DateStart, DateEnd, TimeIncrement, Level, ActionAttributionWindows, AdAccountId, AdAccountName, CampaignId, CampaignName, AdSetId, AdSetName, AdId, AdName, ActionType, ActionValue, Action1dClick, Action1dView, Action7dClick, Action7dView, Action28dClick, Action28dView, ActionCarouselCardId, ActionCarouselCardName, ActionDestination, ActionDevice, ActionReaction, ActionTargetId, ActionVideoSound, ActionVideoType FROM AdInsightsActions where target in (SELECT ID FROM AdAccounts) and level='ad' and DatePreset='today' and TimeIncrement=1 and ActionAttributionWindows='1d_view,1d_click,7d_view,7d_c lick,28d_view,28d_click'</pre>
Ads	Query information about an Ad or the Ads in a specific Ad Set, Campaign, or Ad Account. Accessing Ad Information requires the ads_read permission. The following is an example in which dynamic targets have been

Name	Description
	<p>implemented:</p> <pre>select * from Ads where target in (SELECT ID FROM AdAccounts) and AdStatus='ACTIVE'</pre>
AdScheduledReports	<p>Query the Scheduled Reports for a given Ad Account. Accessing Scheduled Report information requires the ads_read permission.</p>
AdSets	<p>Query information about an Ad Set or the Ad Sets on a specific Campaign or Ad Account. Accessing Ad Set information requires the ads_read permission.</p>
Albums	<p>Query Albums associated with a Target. Accessing Album information typically requires the user_photos permission.</p>
Applications	<p>Query the Application specified by the Id.</p>
Books	<p>Query the Books a User is interested in. Accessing Book information typically requires the user_books permissions.</p>
Campaigns	<p>Query information about a Campaign or the Campaigns on a specific Ad Account. Accessing Campaign information requires the ads_read permission.</p> <p>For example:</p> <pre>SELECT * FROM Campaigns WHERE Target = 'act_123456'</pre> <p>The following is an example in which dynamic targets have been implemented:</p>

Name	Description
	SELECT * FROM Campaigns where Target in (SELECT ID FROM AdAccounts) and Status='ACTIVE'
Events	Query the Events for a Target based on either the Target or SearchTerms. May require the user_events permission.
Friends	Query Friends of the authenticated User or Target User. Requires the user_friends permission. A Target may be specified to request Friend information for, but Friends may only be retrieved for the authenticated User or Friends of the authenticated User that use the same Facebook app.
Games	Query the Games a User is interested in. Accessing Game information may require the user_likes and user_interests permissions.
Groups	Query the Groups based on the supplied SearchTerms, Id, or Target. Groups may require the user_groups permission.
GroupMemberShip s	Query the Groups based on the supplied GroupId. Groups may require the user_groups permission. GroupMemberships in Facebook are the individual members of a given group. GroupMemberships are currently only exposed as a view. When selecting group memberships, a GroupID must be specified. For example, to retrieve the members of a group: SELECT * FROM GroupMemberships WHERE GroupId = 'GroupId'

Name	Description
	<pre>SELECT * FROM GroupMemberships WHERE GroupId in (select ID from Groups)</pre>
InsightsByConsumptionType	<p>Allows retrieval of insights by consumption type.</p> <pre>SELECT * FROM InsightsByConsumptionType where target ='xxxxxxxx' and InsightName = 'PAGE_CONSUMPTIONS_BY_CONSUMPTION_TYPE' and Period = 'day'</pre> <p>The following are values for the parameters InsightName:</p> <p>PAGE_CONSUMPTIONS_BY_CONSUMPTION_TYPE PAGE_CONSUMPTIONS_BY_CONSUMPTION_TYPE_UNIQUE</p>
InsightsByFeedbackType	<p>Allows retrieval of insights by feedback type.</p> <pre>SELECT * FROM InsightsByFeedbackType where target ='xxxxxxxx' and InsightName = 'PAGE_NEGATIVE_FEEDBACK_BY_TYPE' and Period = 'day'</pre> <p>The following are values for the parameters InsightName:</p> <p>PAGE_NEGATIVE_FEEDBACK_BY_TYPE PAGE_NEGATIVE_FEEDBACK_BY_TYPE_UNIQUE PAGE_POSITIVE_FEEDBACK_BY_TYPE PAGE_POSITIVE_FEEDBACK_BY_TYPE_UNIQUE</p>

Name	Description
	<p>POST_NEGATIVE_FEEDBACK_BY_TYPE POST_NEGATIVE_FEEDBACK_BY_TYPE_UNIQUE</p>
InsightsByLikeSourceType	<p>Allows retrieval of insights by like source type.</p> <p>The following are values for the parameters InsightName:</p> <p>PAGE_FANS_BY_LIKE_SOURCE PAGE_FANS_BY_LIKE_SOURCE_UNIQUE PAGE_FANS_BY_UNLIKE_SOURCE_UNIQUE</p>
InsightsByPaidStatus	<p>Allows retrieval of insights by paid status.</p> <p>The following are values for the parameters InsightName:</p> <p>PAGE_IMPRESSIONS_BY_PAID_NON_PAID PAGE_IMPRESSIONS_BY_PAID_NON_PAID_UNIQUE POST_IMPRESSIONS_BY_PAID_NON_PAID POST_IMPRESSIONS_BY_PAID_NON_PAID_UNIQUE PAGE_POSTS_IMPRESSIONS_BY_PAID_NON_PAID PAGE_POSTS_IMPRESSIONS_BY_PAID_NON_PAID_UNIQUE</p> <p>The following is an example in which dynamic targets have been implemented:</p> <pre>SELECT EndTime as Date, Paid, Unpaid FROM InsightsByPaidStatus where target in (SELECT ID FROM Pages) and InsightName = 'PAGE_IMPRESSIONS_BY_PAID_NON_PAID' and Period = 'day'</pre>

Name	Description
InsightsByTabType	<p>Allows retrieval of insights by tab type</p> <p>The following are values for the parameters InsightName:</p> <ul style="list-style-type: none"> PAGE_TAB_VIEWS_LOGIN_TOP_UNIQUE PAGE_TAB_VIEWS_LOGIN_TOP PAGE_TAB_VIEWS_LOGOUT_TOP
LeadValues	<p>Query information about a lead. Accessing Lead Information requires the ads_read permission.</p>
Music	<p>Query the Music a User is interested in. Accessing Music requires the user_likes and user_interests permissions.</p>
Pages	<p>Query the Pages based on the supplied SearchTerms or Id.</p>
Permissions	<p>Query the Permissions the User has granted the current application.</p>
Photos	<p>Query Photos associated with a Target. Accessing Photo information typically requires the user_photos permission.</p>
Places	<p>Query the Places based on the supplied SearchTerms or Id. Places are stored as Pages in Facebook.</p>
ScheduledReportRuns	<p>Query a list of the recently completed runs of a given Scheduled Report. Accessing Scheduled Report information requires the ads_read permission.</p>
SimpleInsights	<p>Allows the retrieval of simple insights with a single value in the response.</p> <p>The following are values for the parameters InsightName:</p>

Name	Description
	PAGE_STORIES
	POST_STORIES
	POST_STORYTELLERS
	PAGE_IMPRESSIONS
	PAGE_IMPRESSIONS_UNIQUE
	PAGE_IMPRESSIONS_PAID
	PAGE_IMPRESSIONS_PAID_UNIQUE
	PAGE_IMPRESSIONS_ORGANIC
	PAGE_IMPRESSIONS_ORGANIC_UNIQUE
	PAGE_IMPRESSIONS_VIRAL
	PAGE_IMPRESSIONS_VIRAL_UNIQUE
	PAGE_IMPRESSIONS_FREQUENCY_DISTRIBUTION
	PAGE_IMPRESSIONS_VIRAL_FREQUENCY_DISTRIBUTIO
N	
	PAGE_ENGAGED_USERS
	PAGE_CONSUMPTIONS
	PAGE_CONSUMPTIONS_UNIQUE
	PAGE_PLACES_CHECKIN_TOTAL
	PAGE_PLACES_CHECKIN_TOTAL_UNIQUE
	PAGE_PLACES_CHECKIN_MOBILE
	PAGE_PLACES_CHECKIN_MOBILE_UNIQUE
	PAGE_NEGATIVE_FEEDBACK
	PAGE_NEGATIVE_FEEDBACK_UNIQUE
	PAGE_FANS_ONLINE
	PAGE_FANS_ONLINE_PER_DAY
	PAGE_FANS
	PAGE_FAN_ADDS
	PAGE_FAN_ADDS_UNIQUE
	PAGE_FAN_REMOVE
	PAGE_FAN_REMOVE_UNIQUE
	PAGE_VIEWS
	PAGE_VIEWS_UNIQUE

Name	Description
	PAGE_VIEWS_LOGIN
	PAGE_VIEWS_LOGIN_UNIQUE
	PAGE_VIEWS_LOGOUT
	PAGE_VIDEO_VIEWS
	PAGE_VIDEO_VIEWS_PAID
	PAGE_VIDEO_VIEWS_ORGANIC
	PAGE_VIDEO_VIEWS_AUTOPLAYED
	PAGE_VIDEO_VIEWS_CLICK_TO_PLAY
	PAGE_VIDEO_VIEWS_UNIQUE
	PAGE_VIDEO_REPEAT_VIEWS
	PAGE_VIDEO_COMPLETE_VIEWS_30S
	PAGE_VIDEO_COMPLETE_VIEWS_30S_PAID
	PAGE_VIDEO_COMPLETE_VIEWS_30S_ORGANIC
	PAGE_VIDEO_COMPLETE_VIEWS_30S_AUTOPLAYED
	PAGE_VIDEO_COMPLETE_VIEWS_30S_CLICK_TO_PLAY
	PAGE_VIDEO_COMPLETE_VIEWS_30S_UNIQUE
	PAGE_VIDEO_COMPLETE_VIEWS_30S_REPEAT_VIEWS
	PAGE_POSTS_IMPRESSIONS
	PAGE_POSTS_IMPRESSIONS_UNIQUE
	PAGE_POSTS_IMPRESSIONS_PAID
	PAGE_POSTS_IMPRESSIONS_PAID_UNIQUE
	PAGE_POSTS_IMPRESSIONS_ORGANIC
	PAGE_POSTS_IMPRESSIONS_ORGANIC_UNIQUE
	PAGE_POSTS_IMPRESSIONS_VIRAL
	PAGE_POSTS_IMPRESSIONS_VIRAL_UNIQUE
	POST_IMPRESSIONS
	POST_IMPRESSIONS_UNIQUE
	POST_IMPRESSIONS_PAID
	POST_IMPRESSIONS_PAID_UNIQUE
	POST_IMPRESSIONS_FAN
	POST_IMPRESSIONS_FAN_UNIQUE
	POST_IMPRESSIONS_FAN_PAID

Name	Description
	POST_IMPRESSIONS_FAN_PAID_UNIQUE POST_IMPRESSIONS_ORGANIC POST_IMPRESSIONS_ORGANIC_UNIQUE POST_IMPRESSIONS_VIRAL POST_IMPRESSIONS_VIRAL_UNIQUE POST_CONSUMPTIONS POST_CONSUMPTIONS_UNIQUE POST_ENGAGED_USERS POST_NEGATIVE_FEEDBACK POST_NEGATIVE_FEEDBACK_UNIQUE POST_VIDEO_AVG_TIME_WATCHED POST_VIDEO_COMPLETE_VIEWS_ORGANIC POST_VIDEO_COMPLETE_VIEWS_ORGANIC_UNIQUE POST_VIDEO_COMPLETE_VIEWS_PAID POST_VIDEO_COMPLETE_VIEWS_PAID_UNIQUE POST_VIDEO_VIEWS_ORGANIC POST_VIDEO_VIEWS_ORGANIC_UNIQUE POST_VIDEO_VIEWS_PAID POST_VIDEO_VIEWS_PAID_UNIQUE
TaggedBy	Query information about Posts, Statuses, Photos, and other entities that have tagged the User or Page. This view is a derivative of the Wall connection where only entries that have tagged the Target User or Page will be returned. In general it is only available for Pages.
Television	Query the TV shows a User is interested in. Accessing Television information may require the user_likes and user_interested permissions.
Videos	Query Videos from a Target. Normally requires the user_videos permission.

Name	Description
Wall	Query Posts from the Wall of a Target.

Note: For the following tables: InsightsByConsumptionType, InsightsByFeedbackType, InsightsByLikeSourceType, InsightsByPaidStatus, InsightsByTabType, SimpleInsights, the Object Insights period {day, week, month, lifetime} lifetime is not supported for all measures. For more information, see Facebook's [API documentation](#).

Dynamic Targets

Some Facebook tables allow you to define a target, such as an account, to retrieve specific data. This is used for some tables, such as AdInsights, where a target can be specified.

While you can define the value of the target parameter in your connection string, if you support multiple accounts, you can use a dynamic target to retrieve data from each of those accounts.

The following is an example of the a dynamic target as the value of Target:

```
SELECT Target, DateStart, DateEnd, AdAccountId,
AdAccountName, CampaignId, CampaignName, AdSetId, AdSetName,
AdId, AdName, Placement, Clicks, CPC, CPM, CPP, CTR,
Frequency, Impressions, Reach, Spend, TotalActions,
TotalUniqueActions, UniqueClicks, UniqueCTR, WebsiteClicks
FROM AdInsights where target in (SELECT ID FROM AdAccounts
) and level='ad' and DatePreset='last_7_days' and
TimeIncrement=1
```

Accumulative Builds

Sisense support accumulative builds for all numeric and dates data types. However, the data must be sorted before building the ElastiCube.

Connecting via ODBC Drivers

Open Database Connectivity

The **ODBC** provider allows access to a data source via an installed driver on the operating system regardless of the architecture of the data source.

In Sisense, it serves as a connector to data providers, such as MS Sql, MySql, Oracle, Hive and so forth.

DSN vs. Connection-String (DSN-Less)

A connection to the provider is configured by a connection-string.

Microsoft windows allows you to store the connection string properties in its registry for further use. It supplies an ODBC configuration wizard and requests an identifying Data Source Name (DSN).

When you add an ODBC table in the ElastiCube Manager, the connection configuration dialog will provide you with two options:

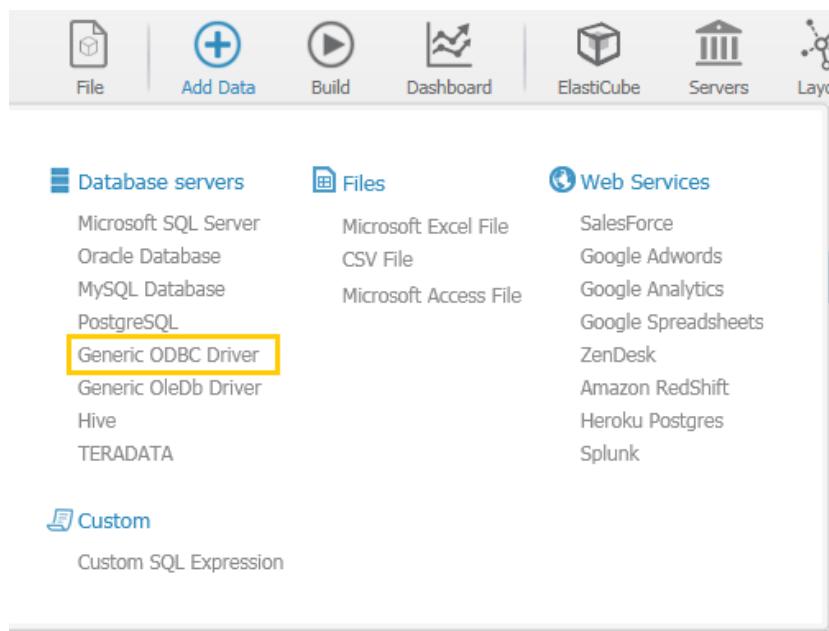
- ▶ DSN: Choosing a data source name.
- ▶ Connection String (DSN-Less): Typing the actual connection string

Adding a Table to the ElastiCube Manager:

1. Click **Add Data** in the top menu of the ElastiCube Manager.
2. Under the **Database servers** category, select **Generic ODBC Driver**.
3. From the ODBC Connectivity Properties dialog choose **DSN or Connection String (DSN-Less)**.

- ▶ **DSN:** Choose one of the existing DSNs from the drop-down list (each item is represented by its name), or click **Add DSN**...to open the **Create New Data Source** window. In this window, you'll be asked to select one of the installed ODBC drivers, select a name for the DSN, and configure your connection using the wizard steps. At the end, the newly added DSN will be added to the drop-down list as the selected item.
- ▶ **Connection String:** Type the connection string in the input box. For known connection strings, click the Help link to navigate to [Connection Strings](#) (External site from Sisense). Click **Test Connection** to test the connection. Click **OK** to move to the final dialog. From the tables selection dialog, select tables that you want to add and click **Add** to complete the wizard.

Note: You may need to update the executed table query.



Setting the table query (Troubleshoot)

ODBC and OLE DB are generic drivers; therefore the SQL syntax being executed against the actual data source may vary from one provider to another.

Sisense sets by default the standard SQL for the ODBC/OLE DB tables which covers most of today's known data providers. For cases where non-standard SQL must be supplied, the table addition dialog provides a quick preview of the current SQL query and an edit button that opens the **Query Properties** dialog.

In **Query Properties**, you can set the following options:

- ▶ **Auto Query Structure:** Choosing the proper SQL syntax delimiter.
- ▶ **Manual Query Input:** Type the desired query.

Note

*The automatic options can be applied on the current table or on all tables, while the manual option defines a single table and can only be applied on the current table. **Apply All** and **Apply Table** will be enabled according to the selection made.*

Connecting via OLE DB Drivers

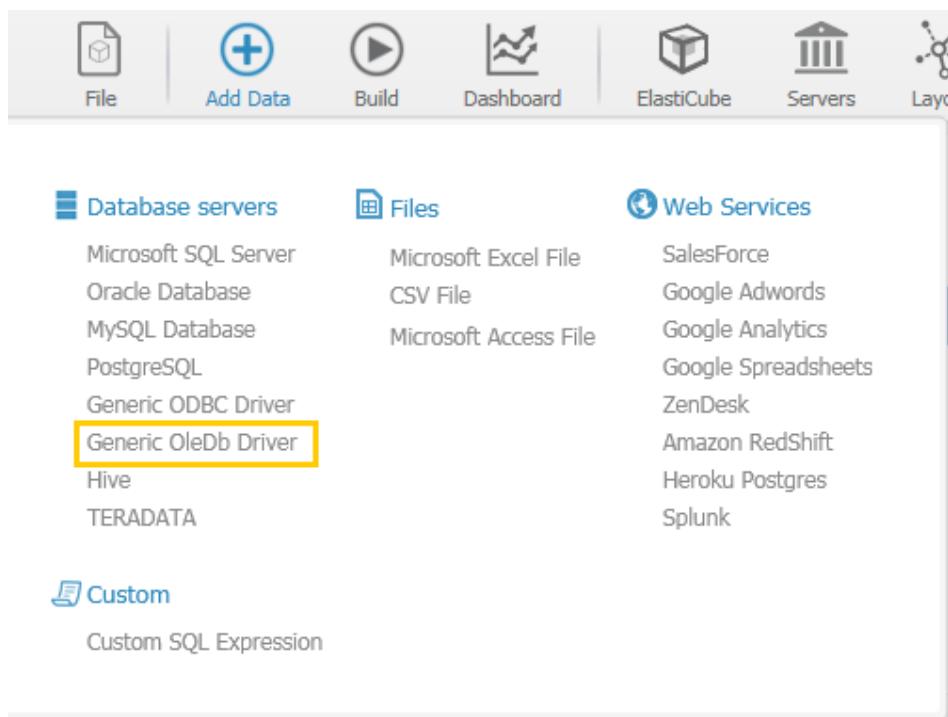
Object Linking and Embedding Database

The **OLE DB** provides access to a data source via an installed driver on the operating system regardless of the architecture of the data source.

In Sisense, it serves as a connector to data providers that are not included in the common providers list, such as MS Sql, MySql, Oracle and so forth.

Adding a table to the ElastiCube manager:

1. Click **Add Data** on the top menu of the ElastiCube Manager.
2. Under the **Database servers** category select **Generic OleDb Driver**.

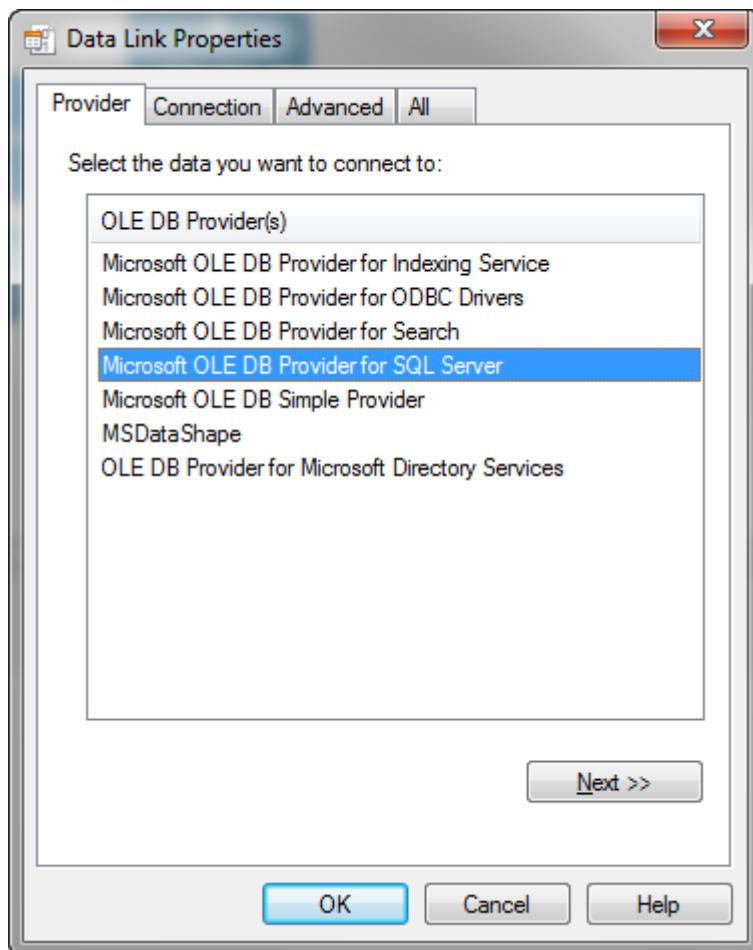


The screenshot shows the Sisense ElastiCube Manager interface. At the top, there is a navigation bar with icons for File, Add Data (highlighted in blue), Build, Dashboard, ElastiCube, Servers, and Layout. Below the navigation bar, there are three main categories: Database servers, Files, and Web Services. The Database servers category is expanded, showing a list of data sources including Microsoft SQL Server, Oracle Database, MySQL Database, PostgreSQL, Generic ODBC Driver, and Generic OleDb Driver. The 'Generic OleDb Driver' option is highlighted with a yellow box. Other options like Hive and TERADATA are also listed. The Files category lists Microsoft Excel File, CSV File, and Microsoft Access File. The Web Services category lists SalesForce, Google Adwords, Google Analytics, Google Spreadsheets, ZenDesk, Amazon RedShift, Heroku Postgres, and Splunk. At the bottom, there is a Custom section for 'Custom SQL Expression'.

Database servers	Files	Web Services
Microsoft SQL Server	Microsoft Excel File	SalesForce
Oracle Database	CSV File	Google Adwords
MySQL Database	Microsoft Access File	Google Analytics
PostgreSQL		Google Spreadsheets
Generic ODBC Driver		ZenDesk
Generic OleDb Driver		Amazon RedShift
Hive		Heroku Postgres
TERADATA		Splunk

Custom
Custom SQL Expression

3. In the **Data Link Properties** dialog, select a connection by choosing the relevant OLE DB driver and the connection properties.



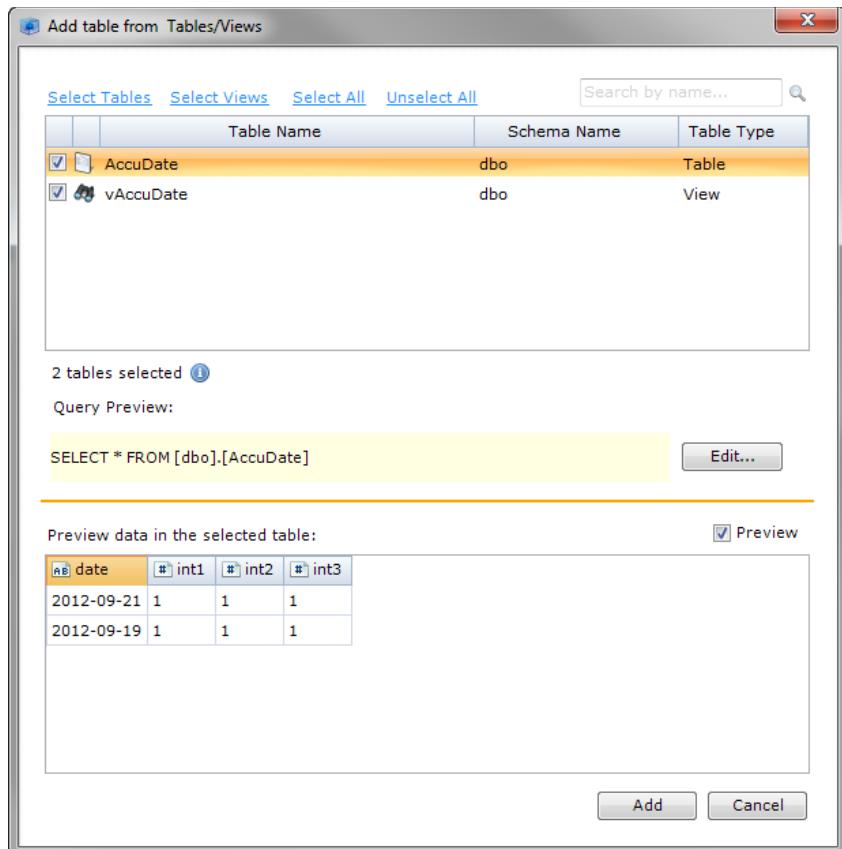
 **Note**

When the data source server requires a login credentials, you must select the Allow saving password checkbox.

Follow all wizard steps and click OK to move to the final dialog.

4. From the tables selection dialog, select tables that you want to add and click **Add** to complete the wizard.

Note: You may need to update the executed table query described below.



Setting the Table Query (Troubleshoot)

ODBC and OLE DB are generic drivers; therefore the SQL syntax being executed against the actual data source may vary from one provider to the other.

Sisense sets by default the standard SQL for the ODBC/OLE DB tables which covers most of the known data providers. For cases where non-standard SQL must be supplied, the table addition dialog provides a quick preview of the current SQL query and an edit button which opens the *Query Properties* dialog.

In the **Query Properties** dialog, two options are available:

- ▶ **Auto Query Structure:** Choose the proper SQL syntax delimiter
- ▶ **Manual Query Input:** Type the desired query.

Note: The automatic options can be applied on the current table or on all tables, while the manual option defines a single table and can only be applied on the current table. The **Apply All** and **Apply Table** buttons will be enabled according to the selection made.

Misc Troubleshooting:

- ▶ When an ODBC table is added by a DSN connection, the DSN must exist on the target ElastiCube Server machine, or else the server will fail to connect and import data from that table. In general, setting a connection by a DSN (not by connection string) is not recommended practice.
- ▶ Regarding ODBC only, ensure that the addressed ODBC driver platform (32 or 64 bit) matches the current installation of Sisense (Manager and Server). Platform mismatches will lead to connectivity errors.
- ▶ When setting an OLE DB connection, and the data source server requires login credentials, the Allow saving password checkbox must be selected.

Connecting to Gmail

The Sisense Gmail connector is a standalone connector that allows you to import data from Gmail's API into the ElastiCube Manager.

After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the Gmail APIs. To obtain a connection string, you will need to register a Gmail app.

Once you have connected to Gmail, you can import a variety of tables from the Gmail API.

This section describes how to install the Gmail connector, how to connect to Gmail with a connection string, and what tables you can import into the ElastiCube Manager:

Installing the Gmail Connector

Sisense provides the Gmail connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Gmail connector:

1. [Download](#) the Gmail installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.

The Gmail connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

Connecting to the Gmail REST API

To access Gmail's REST API from the ElastiCube Manager, you must provide valid Oauth Gmail credentials through a connection string.

These credentials are provided by Gmail when you register an application.

After you receive your credentials from Gmail, you can create the connection string and provide Sisense with it to connect to your data.

Registering an App

Follow the steps below to obtain the OAuth client credentials, the OAuthClientId and OAuthClientSecret:

1. Log into the Google API Console.
2. Click **Create Project** or select an existing project.
3. In the API Manager, click **Credentials > Create Credentials > OAuth Client Id**.
4. Select the application type. If you are making a Web application, select Web application. In the Authorized Redirect URIs box, enter the URL you want to be used as a trusted redirect URL, where the user will return with the token that verifies that they have granted your app access. If you are making a desktop application, select Other.
5. Click **Create**. The OAuthClientId and OAuthClientSecret are displayed.
6. Click **Library > Gmail API > Enable API**.

Authenticating through Gmail

After setting the following connection properties, you are ready to connect:

- ▶ **InitiateOAuth:** Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken connection property.
- ▶ **OAuthClientId:** Set this to the Client ID in your app settings.
- ▶ **OAuthClientSecret:** Set this to the Client secret in your app settings.
- ▶ **User:** Set this to the user of the Gmail account.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver then completes the OAuth process:

1. Extracts the access token from the callback URL and authenticates requests.
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Adding Gmail Tables to your ElastiCube

Sisense uses connection strings to connect to Gmail and import data into the ElastiCube Manager.

The connection string to connect to Gmail has the following structure:

`jdbc:Gmail:Property1=Value1;Property2=Value2;`

The following is an example of a Gmail connection string:

```
jdbc:gmail:user=xxxx@xxxxx.com;InitiateOAuth=GETANDREFRESH;o
AuthClientId=
xxxxxxxxxxxxxxxxxxxxxx;OAuthClientSecret=xxxxxxxxxxxxxxxxxxxxxx
xxxx';
```

Note: To switch between accounts, you need to delete the file OAuthsettings.txt file located at
.\\Users\\xxx\\AppData\\Roaming\\CData\\Gmail Data Provider.

To add Gmail data:

1. In ElastiCube Manager, click **Add Data** and then, **Gmail**. The Connect to Gmail window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Gmail is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Gmail and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**. The tables are displayed in the ElastiCube Manager.

Switching between Accounts

When you connect to the Gmail data source, Sisense saves your OAuth values in the file OAuthsettings.txt file located at
.\\Users\\xxx\\AppData\\Roaming\\CData\\Gmail Data Provider on your Sisense server. To connect to the Gmail data source with another user on the same machine, you must delete the OAuthsettings.txt file. Sisense will then generate a new file for that user.

Another option to support multiple users is to define the location and file name of an OAuthsettings file for each unique user in your connection string through the **OAuthSettingsLocation** parameter. When each user connects to the data source, Sisense generates the OAuth file with the file name you specify in the location you define. In the examples below, two users are allowed to access the Gmail data

source and for each user, Sisense generates a file that contains that user's OAuth values in the location defined in the string.

```
jdbc:Gmail:OAuthSettingsLocation=C:\Gmail\auth\john.txt;OAuth  
ClientId=11276856774486;OAuthClientSecret  
=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Vers  
ion=2.8;CallbackURL=http://localhost/;
```

```
jdbc:Gmail:OAuthSettingsLocation=C:\Gmail\auth\sally.txt;OAuth  
ClientId=11276856774486;OAuthClientSecret  
=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Vers  
ion=2.8;CallbackURL=http://localhost/;
```

In the example above, two OAuth files are created, one for John and one for Sally in the location C:\Gmail\auth\.

This is useful if you support many users who each need to access the Gmail data source.

Gmail Tables

Gmail's RESTful APIs expose the following Gmail tables that you can import into the ElastiCube Manager through the Sisense Gmail connector:

- ▶ Gmail/All Mail
- ▶ Gmail/Drafts
- ▶ Gmail/Important
- ▶ Gmail/Sent Mail
- ▶ Gmail/Spam
- ▶ Gmail/Starred
- ▶ Gmail/Trash
- ▶ Inbox
- ▶ Test Label

Limitations

1. The default Gmail table names contain a backslash; some also contain spaces. You can use square brackets to escape the table names for the default Gmail folders.

For example:

```
SELECT * FROM [Gmail/All Mail]
```

2. The default number of messages returned is 100. This can be changed by setting either LIMIT or MaxItems. If you wish to return all mail within a mailbox, specify a value of 0 or -1.
3. Aggregate functions not supported.

Connecting to Google Adwords

The ElastiCube Manager enables easy and quick access to tables contained within Google Adwords. The steps below describe how to connect to this type of data source.

Note

Importing data from Google Adwords is performed using the Sisense developer key and Sisense pays for the data imported from AdWords. For this reason your AdWords account must have access to the ‘Campaign Service’ and the user needs to have Standard Access.

To check the type of access, log into your AdWords account, then go to My Account > Account Access. A list of all associated campaigns and permission levels will be displayed.

Increasing the permission level can only be performed by a campaign administrator using the “invite user functionality”.

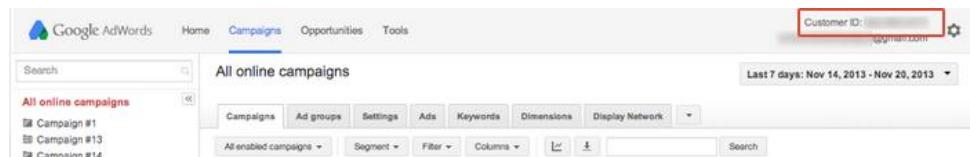
See [Campaign Management and Users](#) for more details.

In addition, you will need an MCC account. For more information, [click here](#).

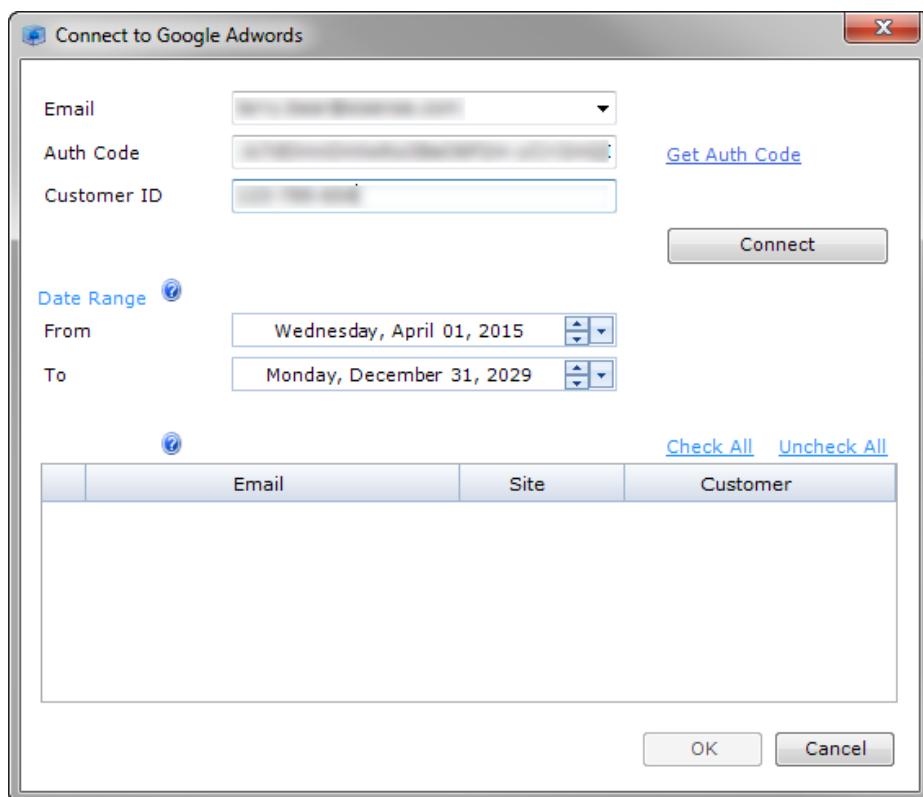
1. Click **Add Data** in the top menu of the ElastiCube Manager.
2. Under the **Web Services** category, select **Google AdWords**.
3. You will be prompted to enter the following information:
 - **Email:** Enter email associated with your Google Analytics account.
 - **Auth Code:** Google requires an authentication code to enable access from the ElastiCube to data associated with a Google Analytics account (site name and profile IDs). To attain the Auth code click on the link **Get Auth Code**. The browser will open and you will need to sign in to your Google Analytics account. After signing in, click on the ‘Allow access’ button. This will

generate a temporary code that you must paste into the **Auth Code** text box in the connection window.

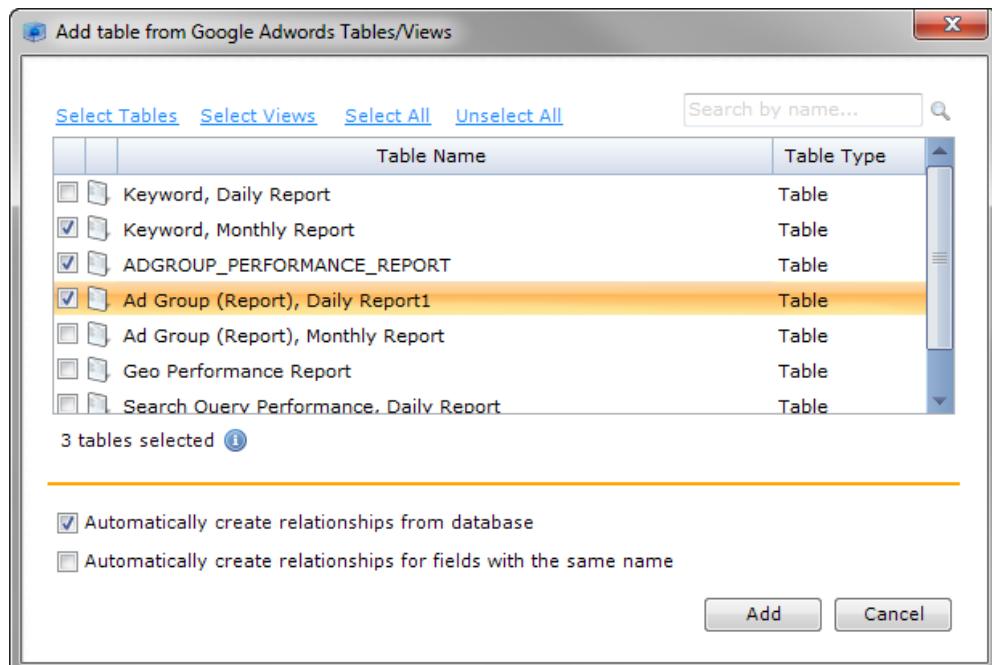
- **Customer ID:** Enter your Google AdWords Customer ID. Your Customer ID appears at the top corner of your AdWords screen (after logging into your AdWords account).



4. Click **Connect**.



5. Select the relevant client email associated with Adwords you want to work with and click **OK**.
All tables and views associated with the website will appear in a new window.



6. Select the checkbox next to each table you want to use.
7. Once all relevant tables are selected, click **Add**.

Note about possible build failures: If you upgrade your version of Sisense to 5.8.1, and the build fails, it may be due to the fact that the newer Sisense version is using a newer version of the Google AdWords API (v201506). If this is the case, then this API version includes changes to some of the AdWords predefined columns. Consequently, it is required for you to click refresh in the 'Change Connectivity Settings' window next to your AdWords data source and rebuild your ElastiCube. Unless this action is taken, your ElastiCube build will fail due to the change in Google AdWords schema.

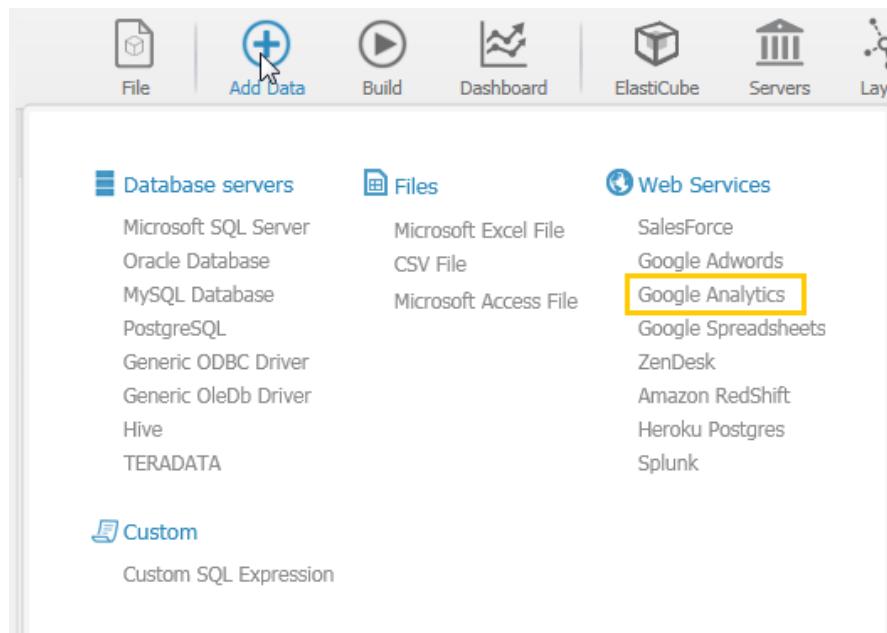
See also Customizing Google Analytics and AdWords Tables.

Connecting to Google Analytics

The ElastiCube Manager enables easy and quick access to tables contained within Google Analytics. The steps below describe how to connect to this type of data source.

Note: Google Analytics provides a quota on the free daily API requests. For this reason a developer key must be supplied. Read more details about [Quota Limits](#).

1. Click **Add Data** in the top menu of the ElastiCube Manager.
2. Under the **Web Services** category, select **Google Analytics**.

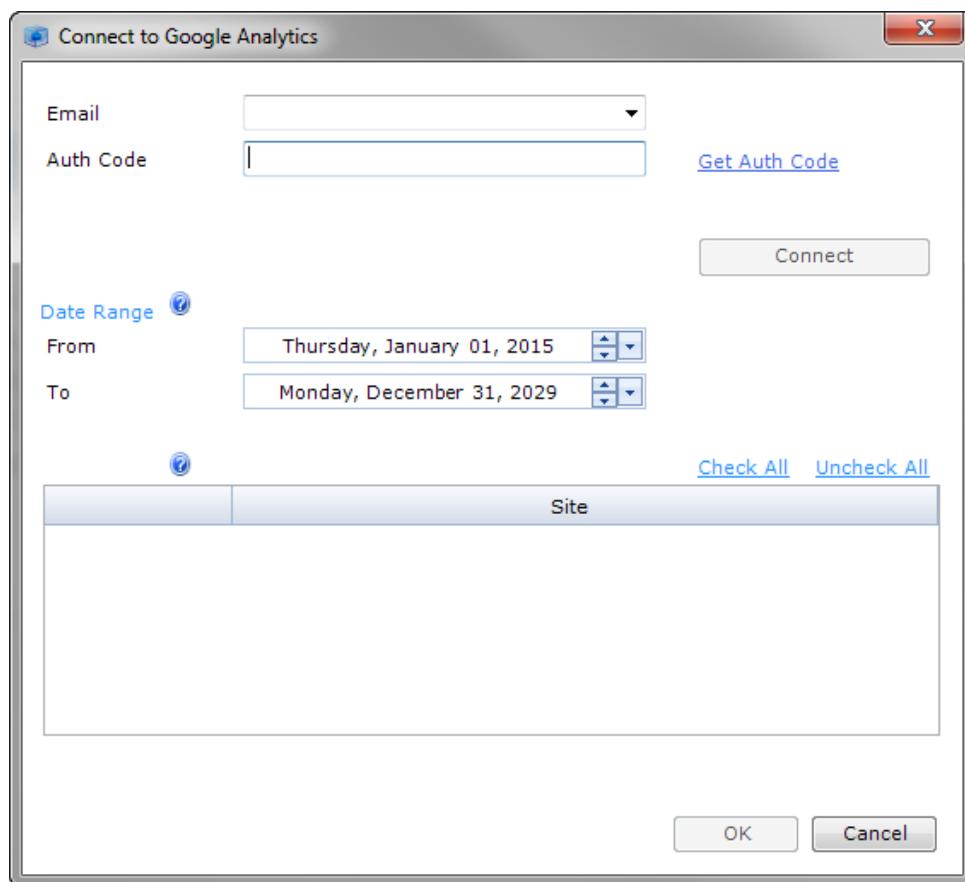


3. You will be prompted to enter the following information:
 - ▶ **Email:** Enter email associated with your Google Analytics account.
 - ▶ **Auth Code:** Google requires an authentication code to enable access from the ElastiCube to data associated with a Google Analytics account (site name and profile IDs). To attain the Auth code click on the link **Get Auth Code**. The browser will open

and you will need to sign in to your Google Analytics account.

After signing in, click on the 'Allow access' button. This will generate a temporary code that you must paste into the **Auth Code** text box in the connection window.

- ▶ **Date Range:** Select the date range of the analytics data you require.



4. Click **Connect** to display a list of available websites the list box below.
5. Select the relevant website that you want to work with, and click **OK**.
6. All tables and views associated with the website appear in a new window.
7. Select the checkbox next to each table you want to use.

Existing relationships between tables can be automatically replicated in the ElastiCube by selecting the **Automatically create relationships from database** checkbox. Likewise fields with similar names can be linked by selecting the **Automatically create relationships for fields with the same name** checkbox.

8. Once all relevant tables are selected, click **Add**.

See Customizing Google Analytics and AdWords Tables for in-depth info on controlling your Google Analytics tables.

Avoiding Sampling in Google Analytics

Google Analytics applies sampling whenever more than 500,000 sessions are returned within a given query. For example, if we are asking for a 1-month time period that had 1,000,000 session in it, Google will apply sampling and we will only get 50% of the actual sessions. Whereas in some cases sampling might only slightly affect the results accuracy, in many cases it is crucial.

Sisense supports 2 configuration options to reduce the chances of sampling:

1. **Split Intervals:** A Google Analytics table can be configured to be split into a number of given intervals. For example, if intervals=4, then a one month period will be split into 4 query requests equal in days.
2. **Day Intervals:** A Google Analytics table can be configured to be split into multiple intervals where each is the size of the given number of days. For example, if day_intervals=10, then a 30 day time period will be split into 3 intervals.

Since Google limitations are applied separately per request, this reduces the risk of overall sampling.

Configuring Google Analytics sampling intervals

Edit the following xml file (if it doesn't exist, create it):

```
%appdata%\Sisense\PrismServer\LocalRepository2.0\Resources\Analytics\GAnalytics.Tables.xml
```

The xml file needs to have a table node for each table, under a tables root:

```
<?xml version="1.0" encoding="utf-8" ?>

<tables>

    <table
        name="MyGoogleAnalyticsTable" days_interval="2" split_by="10"
    />

</tables>
```

For Split Intervals as described above use the "**split_by**" attribute. For Day Intervals use the "**days_interval**" attribute.

Note that only one of these attributes should be used. "split_by" will be used in case both are available.

Connecting to Google BigQuery

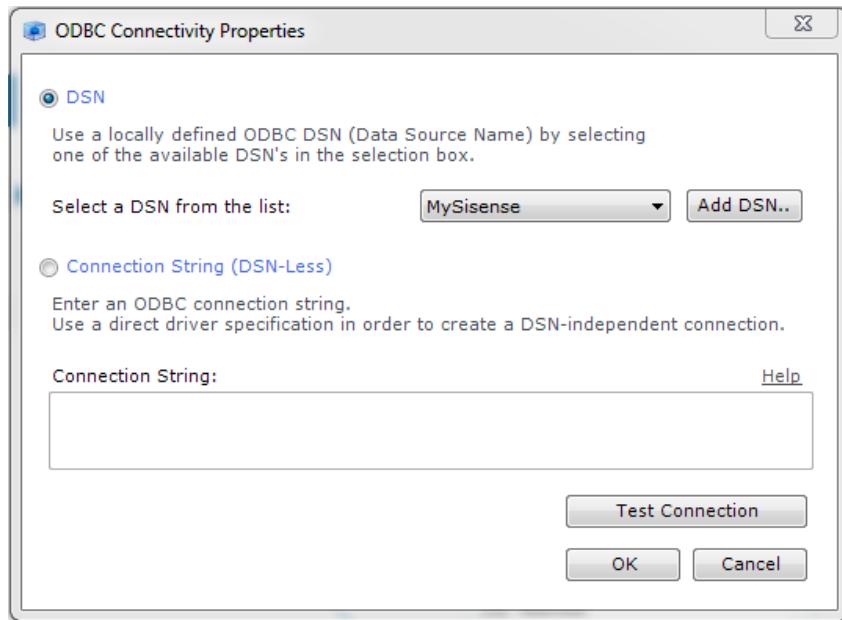
Google BigQuery Overview

The ElastiCube Manager provides you with quick and easy access to Google BigQuery tables enabling you to generate and analyze your data.

There are two connection types you can configure when connecting to your BigQuery tables, DSN and Connection String (DSN-Less). For more information about the differences between the connection types, click [here](#).

Downloading and Connecting the Sisense Google BigQuery ODBC Driver

1. [Download](#) and install the Sisense BigQuery ODBC Driver.
2. In ElastiCube Manager, click **Add Data** and then, **Generic ODBC Driver**.
3. Select **DSN**. If a DSN file has already been created, select it in the dropdown list, and click **Test Connection**. If you need to add a DSN, see Adding a DSN. If your connection has been set up, you can click **OK** and review the tables generated by the Sisense BigQuery ODBC driver.



OR

Select Connection String (DSN-Less) and enter your Connection String. Click **Test Connection**. For more information, see Connecting without a DSN.

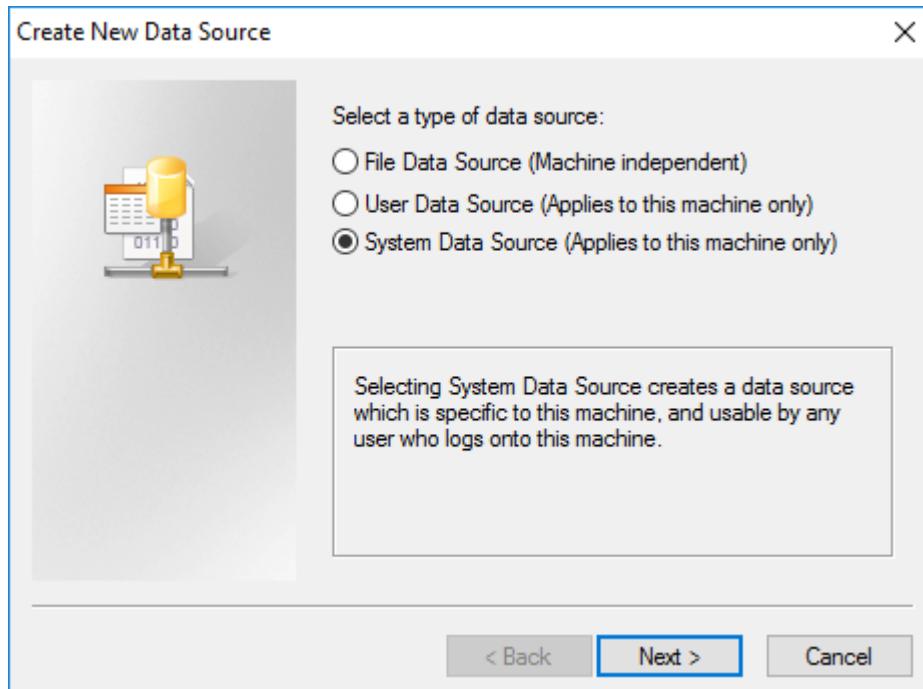
Adding a DSN

If a DSN has not yet been created, you must add the DSN manually.

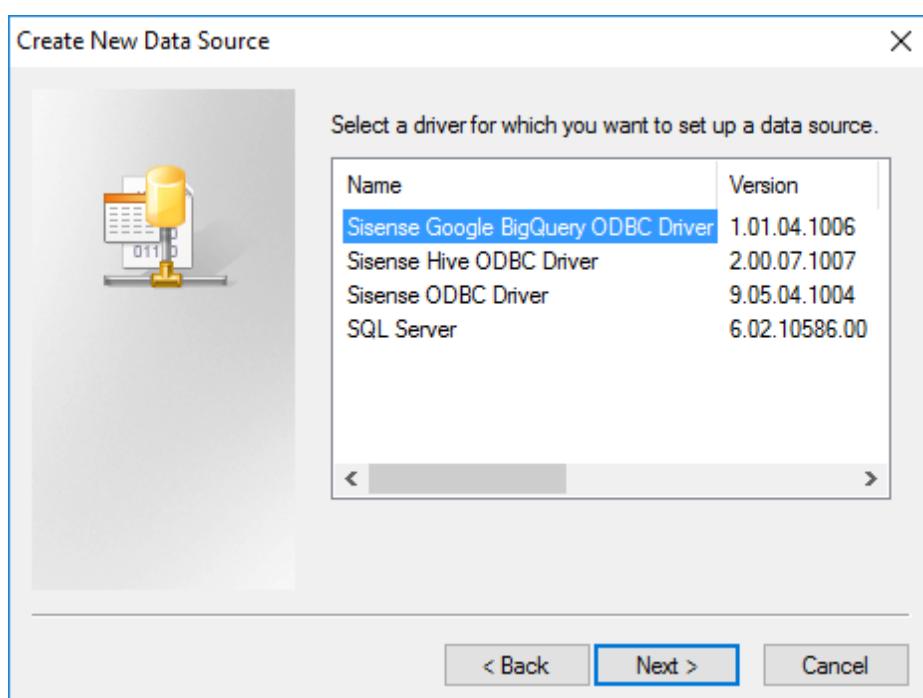
Note: To add a DSN you must run the Sisense ElastiCube Manager as an administrator.

To add a DSN:

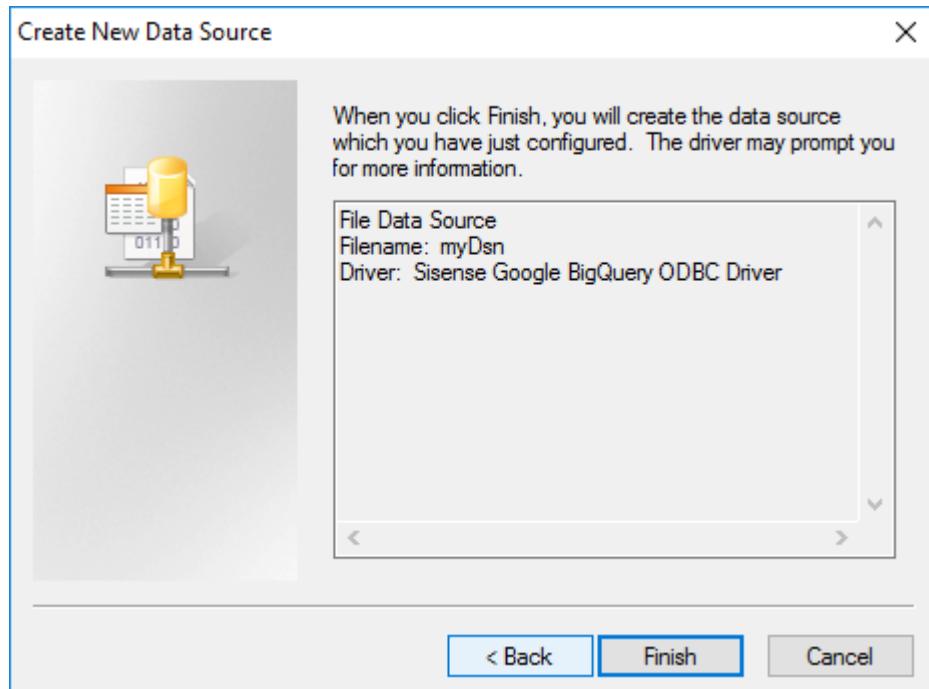
1. Click **Add DSN**.



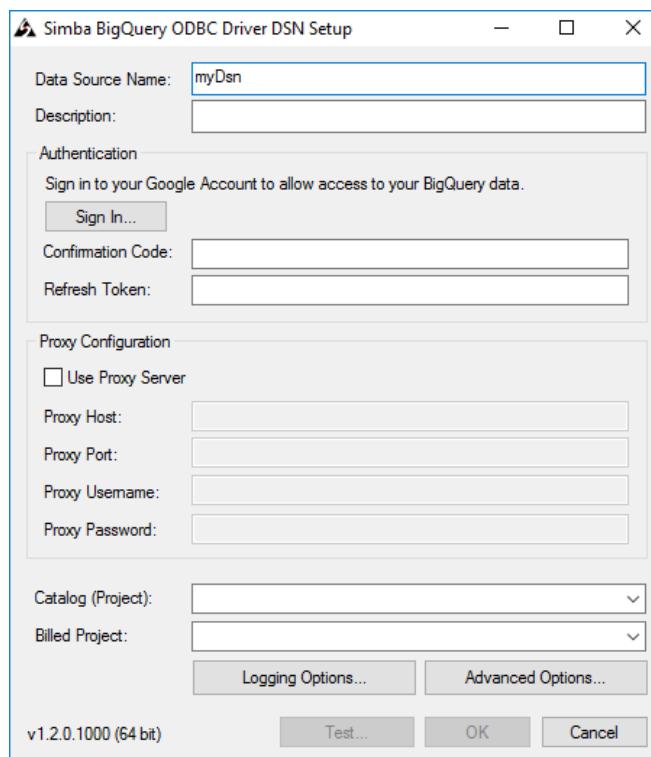
2. Select the **System Data Source** option. The created file applies to all users in a specific machine only. Click **Next**.
3. Select the Sisense Google BigQuery ODBC Driver, and click **Next**.



4. Click **Finish**.



5. In the Sisense BigQuery ODBC Driver DSN Setup window, do the following:



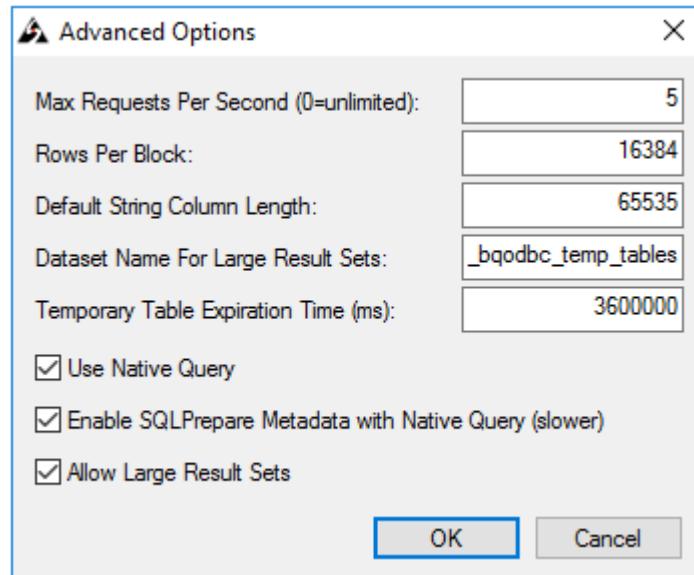
Field	Description
Data Source Name	Enter your name for the DSN that is to be displayed in the DSN dropdown list.
Description	Enter your description of the DSN.
Confirmation Code	Click Sign In . Your browser opens Google's BigQuery Authentication page. Enter your credentials and click Sign In . Google prompts you to allow BigQuery Client Tools to access your data in Google BigQuery, click Accept . Your Confirmation Code is displayed. Copy and paste this value into the Confirmation Code field.
Refresh Token	This field is automatically populated when you enter your Confirmation Code. The refresh token is used whenever the driver needs to access your BigQuery data.
Use Proxy Server	(Optional) Select to connect to a proxy server. After selecting the checkbox, enter the following information: Proxy Host: Enter the IP address or hostname of the proxy server. Proxy Port: Enter the port number that the proxy server is listening to. Proxy Username: Enter your proxy server user name. Proxy Password: Enter your proxy server password.
Catalog (Project)	Select the name of your BigQuery project.
Billed Project	Select the project to bill for queries that are run using the DSN.

6. (Optional) To configure advanced driver options, click **Advanced Options**. For more information, see Advanced Options.
7. (Optional) To configure logging behavior for the driver, click **Logging Options**. For more information, see Logging Options.

8. Click **Test**. A pop window is displayed that indicates the connection was successful.
9. Click **OK**.
10. In the ODBC Connectivity Properties window, click **Test Connection**.
11. Click **OK** to save and close the ODBC Data Source Administrator.

Advanced Options

You can configure advanced driver by clicking **Advanced Options** in the Sisense BigQuery ODBC Driver DSN Setup window. This displays the Advanced Options window from which you can configure the following options:



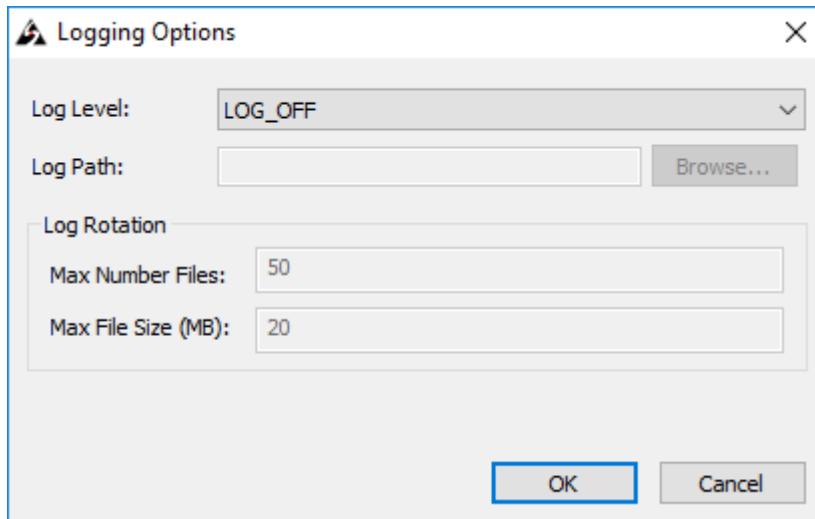
Field	Description
Max Requests Per Second (0=unlimited)	Enter the maximum number of requests that can be made per second.

Field	Description
	<p>Note: To allow an unlimited number of requests per second with no throttling, type 0.</p>
Rows Per Block	Enter the maximum number of rows to fetch for each data request.
Default String Column Length	Enter the maximum data length for String columns.
Dataset Name For Large Result Sets	<p>Enter the name of the BigQuery dataset to use to store temporary tables.</p> <p>Note: The dataset created from the default ID is hidden.</p>
Temporary Table Expiration Time(ms)	Enter the length of time (in milliseconds) that the temporary table exists for.
Use Native Query	<p>Select the Use Native Query checkbox to disable the SQL Connector feature and allow the driver to execute BigQuery SQL directly.</p> <p>When Use Native Query is selected, the driver does not transform the queries emitted by an application, so the native query is used.</p> <p>When Use Native Query is disabled, the driver transforms the queries emitted by an application and converts them into an equivalent form in BigQuery SQL.</p> <p>Note: If an application already emits BigQuery SQL, then enable this option to avoid the extra overhead of query transformation. (Optional)</p>
Enable SQLPrepare Metadata with Native Query (slower)	To enable the driver to retrieve metadata during the prepare stage of a query when working in Native Query mode, select the Enable SQLPrepare Metadata with Native Query checkbox.

Field	Description
	<p>Note: This option is available only if the Use Native Query checkbox is selected.</p>
Allow Large Result Sets	<p>Select to allow query results larger than 128MB in size.</p> <p>After you select this checkbox, the Dataset Name For Large Result Sets and Temporary Table Expiration Time(ms) fields are enabled.</p>

Logging Options

You can configure logging options by clicking **Logging Options** in the Sisense BigQuery ODBC Driver DSN Setup window. This displays the Logging Options window.



The ODBC Data Source Administrator provides tracing functionality, which you can activate to help troubleshoot issues.

Important: Only enable logging long enough to capture an issue. Logging decreases performance and can consume a large quantity of disk space.

The driver allows you to set the amount of detail included in log files.

The table below lists the logging levels provided by the Sisense ODBC Driver with SQL Connector for Google BigQuery, in order from least verbose to most verbose.

In the Logging Options window, you can configure the following information:

Field	Description
Log Level	<p>Select the Log Level. There are seven possible options:</p> <p>LOG_OFF: Disables all logging.</p> <p>LOG_FATAL: Logs very severe error events that lead the driver to abort.</p> <p>LOG_ERROR: Logs error events that might still allow the driver to continue running.</p> <p>LOG_WARNING: Logs potentially harmful situations.</p> <p>LOG_INFO: Logs general information that describes the progress of the driver.</p> <p>LOG_DEBUG: Logs detailed information that is useful for debugging the driver.</p> <p>LOG_TRACE: Logs more detailed information than the DEBUG level.</p>
Log Path	<p>Enter the full path to the folder where you want to save log files.</p> <p>OR</p> <p>Click Browse and select the folder where you want to save log files.</p>
Log Rotation	<p>Enter the maximum number of log files to keep in the Max Number Files field.</p> <p>Note: After the maximum number of log files is reached, each time an additional file is created, the driver deletes the oldest log file.</p> <p>Enter the maximum size of each log file in megabytes (MB) in the Max File Size field.</p>

Field	Description
	<p>Note: After the maximum file size is reached, the driver creates a new file and continues logging.</p>

Connecting without a DSN

When you choose to connect without a DSN, you must define the connection string that sets the configuration options.

The following is an example connection string that sets advanced options:

```
DSN=Sample Sisense BigQuery DSN; Catalog=public_data;
MaxRequestsPerSecond=20; UseNativeQuery=0;
AllowLargeResults=0;
LargeResultsDataSetId=_bqodbc_temp_tables;
LargeResultsTempTableExpirationTime=3600000
```

The table below provides a list of possible keys you can add to your string and their descriptions:

Key Name	Mandatory	Description
AllowLargeResults	No	<p>The value of this key is 0 or 1.</p> <p>0: Disabled. The driver returns an error when query results are larger than 128MB in size.</p> <p>1: Enabled. The driver allows query results that are larger than 128MB in size.</p>

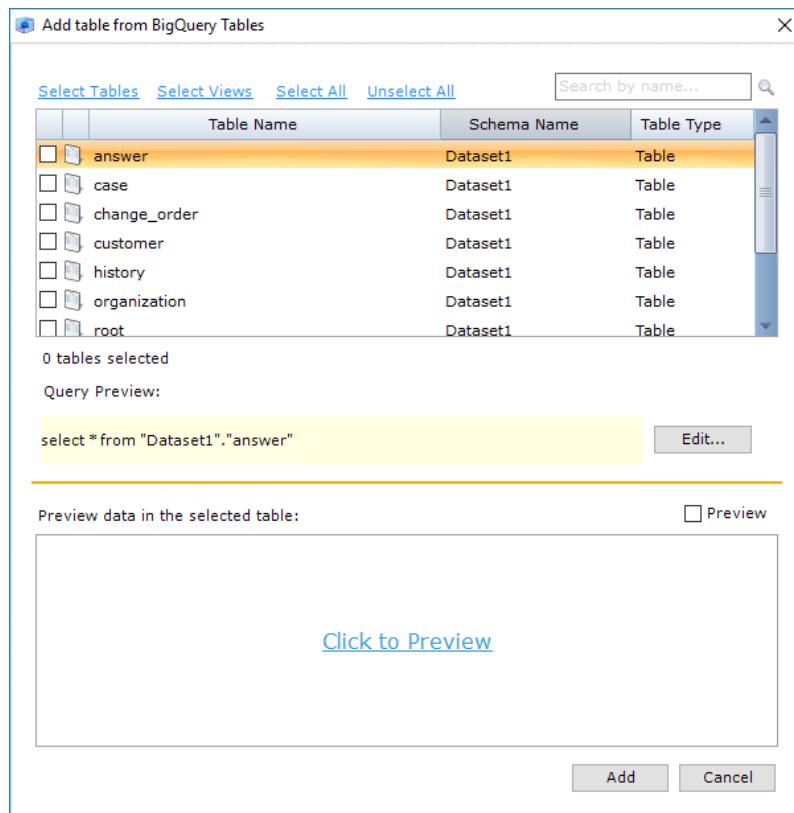
Key Name	Mandatory	Description
ExecCatalog	Yes	The name of the project to bill for queries that are run using the DSN.
Project or Catalog	Yes	The name of your BigQuery project.
LargeResultDataSetId	Yes*	The ID of the BigQuery dataset to use to store temporary tables.
		<p>Note: This option is available only when the Allow Large Result Sets option is enabled (the AllowLargeResults key is set to 1). The dataset created from the default ID is hidden.</p>
		*Required if AllowLargeResults is enabled.
DefaultStringColumnLength	No	The maximum data length for STRING columns.
UseNativeQueryMetadata	No	<p>The value of this key is 0 or 1.</p> <p>0: Disabled. The driver does not retrieve metadata during the prepare stage.</p> <p>1: Enabled. The driver retrieves metadata during the prepare stage of a query when working in Native Query mode.</p>
		<p>Note: This option is available only when UseNativeQuery key is set to 1.</p>
MaxRequestsPerSecond	No	The maximum number of requests that can be made per second. To allow an unlimited number of requests per second with no throttling, set the value to 0.

Key Name	Mandatory	Description
ProxyHost	No	The IP address or hostname of the proxy server. If this key is not set for the Linux or Mac OS X driver, then support for connecting to proxy servers is disabled.
ProxyPwd	Yes*	The password corresponding to the user name provided in the Proxy Username field (the ProxyUid key).
ProxyPort	Yes*	The number of the port on which the proxy server is listening. *Required if connecting to a proxy server.
ProxyUid	Yes*	The user name that you use to access the proxy server. *Required if connecting to a proxy server.
RefreshToken	Yes	The refresh token that you obtain from Google for authorizing access to BigQuery. When you configure a DSN with the Windows driver, the refresh token is generated automatically after you provide the confirmation code. When you configure a DSN with the Linux or Mac OS X versions of the driver, you must use the Refresh Token configuration tool to generate the token.
RowsFetchedPerBlock	No	The maximum number of rows that the driver can fetch for each data request.

Key Name	Mandatory	Description
LargeResultsTempTable ExpirationTime	Yes*	<p>The length of time (in milliseconds) for which a temporary table exists.</p> <p>Note: This option is available only when AllowLargeResults key is set to 1. The default value is 1 hour in milliseconds.</p>
UseNativeQuery	No	<p>The value of this key is 0 or 1.</p> <p>0:Disabled. The driver transforms the queries emitted by an application and converts them into an equivalent form in BigQuery SQL.</p> <p>1:Enabled. The driver does not transform the queries emitted by an application, so the native query is used.</p> <p>Note: If an application already emits BigQuery SQL, then enable this option to avoid the extra overhead of query transformation.</p>

Adding BigQuery Tables to your Project

After setting up the DSN or the DSN-less connection, the Add table from BigQuery Tables window is displayed.



From this window, you add your BigQuery tables to your ElastiCube Manager. In addition, you can view the SQL syntax in the Query Preview section and click **Edit** to customize it.

To add BigQuery Tables to your Project:

1. Connect to Google BigQuery via the Sisense ODBC Tool.
2. Add your data source.
3. In the Add table from BigQuery Tables window, select the table you want to add to the ElastiCube Manager.
4. Click **Add**. The selected tables are added to your ElastiCube Manager.

Connecting to Google Spreadsheets

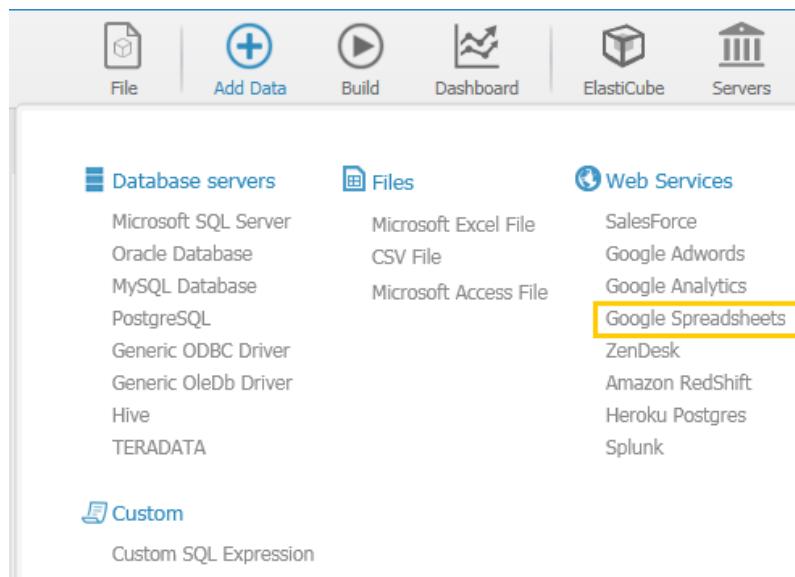
The ElastiCube Manager enables easy and quick access to tables contained within Google Spreadsheets.

Note: To work with Google Spreadsheets, you must have a professional Google account, since Google's API is only available to business accounts.

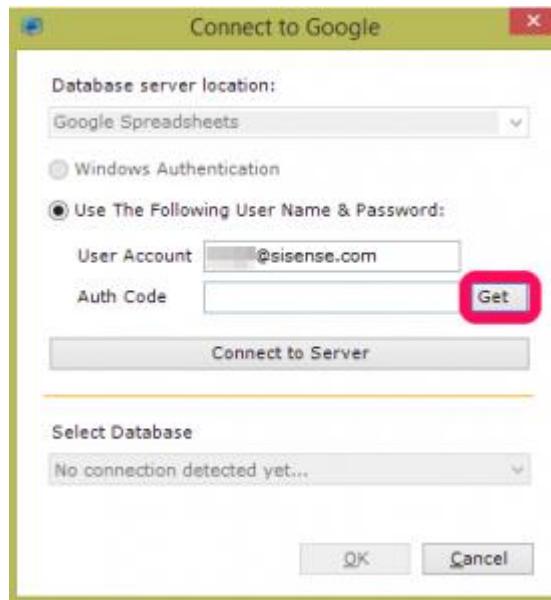
To connect to Google Spreadsheets:

1. Click the **Add Data** icon in the top menu of the ElastiCube Manager.
2. Under the **Web Services** category, select **Google Spreadsheets**.

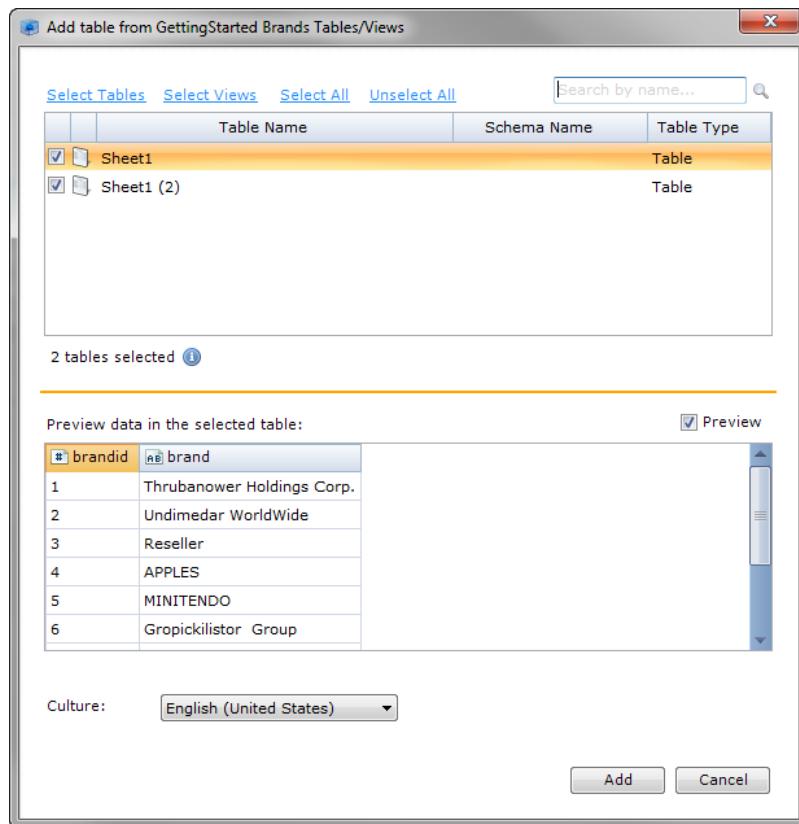
Note: Make sure you are working with Sisense version 5.7.6.71 or above. If you are working with a previous version and cannot connect to Google Spreadsheets, [click here](#) to upgrade your Sisense version and learn how to refresh your data.



The following screen appears.



3. In the **User Account** field, enter your user account email.
4. Click **Get** to obtain your Google 2-Step Verification code, and enter it in the **Auth Code** field.
5. Click **Connect to Server**.
6. From the **Select Database** drop-down list, select the relevant spreadsheet you want to work with and click **OK**.
7. Select the checkbox next to each sheet you want to use.



8. Once all relevant tables are selected, click **Add**.

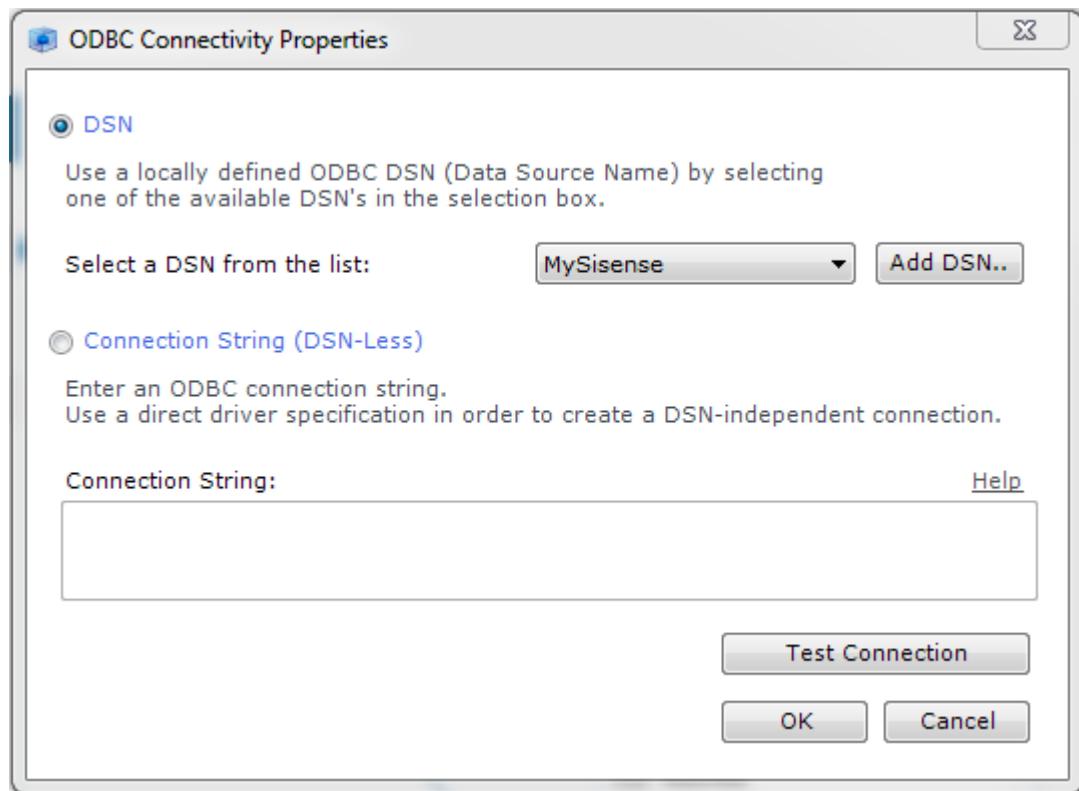
Connecting to Hadoop via Hive

Hive Overview

The Sisense ODBC Driver with SQL Connector for Apache Hive enables standard SQL-92 access directly to Apache Hive distributions. It efficiently maps SQL to HiveQL delivering full SQL application functionality and real-time analytic and reporting capabilities to viewers.

Downloading and Connecting the Sisense Hive ODBC Driver

1. [Download](#) and install the Sisense Hive ODBC Driver.
2. In ElastiCube Manager, click **Add Data** and then, **Generic ODBC Driver**.
3. Select **DSN**. If a DSN file has already been created, select it in the dropdown list, and click **Test Connection**. If you need to add a DSN, see Adding a DSN. If your connection has been set up, you can click **OK** and review the tables generated by the Sisense Hive ODBC driver.



OR

Select **Connection String (DSN-Less)** and enter your Connection String. Click **Test Connection**. For more information, see Connecting without a DSN.

Adding a DSN

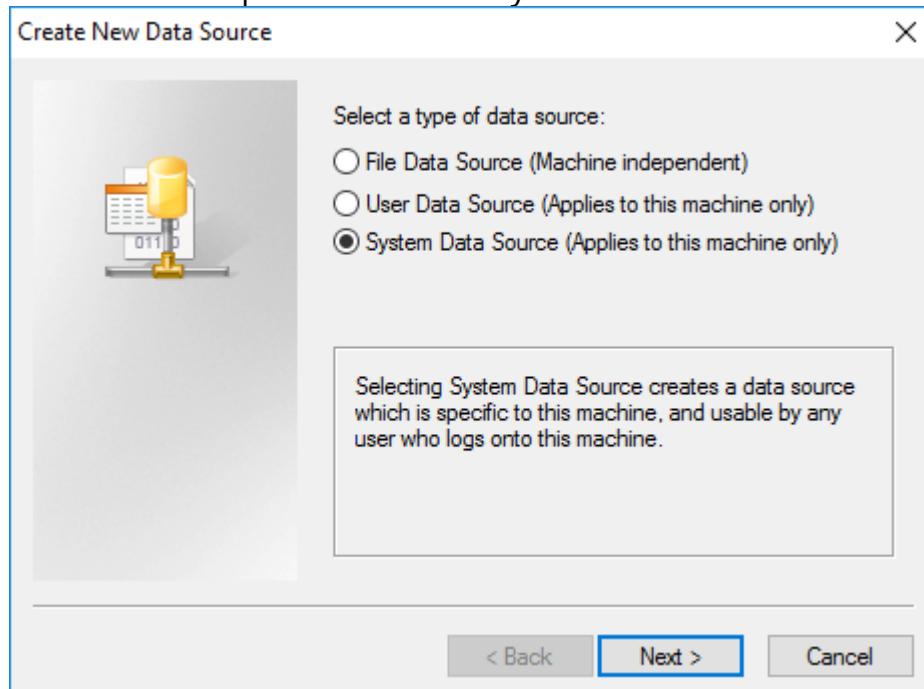
If a DSN has not yet been created, you must add the DSN manually.

Note: To add a DSN you must run the Sisense ElastiCube Manager as an administrator.

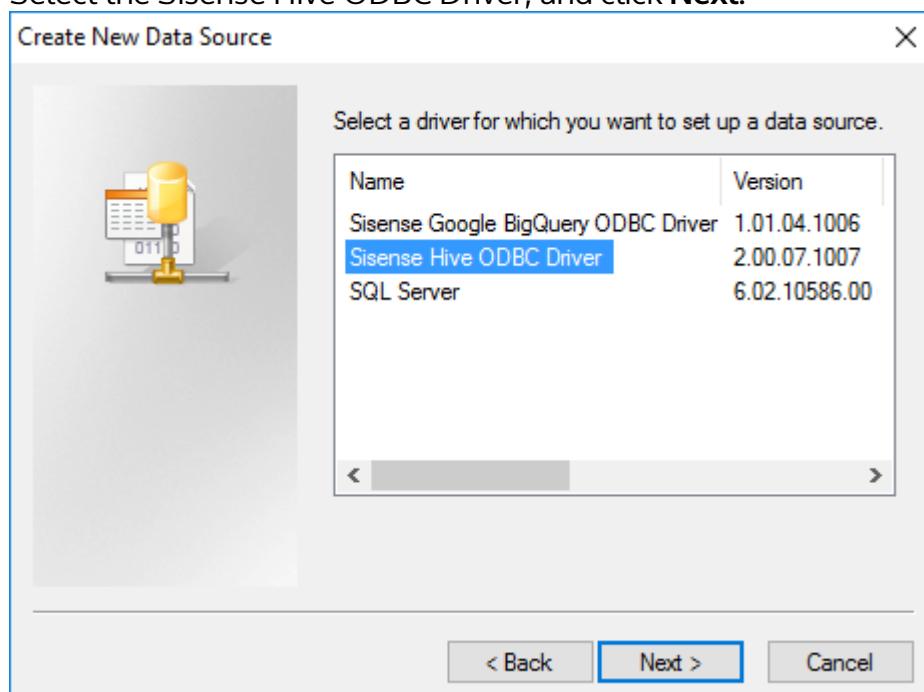
To add a DSN:

1. Click **Add DSN**.

2. Select the **System Data Source** option. The created file applies to all users in a specific machine only. Click **Next**.

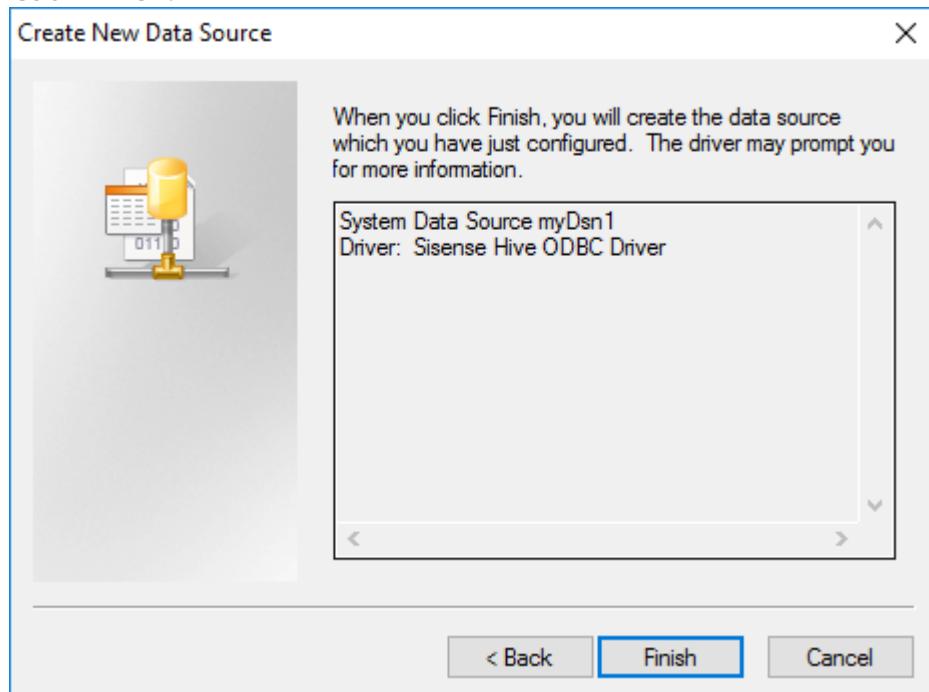


3. Select the Sisense Hive ODBC Driver, and click **Next**.

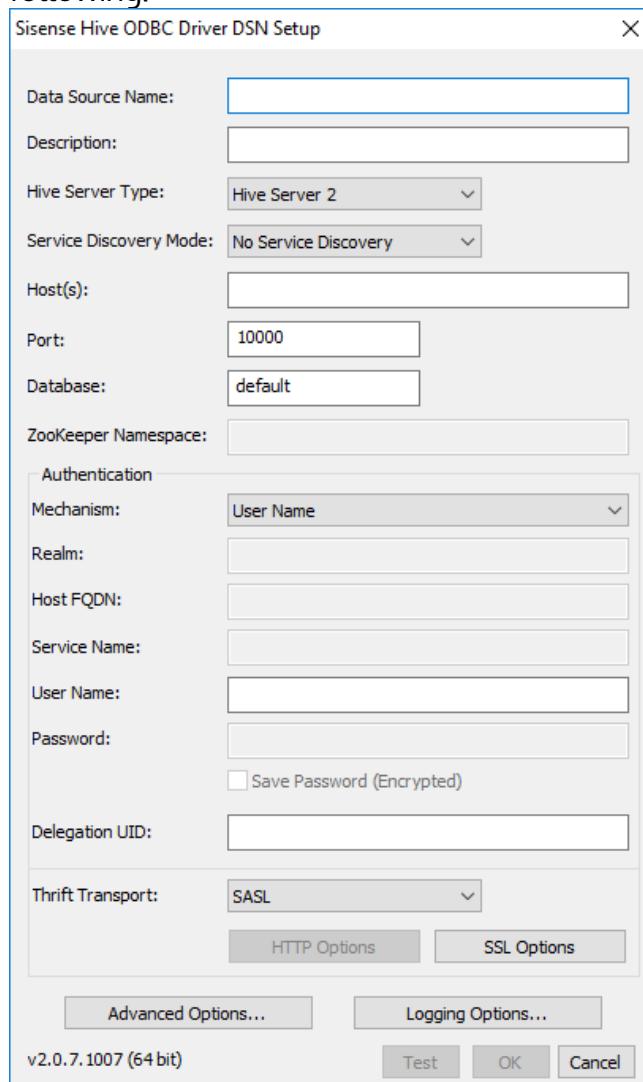




4. Click **Finish**.



5. In the Sisense Hive ODBC Driver DSN Setup window, do the following:



Field	Procedure
Data Source Name	Enter your name for the DSN that is to be displayed in the DSN dropdown list.
Description	Enter your description of the DSN.
Hive Server Type	Select the server type used by your data source. Note: If you are connecting through Apache ZooKeeper, then Hive Server 1 is not supported.

Field	Procedure
Service Discovery Mode	<p>Select the Service Discovery Mode. There are two possible options you can select:</p>
	<p>No Service Discovery: The driver connects to Hive without using the ZooKeeper service.</p>
	<p>ZooKeeper: The driver discovers Hive Server 2 services via the ZooKeeper service.</p>
Host(s)	<p>If you selected No Service Discovery above as the Service Discovery Mode, then enter the IP address or hostname of the Hive server.</p> <p>OR</p> <p>If you selected ZooKeeper as the Service Discovery Mode, enter a comma-separated list of ZooKeeper servers.</p> <p>Use the following format, where zk_host is the IP address or hostname of the ZooKeeper server and zk_port is the number of the port that the ZooKeeper server uses: zk_host1:zk_port1,zk_host2:zk_port2 .</p>
Port	<p>If you selected No Service Discovery above as the Service Discovery Mode, then enter the number of the TCP port on which the Hive server is listening. Otherwise, do not type a value in the field.</p>
Database	<p>Enter the name of the database schema to use when a schema is not explicitly specified in a query. You can still issue queries on other schemas by explicitly specifying the schema in the query. To inspect your databases and determine the appropriate schema to use, type the show databases command at the Hive command prompt.</p>

Field	Procedure
ZooKeeper Namespace	<p>If you selected ZooKeeper as the Service Discovery Mode, enter the namespace on ZooKeeper under which Hive Server 2 znodes are added.</p> <p>Otherwise, do not type a value in the field.</p>
Authentication	<p>Some Hive servers are configured to require authentication for access. To connect to a Hive server, you must configure the Sisense Hive ODBC Driver to use the authentication mechanism that matches the access requirements of the server and provides the necessary credentials</p> <p>Hive Server 1 does not support authentication.</p> <p>Most default configurations of Hive Server 2 require User Name authentication. To verify the authentication mechanism that you need to use for your connection, check the configuration of your Hadoop/Hive distribution.</p>
Mechanism	<p>If your Hive server is configured to use authentication, select the appropriate authentication mechanism and provide the necessary credentials and authentication settings in the fields below the list.</p>
Realm	<p>This field is available when Kerberos is selected as your authentication mechanism.</p> <p>Enter the realm of the Hive Server 2 host.</p> <p>If your Kerberos configuration already defines the realm of the Hive Server 2 host as the default realm, then you do not need to configure this option.</p>
Host FQDN	<p>This field is available when Kerberos is selected as your authentication mechanism.</p> <p>Enter the HS2HostFQDN key to the fully qualified domain name of the Hive Server 2 host.</p>

Field	Procedure
Service Name	<p>This field is available when Kerberos is selected as your authentication mechanism.</p> <p>Enter the HS2KrbServiceName key to the service name of the Hive Server 2 host.</p>
User Name	<p>This field is available when User Name, User Name and Password, or Windows Azure is selected as your authentication mechanism.</p> <p>The user name that you use to access Hive Server 2.</p>
Password	<p>This field is available when User Name and Password or Windows Azure is selected as your authentication mechanism.</p> <p>The password corresponding to the user name that you provided in the User Name field (the UID key).</p>
Delegation UID	<p>Enter the name of the user to be delegated in the Delegation UID field.</p> <p>The Delegation UID represents a user that is different than the authenticated user for the connection.</p> <p>This option is applicable only when connecting to a Hive Server 2 instance that supports this feature.</p>
Thrift Transport	<p>Select the transport protocol to use in the Thrift layer. For Hive Server 1, select Binary. When using an authentication mechanism, the Binary transport protocol is not supported.</p> <p>For Hive Server 2, select SASL.</p> <p>If you select HTTP, the HTTP Options button is enabled. See Configuring HTTP Options for more information.</p>

Field	Procedure
	To configure client-server verification over SSL, click SSL Options . See SSL Options for more information.

6. (Optional) To configure advanced driver options, click **Advanced Options**. For more information, see Advanced Options.
7. (Optional) To configure logging behavior for the driver, click **Logging Options**. For more information, see Logging Options.
8. Click **Test**. A pop window is displayed that indicates the connection was successful.
9. Click **OK**.
10. In the ODBC Connectivity Properties window, click **Test Connection**.
11. Click **OK** to save and close the ODBC Data Source Administrator.

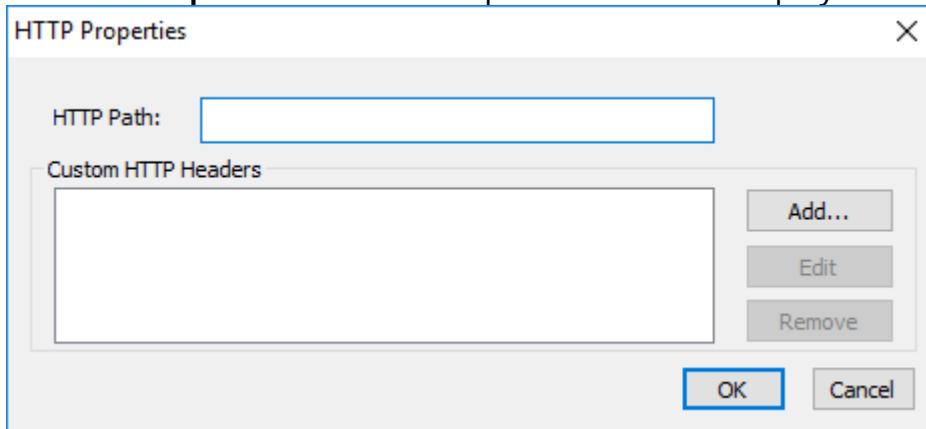
Configuring HTTP Options

You can configure options such as custom headers when using the HTTP transport protocol in the Thrift layer.

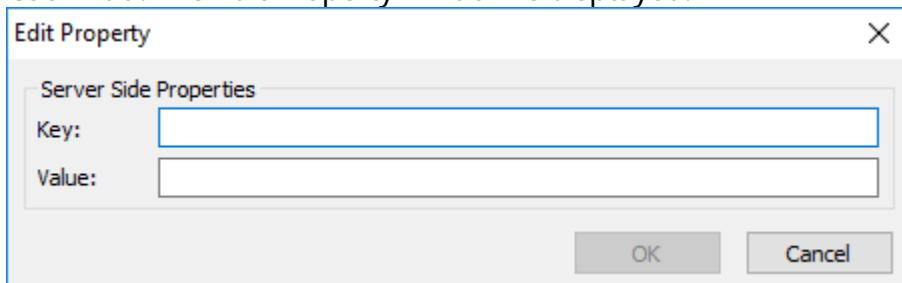
To configure HTTP options:

1. In the Sisense Hive ODBC Driver DSN Setup window, select **HTTP** as the HTTP transport protocol from the Thrift Transport list.

2. Click **HTTP Options**. The HTTP Properties window is displayed.



3. In the **HTTP Path** field, enter the partial URL corresponding to the Hive server.
4. Click **Add**. The Edit Property window is displayed.



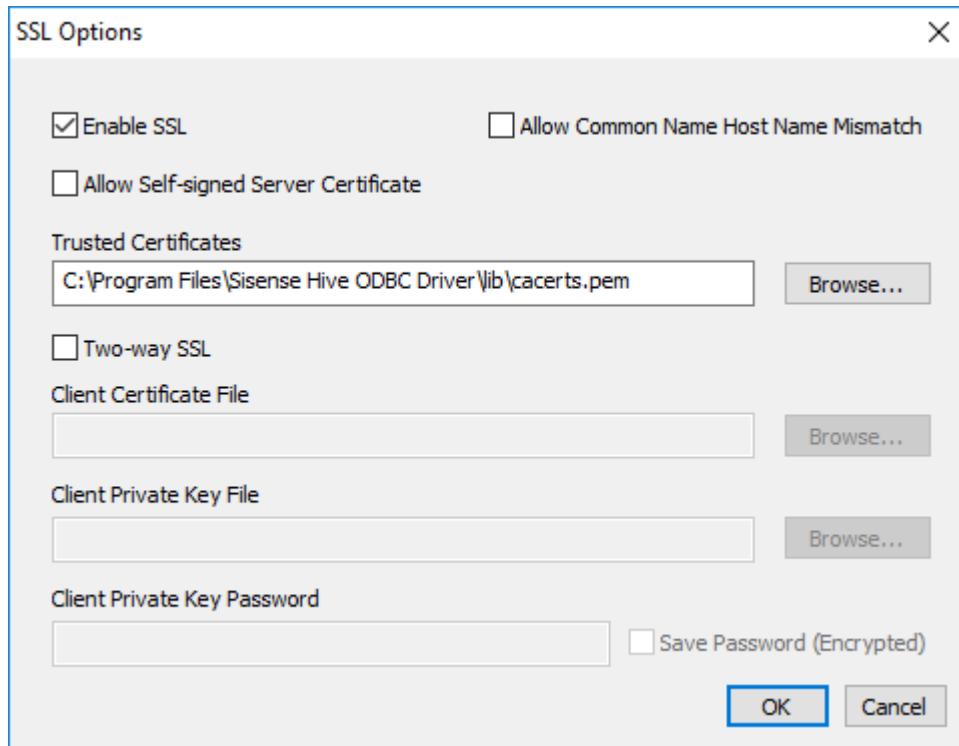
5. In the **Key** and **Value** fields, enter the appropriate keys and values and click **OK**.
6. After you add an HTTP header, you can edit or delete the header by clicking **Edit** or **Delete**.
7. Click **OK**.

SSL Options

You can configure verification between the client and the Hive server over SSL.

To configure SSL verification:

1. In the Sisense Hive ODBC Driver DSN Setup window, click **SSL Options**. The SSL Options window is displayed.



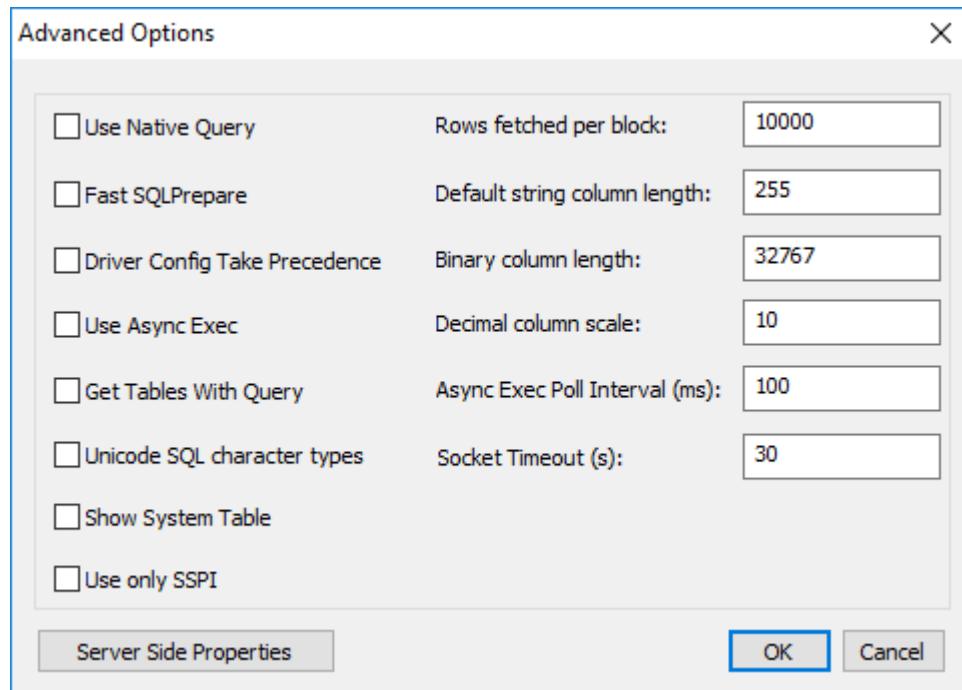
2. Select **Enable SSL**.
3. Select the **Allow Self-signed Server Certificate** checkbox to allow self-signed certificates from the server.
4. Select the **Allow Common Name Host Name Mismatch** checkbox to allow the common name of a CA-issued SSL certificate to not match the host name of the Hive server.
5. In the **Trusted Certificates** field, enter the path or click **Browse** and navigate to the location of your PEM file if you want the driver to load your SSL certificates when verifying the server.
OR
To use the trusted CA certificates PEM file that is installed with the driver, leave the Trusted Certificates field empty.
6. Select the **Two-Way SSL** checkbox if you want to configure two-way SSL verification. The Client authentication fields are enabled.
7. In the **Client Certificate File** field, enter the path or click **Browse** and navigate to the location of your PEM containing the client's certificate.

8. In the **Client Private Key File** field, enter the path or click **Browse** and navigate to the location of your file containing the client's private key.
9. In the **Client Private Key Password** field, if the private key file is protected with a password, enter the password. Select the **Save Password (Encrypted)** checkbox to save the password.
10. Click **OK**.

Advanced Options

You can configure advanced driver by clicking **Advanced Options** in the Sisense Hive ODBC Driver DSN Setup window.

This displays the Advanced Options window from which you can configure the following options:



Field	Description
Use Native Query	Select to disable the SQL Connector feature. Select Use Native Query to disable the SQL Connector feature and allow the driver to execute HiveQL directly.

Field	Description
	<p>When Use Native Query is selected, the driver does not transform the queries emitted by an application, so the native query is used.</p> <p>When Use Native Query is disabled, the driver transforms the queries emitted by an application and converts them into an equivalent form for Hive.</p>
Fast SQLPrepare	Select to defer query execution to SQLExecute.
Driver Config Take Precedence	Select to allow driver-wide configurations to take precedence over connection and DSN settings.
Use Async Exec	<p>Select to use the asynchronous version of the API call against Hive for executing a query.</p> <p>This option is applicable only when connecting to a Hive cluster running Hive 0.12.0 or later.</p>
Get Tables With Query	<p>Select to retrieve the names of tables in a database by using the SHOW TABLES query.</p> <p>This option is applicable only when connecting to Hive Server 2.</p>
Unicode SQL character types	Select to enable the driver to return SQL_WVARCHAR instead of SQL_VARCHAR for STRING and VARCHAR columns, and SQL_WCHAR instead of SQL_CHAR for CHAR columns.
Show System Table	Select to enable the driver to return the hive_system table for catalog function calls such as SQLTables and SQLColumns.
Use only SSPI	Select to handle Kerberos authentication using the SSPI plugin instead of Kerberos by default.

Field	Description
Rows fetched per block	Enter the number of rows to be fetched per block.
Default string column length	Enter the maximum data length for STRING columns.
Binary column length	Enter the maximum data length for BINARY columns.
Decimal column	Enter the maximum number of digits to the right of the decimal point for numeric data types.
Async Exec Poll Interval (ms)	Enter the time in milliseconds between each poll for the query execution status. This option is applicable only to HDInsight clusters.
Socket Timeout (s)	Define the amount of time until the connection times out. The default amount of time is 30 seconds.

1. Click **Server Side Properties** to configure the driver to apply configuration properties to the Hive server. For more information, see Configuring Server Side Properties.
2. Click **OK**.

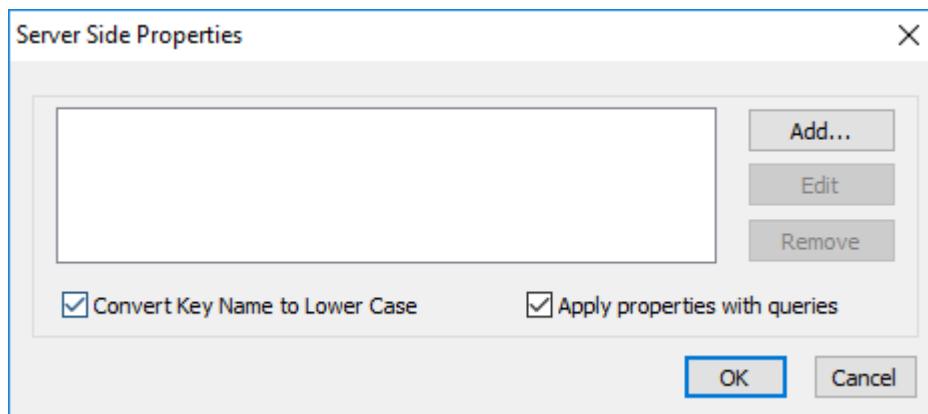
Configuring Server-Side Properties

You can use the driver to apply configuration properties to the Hive server.

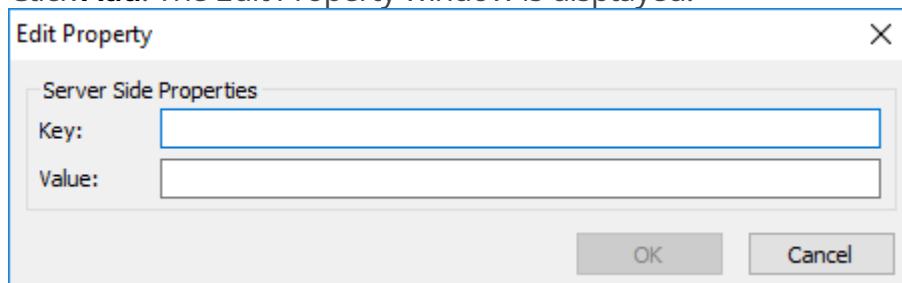
The Sisense ODBC Driver with SQL Connector for Apache Hive allows you to set server-side properties via a DSN. Server-side properties specified in a DSN affect only the connection that is established using the DSN.

To create a server-side property:

1. In the Sisense Hive ODBC Driver DSN Setup window, click **Advanced Options > Side Server Properties**. The Server Side Properties window is displayed.



2. Click **Add**. The Edit Property window is displayed.



3. In the **Key** and **Value** fields, enter the appropriate keys and values and click **OK**.

For a list of all Hadoop and Hive server-side properties that your implementation supports, type set -v at the Hive CLI command line or Beeline. You can also execute the set -v query after connecting using the driver. After you add your properties, you can edit or delete the properties by clicking **Edit** or **Delete**.

4. Click **OK**.
5. Select the **Apply properties with queries** checkbox to configure the driver to apply each server-side property by executing a query when opening a session to the Hive server.

OR

Clear the Apply Server Side Properties with Queries checkbox to configure the driver to use a more efficient method for applying server-side properties that does not involve additional network round-tripping.

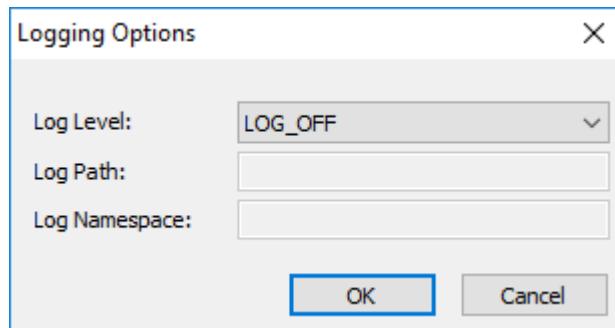
The more efficient method is not available for Hive Server 1, and

it might not be compatible with some Hive Server 2 builds. If the server-side properties do not take effect when the checkbox is clear, then select the checkbox.

6. Select the **Convert Key Name to Lower Case** checkbox to force the driver to convert server-side property key names to all lower case characters.
7. Click **OK**.

Logging Options

You can configure logging options by clicking **Logging Options** in the Sisense Hive ODBC Driver DSN Setup window. This displays the Logging Options window.



The ODBC Data Source Administrator provides tracing functionality, which you can activate to help troubleshoot issues.

Important: Only enable logging long enough to capture an issue. Logging decreases performance and can consume a large quantity of disk space.

The driver allows you to set the amount of detail included in log files.

The table below lists the logging levels provided by the Sisense ODBC Driver with SQL Connector for Hive, in order from least verbose to most verbose.

In the Logging Options window, you can configure the following information:

Field	Description
Log Level	<p>Select the Log Level. There are seven possible options:</p> <p>LOG_OFF: Disables all logging.</p> <p>LOG_FATAL: Logs very severe error events that lead the driver to abort.</p> <p>LOG_ERROR: Logs error events that might still allow the driver to continue running.</p> <p>LOG_WARNING: Logs potentially harmful situations.</p> <p>LOG_INFO: Logs general information that describes the progress of the driver.</p> <p>LOG_DEBUG: Logs detailed information that is useful for debugging the driver.</p> <p>LOG_TRACE: Logs more detailed information than the DEBUG level.</p>
Log Path	<p>Enter the full path to the folder where you want to save log files.</p> <p>OR</p> <p>Click Browse and select the folder where you want to save log files.</p>
Log Namespace	Enter a name for the log.

Connecting without a DSN

Some client applications provide support for connecting to a data source using a driver without a DSN. To configure a DSN-less connection, you can use a connection string.

The following is an example of a connection string for a DSN-less connection:

```
Driver=DriverNameOrFile;HOST=MyHiveServer;PORT=PortNumber;
Schema=DefaultSchema;HiveServerType=ServerType
```

The table below provides a list of possible keys you can add to your string and their descriptions:

Key Name	Description
DriverNameOrFile	The absolute path of the shared object file for the driver.
MyHiveServer	The IP address or host name of the Hive server.
PortNumber	The number of the port that the Hive server uses.
DefaultSchema	The database schema to use when a schema is not explicitly specified in a query.
ServerType	The server type, either 1(for Hive Server 1) or 2 (for Hive Server 2).

Adding Hive Tables to your Project

After setting up the DSN or the DSN-less connection, the Add table from Hive Tables window is displayed.

From this window, you add your Hive tables to your ElastiCube Manager. In addition, you can view the SQL syntax in the Query Preview section and click **Edit** to customize it.

To add Hive Tables to your Project:

1. Connect to Hive via the Sisense ODBC Tool.
2. Add your data source.
3. In the Add table from Hive Tables window, select the table you want to add to the ElastiCube Manager.
4. Click **Add**. The selected tables are added to your ElastiCube Manager.

Connecting to HubSpot

The Sisense HubSpot connector is a standalone connector that allows you to import data from HubSpot's API into the ElastiCube Manager.

After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the HubSpot APIs. To obtain a connection string, you will need to create a HubSpot app.

Once you have connected to HubSpot, you can import a variety of tables from the HubSpot API.

This section describes how to install the HubSpot connector, how to connect to HubSpot with a connection string, and what tables you can import into the ElastiCube Manager.

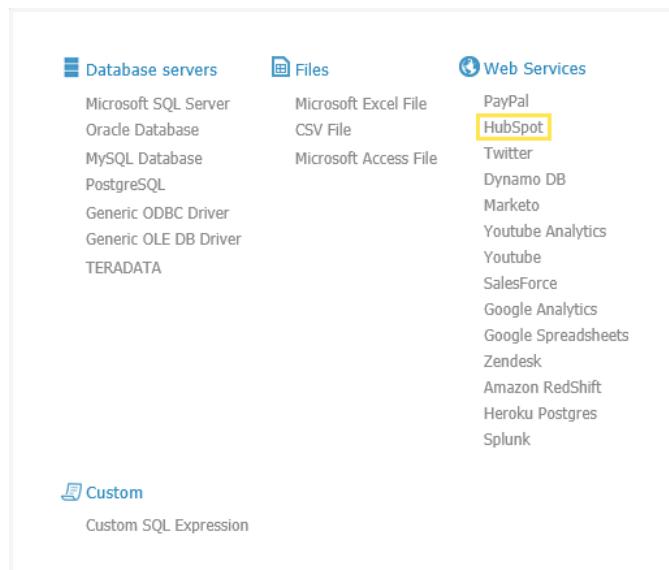
Installing the HubSpot Connector

Sisense provides the HubSpot connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the HubSpot connector:

1. [Download](#) the HubSpot installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.

The HubSpot connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.



Connecting to the HubSpot REST API

To access HubSpot's REST API from the ElastiCube Manager, you must provide valid Oauth HubSpot credentials through a connection string. These credentials are provided by HubSpot when you create an application.

After you receive your credentials from HubSpot, you can create the connection string and provide Sisense with it to connect to your data.

Creating an App

Follow the steps below to obtain the OAuth client credentials, the OAuthClientId and OAuthClientSecret:

1. Log into your HubSpot developer account.
2. Click **Create App**.
3. If you will only use the app to connect to your portal, select One Off. If other users will use the app to connect to their own portals, select For Everyone.
4. Click the name of your app.

5. Enter values to be displayed to users when you connect. These values include the app name, author name, and a description of the app.

Authenticating through HubSpot

After setting the following connection properties, you are ready to connect:

- ▶ **OAuthClientId**: Set this to the client Id in your app settings.
- ▶ **OAuthClientSecret**: Set this to the client secret in your app settings.
- ▶ **CallbackURL**: Set this to `http://localhost`.
- ▶ **HubId**: Set this to the Hub Id of the HubSpot account you are connecting to.
- ▶ **InitiateOAuth**: Set this to `GETANDREFRESH`. You can use `InitiateOAuth` to avoid repeating the OAuth exchange and manually setting the `OAuthAccessToken` connection property.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver then completes the OAuth process:

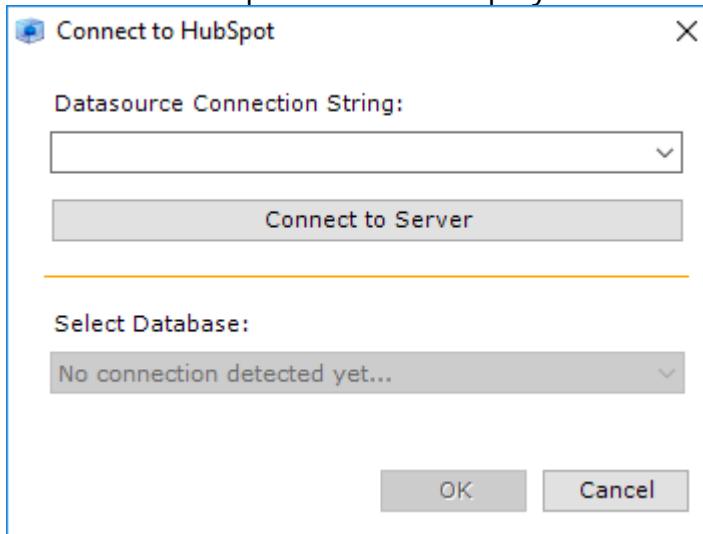
1. Extracts the access token from the callback URL and authenticates requests.
2. Refreshes the access token when it expires.
3. Saves OAuth values in `OAuthSettingsLocation` to be persisted across connections.

Adding HubSpot Tables to your ElastiCube

Sisense uses connection strings to connect to HubSpot and import data into the ElastiCube Manager. Each connection string contains a authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add HubSpot data:

1. In ElastiCube Manager, click **Add Data** and then, **HubSpot**. The Connect to HubSpot window is displayed.



2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. HubSpot is displayed in the Select Database list.
4. Click **OK**. Sisense connects to HubSpot and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.
The tables are displayed in the ElastiCube Manager.

HubSpot Tables

HubSpot's RESTful APIs expose the following HubSpot tables that you can import into the ElastiCube Manager through the Sisense HubSpot connector:

Name	Description
BlogAuthors	Retrieve the available blog authors in HubSpot.
BlogPosts	Retrieve the available blog posts in HubSpot.
Blogs	Retrieve the available blogs in HubSpot.

Name	Description
BlogTopics	Retrieve the available blog topics in HubSpot.
Comments	Retrieve the available comments from your blog in HubSpot.
CompanyProperties	Company properties are the standard and custom pieces of field data that appear in HubSpot.
CompanyPropertyGroups	Company property groups in HubSpot offer a way of organizing individual types of properties for companies. Each company property must belong to a property group.
ContactLists	Contact lists in HubSpot can be used to group together contacts with similar characteristics.
ContactProperties	Contact properties are the standard and custom pieces of field data that appear in HubSpot.
ContactPropertyGroups	Contact property groups in HubSpot offer a way of organizing individual types of properties for contacts. Each contact property must belong to a property group.
DealProperties	Deal properties are the standard and custom pieces of field data that appear in HubSpot.
DealPropertyGroups	Contact property groups in HubSpot offer a way of organizing individual types of properties for contacts. Each contact property must belong to a property group.
Domains	Retrieve the available domains in HubSpot.
EmailCampaigns	Email campaigns in HubSpot allow you to keep track of and update email marketing campaigns. This table allows you to create, update, and delete your email campaigns in HubSpot.

Name	Description
EmailCampaignEvents	The events associated with an email campaign or a recipient.
EmailSubscriptions	The subscription types a given email is subscribed to. An email must be specified to return results. Subscriptions may be removed by deleting them.
EmailSubscriptionTypes	A list of email subscription types for a HubSpot hub.
Files	Retrieves information about the available files in HubSpot.
Folders	Retrieves information about the available folders in HubSpot.
Forms	HubSpot Forms.
Owners	Retrieve the owners in HubSpot.
Pages	Retrieve the available pages in HubSpot.
SocialMediaChannels	List available social media channels in HubSpot.
Settings	HubSpot settings.
SocialMediaMessages	Social media messages.
Templates	Retrieve the available templates in HubSpot.
UrlMappings	Retrieve the available url mappings in HubSpot.
Workflows	HubSpot workflows.

Accumulative Builds

Sisense support accumulative builds for all numeric and dates data types. However, the data must be sorted before building the ElastiCube.

Connecting to MailChimp

This connector is currently in beta and subject to change.

The Sisense MailChimp connector is a standalone connector that allows you to import data from MailChimp's API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to MailChimp's API. To obtain a connection string, you will need to retrieve an API Key from MailChimp or register a MailChimp app.

Once you have connected to MailChimp, you can import a variety of tables from the MailChimp API.

This section describes how to install the MailChimp connector, how to connect to MailChimp with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the MailChimp Connector

Sisense provides the MailChimp connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the MailChimp connector:

1. [Download](#) the MailChimp installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
The MailChimp connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

 Database servers	 Files	 Web Services
PostgreSQL	Microsoft Excel File	ServiceNow
Sisense ElastiCube	CSV File	Salesforce
Microsoft SQL Server	Microsoft Access File	HubSpot
Oracle Database		Gmail
MySQL Database		Dynamo DB
Generic ODBC Driver		Twitter
Generic OLE DB Driver		Box
TERADATA		Microsoft Dynamics CRM
		Amazon Redshift
		Google AdWords
		Sisense Analytics
		MailChimp
		Netsuite
		QuickBooks Online
		Stripe
		Youtube
		Splunk
	 Custom	
	Custom SQL Expression	

Connecting to the MailChimp REST API

There are two authentication methods available for connecting to MailChimp, through an API Key or through OAuth. Both methods expose the same data, the difference is in what information is required by MailChimp to authenticate your account. For the API Key, all you need is a key provided by MailChimp in your account. For OAuth, you need to register an app with MailChimp, and then pass those details to Sisense.

After you have the relevant information from MailChimp, you create a connection string and connect to MailChimp.

The steps below describe how to retrieve the API Key and OAuth credentials that need to be included in the connection string.

Using the API Key to Connect to MailChimp

The APIKey grants full access to your MailChimp account. To obtain the APIKey, log into MailChimp and click **Account > Extras > API Keys**.

With the API Key, you can connect to MailChimp by passing the key in a connection string.

Using OAuth to Connect to MailChimp

To access MailChimp's REST API from Sisense, you must provide valid Oauth MailChimp credentials. These credentials are provided by MailChimp when you register an application.

OAuth requires the authenticating user to interact with MailChimp using the browser. The driver facilitates this in various ways as described below.

Register Your Application

Follow the steps below to obtain the OAuth client credentials, the OAuthClientId and OAuthClientSecret:

1. Log into your MailChimp account and click **Account > Extras > API Keys > Register and Manage Your Apps**.
2. Enter the information you want to be displayed to users when they are prompted to grant permissions to your application. This information includes your app name, company, and website.
3. If you are making a desktop application, in the Redirect URI box enter `http://127.0.0.1`. If you are making a Web application, in the Redirect URI box enter a URL where you would like users to be redirected after they grant permissions to your application.

Authenticating through MailChimp

After setting the following connection properties, you are ready to connect:

- ▶ **OAuthClientId**: Set this to the consumer key in your app settings.
- ▶ **OAuthClientSecret**: Set this to the consumer secret in your app settings.
- ▶ **CallbackURL**: Set this to the callback URL you specified in your app settings.
- ▶ **InitiateOAuth**: Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken and OAuthAccessTokenSecret.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver then completes the OAuth process:

1. Extracts the access token from the callback URL and authenticates requests.
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Adding MailChimp Tables to Your ElastiCube

After you have retrieved your API Key or OAuth credentials from MailChimp, you provide the relevant information in a connection string. Sisense uses connection strings to connect to MailChimp and import data into the ElastiCube Manager. Each connection string contains authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

For API Keys, the connection string to MailChimp is in the following format:

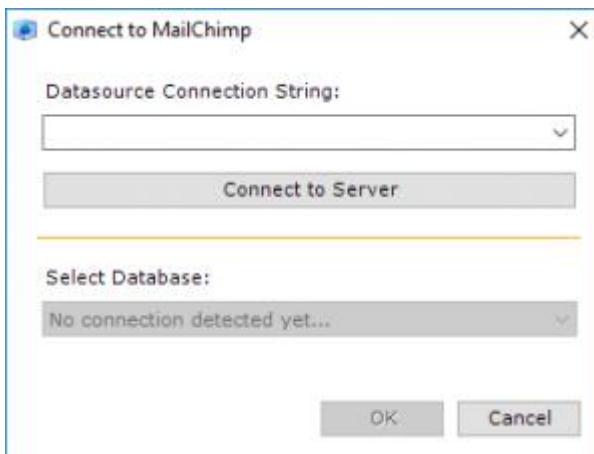
```
jdbc:mailchimp:APIKey=xxxxxxxxxxxxxxxxxxxxxx;
```

For OAuth, the connection string to MailChimp is in the following format:

```
jdbc:mailchimp:OAuthClientId=xxxxxxxx;OAuthClientSecret=xxxx  
xxxxx;CallbackURL=http://127.0.0.1;InitiateOAuth=GETANDREFRES  
H;
```

To add MailChimp data:

1. In ElastiCube Manager, click **Add Data** and then, **MailChimp**.
The Connect to MailChimp window is displayed.



2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. MailChimp is displayed in the Select Database list.
4. Click **OK**. Sisense connects to MailChimp and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**. The tables are displayed in the ElastiCube Manager.

MailChimp Tables

The Sisense MailChimp connector allows you to import the following tables into the ElastiCube Manager.

Available Tables

Name	Description
AuthorizedApps	A list of applications authorized to access the account.
AutomationEmailQueues	A summary of the queue for an email in an automation workflow.
AutomationsRemovedSubscribers	A summary of the subscribers removed from an automation workflow.
CampaignFeedback	A summary of the comment feedback for a specific campaign.
CampaignFolders	Folders for organizing campaigns
Campaigns	A summary of the campaigns within an account.
ConversationMessages	Messages from a specific conversation. Conversation tracking is a feature available to paid accounts that lets you view replies to your campaigns from inside your MailChimp account.
EcommerceCartLines	A list of an ecommerce cart's lines.
EcommerceCarts	A list of an account's ecommerce carts.
EcommerceCustomers	A list of an account's ecommerce customers.
EcommerceOrderLines	A list of an ecommerce order's lines.
EcommerceOrders	A list of an account's ecommerce orders.
EcommerceProducts	A list of an account's ecommerce products.

Name	Description
EcommerceProductVariants	A list of an ecommerce product's variants.
EcommerceStores	A list of an account's ecommerce stores.
FileManagerFiles	A listing of all available images and files within an account's gallery.
FileManagerFolders	A listing of all variable folders within an account's gallery.
ListInterestCategories	A listing of this list's interest categories.
ListInterests	A list of this category's interests
ListMemberNotes	The last 10 notes for a specific list member, based on date created.
ListMembers	Individuals who are currently or have been previously suscribed to this list, including members who have bounced or unsubscribed.
ListMergeFields	The merge field (formerly merge vars) for a given list. These correspond to merge fields in MailChimp's lists and subscriber profiles.
Lists	A collection of subscriber lists associated with this account. Lists contain subscribers who have opted-in to receive correspondence from you or your organization.
ListSegmentMembers	Individuals who are currently or have been previously suscribed to this list, including members who have bounced or unsubscribed.
ListSegments	A list of available segments.

Name	Description
ListsTwitterLeadGenCards	Twitter Lead Generation Cards for given List
ListsWebhooks	Webhooks configured for the given list.
TemplateFolders	Folders for organizing templates
Templates	A list an account's available templates.

Available Views

Name	Description
AutomationEmails	A summary of the emails in an automation workflow.
Automations	A summary of the automations within an account.
Conversations	A collection of this account's tracked conversations. Conversation tracking is a feature available to paid accounts that lets you view replies to your campaigns from inside your MailChimp account.
ListAbuse	A collection of abuse complaints for a specific list. An abuse complaint occurs when your recipient clicks to 'report spam' in their email program.
ListActivity	Up to the previous 180 days of daily detailed aggregated activity stats for a given list. Does not include AutoResponder or Automation activity.
ListClients	Top email clients used, as measured by their user-agent string
ListGrowthHistory	A month-by-month summary of a specific list's growth activity.
ListMemberActivity	The last 50 member events for a list.
ListSignupForms	Collection of List Signup Forms

Name	Description
ReportAbuse	A list of abuse complaints for a specific list.
ReportAdvice	A list of feedback based on a campaign's statistics.
ReportClickDetails	A list of URLs and unique IDs included in HTML and plain-text versions of a campaign.
ReportClickDetailsMembers	A collection of members who clicked on a specific link within a campaign.
ReportDomainPerformance	Statistics for the top-performing email domains in a campaign.
ReportEmailActivity	A list of member's subscriber activity in a specific campaign.
ReportLocations	Top open locations for a specific campaign.
Reports	A list of reports containing campaigns marked as Sent.
ReportSentTo	A list of subscribers who were sent a specific campaign.
ReportUnsubscribes	A list of members who have unsubscribed from a specific campaign.

Connecting to Microsoft Active Directory

The Sisense Microsoft Active Directory connector is a standalone connector that allows you to import data from Microsoft Active Directory's API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the Microsoft Active Directory tables.

Once you have connected to Microsoft Active Directory, you can import a variety of tables from the Microsoft Active Directory API.

This section describes how to install the Microsoft Active Directory connector, how to connect to Microsoft Active Directory with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the Microsoft Active Directory Connector

Sisense provides the Microsoft Active Directory connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Microsoft Active Directory connector:

1. [Download](#) the Microsoft Active Directory installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The Microsoft Active Directory connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

Connecting to the Microsoft Active Directory

Sisense uses connection strings to connect to Microsoft Active Directory and import data into the ElastiCube Manager.

The connection string to connect to Microsoft Active Directory has the following structure:

`jdbc:Microsoft Active Directory:Property1=Value1;Property2=Value2;`

The following is an example of a Microsoft Active Directory connection string:

`jdbc:User=xxxxxxxxx;Password=xxxxxxxxxx;Server=xxxxxxxxxx;Port=389;BaseDN=CN=xxxx,DC=xxxx,DC=xxxxxx,DC=xxxxxx;`

To establish a connection, the following properties under the Authentication section must be provided:

- ▶ Valid User and Password credentials (e.g., Domain\\BobF or cn=Bob F,ou=Employees,dc=Domain).
- ▶ Server information, including the IP or host name of the Server and the Port.
- ▶ BaseDN will limit the scope of LDAP searches to the height of the distinguished name provided.

Note: Specifying a narrow BaseDN may greatly increase performance; for example, cn=users,dc=domain will only return results contained within cn=users and its children.

- ▶ If you define your own custom schemas to work with your ActiveDirectory object classes, set Location to the path to the folder containing the schema files.

Note: To switch between accounts, you need to delete the file OAuthsettings.txt located at

..\Users\xxx\AppData\Roaming\CDATA\Microsoft Active Directory
Data Provider.

To add Microsoft Active Directory data:

1. In ElastiCube Manager, click **Add Data** and then, **Microsoft Active Directory**. The Connect to Microsoft Active Directory window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Microsoft Active Directory is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Microsoft Active Directory and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.
6. The tables are displayed in the ElastiCube Manager.

Switching between Accounts

When you connect to the Microsoft Active Directory data source, Sisense saves your OAuth values in the file OAuthsettings.txt file located at ..\Users\xxx\AppData\Roaming\CDATA\Microsoft Active Directory Data Provider on your Sisense server. To connect to the Microsoft Active Directory data source with another user on the same machine, you must delete the OAuthsettings.txt file. Sisense will then generate a new file for that user.

Another option to support multiple users is to define the location and file name of an OAuthsettings file for each unique user in your connection string through the **OAuthSettingsLocation** parameter.

When each user connects to the data source, Sisense generates the OAuth file with the file name you specify in the location you define. In the examples below, two users are allowed to access the Microsoft Active Directory data source and for each user, Sisense generates a

file that contains that user's OAuth values in the location defined in the string.

```
jdbc:MicrosoftActiveDirectory:OAuthSettingsLocation=C:\MicrosoftActiveDirectory\auth\john.txt;OAuthClientId=11276856774486;
```

```
OAuthClientSecret=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhost/;
```

```
jdbc:MicrosoftActiveDirectory:OAuthSettingsLocation=C:\MicrosoftActiveDirectory\auth\sally.txt;OAuthClientId=11276856774486;
```

```
OAuthClientSecret=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhost/;
```

In the example above, two OAuth files are created, one for John and one for Sally in the location C:\Microsoft Active Directory\auth\.

This is useful if you support many users who each need to access the Microsoft Active Directory data source.

Microsoft Active Directory Tables

Microsoft Active Directory's RESTful APIs expose the following Microsoft Active Directory tables that you can import into the ElastiCube Manager through the Sisense Microsoft Active Directory connector:

Available Tables

Name	Description
Account	The account object class is used to define entries that represent computer accounts.

Name	Description
ApplicationEntity	X.500 base class for applications: Directory Service only uses subclass MSFT-DSA.
ApplicationProcess	X.500 base class for applications: Exchange only uses subclass DSA-Application.
ApplicationSettings	Base class for server-specific application settings.
ApplicationSiteSettings	Contains all site-specific settings.
ApplicationVersion	Can be used by application developers to store version information about their application or its schema.
BuiltinDomain	The container that holds the default groups for a domain.
CertificationAuthority	Represents a process that issues public key certificates, for example, a Certificate Server.
Computer	This class represents a computer account in the domain.
Contact	This class contains information about a person or company that you may need to contact on a regular basis.
Events	Query the Events for a Target based on either the Target or SearchTerms. May require the user_events permission.
CRLDistributionPoint	The object holding Certificate, Authority, and Delta Revocation lists.
DHCPClass	Represents a DHCP Server (or set of servers).
DnsNode	Holds the DNS resource records for a single host.
DnsZone	The container for DNS Nodes. Holds zone metadata.

Name	Description
Domain	Contains information about a domain.
DomainDNS	Windows NT domain with DNS-based (DC=) naming.
DomainPolicy	Defines the local security authority policy for one or more domains.
DomainRelatedObject	The domainRelatedObject object class is used to define an entry that represents a series of documents.
ForeignSecurityPrincipal	The Security Principal from an external source.
Group	Stores a list of user names. Used to apply security principals on resources.
GroupOfNames	Used to define entries that represent an unordered set of names that represent individual objects or other groups of names.
GroupOfUniqueNames	Defines the entries for a group of unique names. In general, used to store account objects.
GroupPolicyContainer	This represents the Group Policy Object. It is used to define group policies.
IpHost	Represents an abstraction of a host or other IP device.
IpNetwork	Represents an abstraction of a network. The distinguished name value of the Common-Name attribute denotes the canonical name of the network.
Organization	Stores information about a company or organization.
OrganizationalPerson	This class is used for objects that contain organizational information about a user, such as the employee number, department, manager, title, office address, and so on.
OrganizationalRole	This class is used for objects that contain information that pertains to a position or role within an organization, such as a system administrator,

Name	Description
	manager, and so on. It can also be used for a nonhuman identity in an organization.
OrganizationalUnit	A container for storing users, computers, and other account objects.
Person	Contains personal information about a user.
PosixAccount	Represents an abstraction of an account with Portable Operating System Interface (POSIX) attributes.
PosixGroup	Represents an abstraction of a group of accounts.
PrintQueue	Contains information about a print queue.
SecurityObject	This is an auxiliary class that is used to identify security principals.
SecurityPrincipal	Contains the security information for an object.
Server	This class represents a server computer in a site.
Site	A container for storing server objects. Represents a physical location that contains computers. Used to manage replication.
Top	The top level class from which all classes are derived.
TrustedDomain	An object that represents a domain trusted by (or trusting) the local domain.
User	This class is used to store information about an employee or contractor who works for an organization. It is also possible to apply this class to long term visitors.

Limitations

1. Accumulated builds are supported because all tables have string columns.
2. Aggregate functions are not supported

Connecting to Microsoft Dynamics CRM

The Sisense Microsoft Dynamics CRM connector is a standalone connector that allows you to import data from Microsoft Dynamics CRM's API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the Microsoft Dynamics CRM tables. To obtain a connection string, you will need to obtain your credentials from Microsoft Dynamics CRM.

Once you have connected to Microsoft Dynamics CRM, you can import a variety of tables from the Microsoft Dynamics CRM API.

This section describes how to install the Microsoft Dynamics CRM connector, how to connect to Microsoft Dynamics CRM with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the Microsoft Dynamics CRM Connector

Sisense provides the Microsoft Dynamics CRM connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Microsoft Dynamics CRM connector:

1. [Download](#) the Microsoft Dynamics CRM installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The Microsoft Dynamics CRM connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

Connecting to the Microsoft Dynamics CRM

The connection string used to connect to the Microsoft Dynamics CRM connects to various instances you have in the Microsoft Dynamics CRM. By providing your Microsoft Dynamics CRM credentials as values in the connection string, you can add Microsoft Dynamic CRM tables to the ElastiCube.

Adding Microsoft Dynamics CRM Tables to your ElastiCube

Sisense uses connection strings to connect to Microsoft Dynamics CRM and import data into the ElastiCube Manager.

The connection string to connect to Microsoft Dynamics CRM has the following structure:

```
jdbc:dynamicscrm:Property1=Value1;Property2=Value2;
```

The following is an example of a Microsoft Dynamics CRM connection string:

```
jdbc:dynamicscrm:User=xxxxxxxxxx;Password=xxxxxxxxxx;URL=http  
s://xxxxxxxxx.dynamics.com/;CRMVersion=CRM Online Office 365;
```

To connect to your instance, set the User and Password properties, under the Authentication section, to valid Dynamics CRM user credentials and set the Url to a valid Dynamics CRM server organization root.

Additionally, set the CRMVersion property. Supported versions are CRM Online Office 365, CRM 4.0, CRM 2011, CRM 2013, CRM 2015.

Note: To switch between accounts, you need to delete the file OAuthsettings.txt file located at
.\\Users\\xxx\\AppData\\Roaming\\CData\\Microsoft Dynamics CRM

Data Provider or you can add OAuthSettingsLocation to the connection string. See [Switching between Accounts](#) for more information.

To add Microsoft Dynamics CRM data:

1. In ElastiCube Manager, click **Add Data** and then, **Microsoft Dynamics CRM**. The Connect to Microsoft Dynamics CRM window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Microsoft Dynamics CRM is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Microsoft Dynamics CRM and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.
6. The tables are displayed in the ElastiCube Manager.

Switching between Accounts

When you connect to the Microsoft Dynamics CRM data source, Sisense saves your OAuth values in the file OAuthsettings.txt file located at .\Users\xxx\AppData\Roaming\CDATA\Microsoft Dynamics CRM Data Provider on your Sisense server. To connect to the Microsoft Dynamics CRM data source with another user on the same machine, you must delete the OAuthsettings.txt file. Sisense will then generate a new file for that user.

Another option to support multiple users is to define the location and file name of an OAuthsettings file for each unique user in your connection string through the **OAuthSettingsLocation** parameter.

When each user connects to the data source, Sisense generates the

OAuth file with the file name you specify in the location you define. In the examples below, two users are allowed to access the Microsoft Dynamics CRM data source and for each user, Sisense generates a file that contains that user's OAuth values in the location defined in the string.

```
jdbc:MicrosoftDynamicsCRM:OAuthSettingsLocation=C:\MicrosoftDynamicsCRM\auth\john.txt;OAuthClientId=11276856774486;
```

```
OAuthClientSecret=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhost/;
```

```
jdbc:MicrosoftDynamicsCRM:OAuthSettingsLocation=C:\MicrosoftDynamicsCRM\auth\sally.txt;OAuthClientId=11276856774486;
```

```
OAuthClientSecret=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhost/;
```

In the example above, two OAuth files are created, one for John and one for Sally in the location C:\MicrosoftDynamicsCRM\auth\.

This is useful if you support many users who each need to access the Microsoft Dynamics CRM data source.

Microsoft Dynamics CRM Tables

Microsoft Dynamics CRM exposes the following Microsoft Dynamics CRM tables that you can import into the ElastiCube Manager through the Sisense Microsoft Dynamics CRM connector:

Available Tables

Name	Description
Accounts	Create, update, delete, and query Account entities in Dynamics CRM.
ActivityMimeTypeAttachment	Create, update, delete, and query ActivityMimeTypeAttachment entities in Dynamics CRM.
ActivityParty	Create, update, delete, and query ActivityParty entities in Dynamics CRM.
ActivityPointer	Create, update, delete, and query ActivityPointer entities in Dynamics CRM.
Annotation	Create, update, delete, and query Annotation entities in Dynamics CRM.
AnnualFiscalCalendar	Create, update, delete, and query annual fiscal calendar entities in Dynamics CRM.
ApplicationFile	Create, update, delete, and query ApplicationFile entities in Dynamics CRM.
Appointment	Create, update, delete, and query Appointment entities in Dynamics CRM.
AsyncOperation	Create, update, delete, and query AsyncOperation entities in Dynamics CRM.
Attachment	Create, update, delete, and query Attachment entities in Dynamics CRM.
AttributeMap	Create, update, delete, and query AttributeMap entities in Dynamics CRM.

Name	Description
Audit	Create, update, delete, and query Audit entities in Dynamics CRM.
BulkDeleteFailure	Create, update, delete, and query BulkDeleteFailure entities in Dynamics CRM.
BulkDeleteOperation	Create, update, delete, and query BulkDeleteOperation entities in Dynamics CRM.
BulkOperation	Create, update, delete, and query BulkOperation entities in Dynamics CRM.
BulkOperationLog	This is a table representing the BulkOperationLog entities in Dynamics CRM.
BusinessUnit	This is a table representing the BusinessUnit entities in Dynamics CRM.
BusinessUnitMap	This is a table representing the BusinessUnitMap entities in Dynamics CRM.
BusinessUnitNewsArticle	This is a table representing the BusinessUnitNewsArticle entities in Dynamics CRM.
Calendar	This is a table representing the Calendar entities in Dynamics CRM.
CalendarRule	This is a table representing the CalendarRule entities in Dynamics CRM.
Campaign	This is a table representing the Campaign entities in Dynamics CRM.

Name	Description
CampaignActivity	This is a table representing the CampaignActivityItem entities in Dynamics CRM.
CampaignActivityItem	This is a table representing the CampaignActivityItem entities in Dynamics CRM.
CampaignItem	This is a table representing the CampaignItem entities in Dynamics CRM.
CampaignResponse	This is a table representing the CampaignResponse entities in Dynamics CRM.
ClientUpdate	This is a table representing the ClientUpdate entities in Dynamics CRM.
ColumnMapping	This is a table representing the ColumnMapping entities in Dynamics CRM.
Commitment	This is a table representing the Commitment entities in Dynamics CRM.
Competitor	This is a table representing the Competitor entities in Dynamics CRM.
CompetitorAddress	This is a table representing the CompetitorAddress entities in Dynamics CRM.
CompetitorProduct	This is a table representing the CompetitorProduct entities in Dynamics CRM.
CompetitorSalesLiterature	This is a table representing the CompetitorSalesLiterature entities in Dynamics CRM.

Name	Description
Connection	This is a table representing the Connection entities in Dynamics CRM.
ConnectionRole	This is a table representing the ConnectionRole entities in Dynamics CRM.
ConnectionRoleAssociation	This is a table representing the ConnectionRoleAssociation entities in Dynamics CRM.
ConnectionRoleObjectTypeCode	This is a table representing the ConnectionRoleObjectTypeCode entities in Dynamics CRM.
ConstraintBasedGroup	This is a table representing the ConstraintBasedGroup entities in Dynamics CRM.
Contact	This is a table representing the Contact entities in Dynamics CRM.
ContactInvoices	This is a table representing the ContactInvoices entities in Dynamics CRM.
ContactLeads	This is a table representing the ContactLeads entities in Dynamics CRM.
ContactOrders	This is a table representing the ContactOrders entities in Dynamics CRM.
ContactQuotes	This is a table representing the ContactQuotes entities in Dynamics CRM.
Contract	This is a table representing the Contract entities in Dynamics CRM.

Name	Description
ContractDetail	This is a table representing the ContractDetail entities in Dynamics CRM.
ContractTemplate	This is a table representing the ContractTemplate entities in Dynamics CRM.
CustomerAddress	This is a table representing the CustomerAddress entities in Dynamics CRM.
CustomerOpportunityRole	This is a table representing the CustomerOpportunityRole entities in Dynamics CRM.
CustomerRelationship	This is a table representing the CustomerRelationship entities in Dynamics CRM.
Dependency	This is a table representing the Dependency entities in Dynamics CRM.
DependencyNode	This is a table representing the DependencyNode entities in Dynamics CRM.
Discount	This is a table representing the Discount entities in Dynamics CRM.
DiscountType	This is a table representing the DiscountType entities in Dynamics CRM.
DisplayString	This is a table representing the DisplayString entities in Dynamics CRM.
DisplayStringMap	This is a table representing the DisplayStringMap entities in Dynamics CRM.

Name	Description
DocumentIndex	This is a table representing the DocumentIndex entities in Dynamics CRM.
DuplicateRecord	This is a table representing the DuplicateRecord entities in Dynamics CRM.
DuplicateRule	This is a table representing the DuplicateRule entities in Dynamics CRM.
DuplicateRuleCondition	This is a table representing the DuplicateRuleCondition entities in Dynamics CRM.
Email	This is a table representing the Email entities in Dynamics CRM.
EmailHash	This is a table representing the EmailHash entities in Dynamics CRM.
EmailSearch	This is a table representing the EmailSearch entities in Dynamics CRM.
EntityMap	This is a table representing the EntityMap entities in Dynamics CRM.
Equipment	This is a table representing the Equipment entities in Dynamics CRM.
Fax	This is a table representing the Fax entities in Dynamics CRM.
FieldPermission	This is a table representing the FieldPermission entities in Dynamics CRM.

Name	Description
FieldSecurityProfile	This is a table representing the FieldSecurityProfile entities in Dynamics CRM.
FilterTemplate	This is a table representing the FilterTemplate entities in Dynamics CRM.
FixedMonthlyFiscalCalendar	This is a table representing the FixedMonthlyFiscalCalendar entities in Dynamics CRM.
Goal	This is a table representing the Goal entities in Dynamics CRM.
GoalRollupQuery	This is a table representing the GoalRollupQuery entities in Dynamics CRM.
Import	This is a table representing the ImportData entities in Dynamics CRM.
ImportData	This is a table representing the ImportEntityMapping entities in Dynamics CRM.
ImportEntityMapping	This is a table representing the ImportEntityMapping entities in Dynamics CRM.
ImportFile	This is a table representing the ImportFile entities in Dynamics CRM.
ImportJob	This is a table representing the ImportJob entities in Dynamics CRM.
ImportLog	This is a table representing the ImportLog entities in Dynamics CRM.

Name	Description
ImportMap	This is a table representing the ImportMap entities in Dynamics CRM.
Incident	This is a table representing the Incident entities in Dynamics CRM.
IncidentResolution	This is a table representing the IncidentResolution entities in Dynamics CRM.
IntegrationStatus	This is a table representing the IntegrationStatus entities in Dynamics CRM.
InternalAddress	This is a table representing the InternalAddress entities in Dynamics CRM.
InterProcessLock	This is a table representing the InterProcessLock entities in Dynamics CRM.
InvalidDependency	This is a table representing the InvalidDependency entities in Dynamics CRM.
Invoice	This is a table representing the Invoice entities in Dynamics CRM.
InvoiceDetail	This is a table representing the InvoiceDetail entities in Dynamics CRM.
IsvConfig	This is a table representing the IsvConfig entities in Dynamics CRM.
KbArticle	This is a table representing the KbArticle entities in Dynamics CRM.

Name	Description
KbArticleComment	This is a table representing the KbArticleComment entities in Dynamics CRM.
KbArticleTemplate	This is a table representing the KbArticleTemplate entities in Dynamics CRM.
Lead	This is a table representing the Lead entities in Dynamics CRM.
LeadAddress	This is a table representing the LeadAddress entities in Dynamics CRM.
LeadCompetitors	This is a table representing the LeadCompetitors entities in Dynamics CRM.
LeadProduct	This is a table representing the LeadProduct entities in Dynamics CRM.
Letter	This is a table representing the Letter entities in Dynamics CRM.
License	This is a table representing the License entities in Dynamics CRM.
List	This is a table representing the List entities in Dynamics CRM.
ListMember	This is a table representing the ListMember entities in Dynamics CRM.
LookUpMapping	This is a table representing the LookUpMapping entities in Dynamics CRM.

Name	Description
MailMergeTemplate	This is a table representing the MailMergeTemplate entities in Dynamics CRM.
Metric	This is a table representing the Metric entities in Dynamics CRM.
MonthlyFiscalCalendar	This is a table representing the MonthlyFiscalCalendar entities in Dynamics CRM.
Notification	This is a table representing the Notification entities in Dynamics CRM.
Opportunity	This is a table representing the Opportunity entities in Dynamics CRM.
OpportunityClose	This is a table representing the OpportunityClose entities in Dynamics CRM.
OpportunityCompetitors	This is a table representing the OpportunityCompetitors entities in Dynamics CRM.
OpportunityProduct	This is a table representing the OpportunityProduct entities in Dynamics CRM.
OrderClose	This is a table representing the OrderClose entities in Dynamics CRM.
Organization	This is a table representing the Organization entities in Dynamics CRM.
OrganizationStatistic	This is a table representing the OrganizationStatistic entities in Dynamics CRM.

Name	Description
OrganizationUI	This is a table representing the OrganizationUI entities in Dynamics CRM.
Owner	This is a table representing the Owner entities in Dynamics CRM.
OwnerMapping	This is a table representing the OwnerMapping entities in Dynamics CRM.
PhoneCall	This is a table representing the PhoneCall entities in Dynamics CRM.
PickListMapping	This is a table representing the PickListMapping entities in Dynamics CRM.
PluginAssembly	This is a table representing the PluginAssembly entities in Dynamics CRM.
plug-intype	This is a table representing the plug-in type entities in Dynamics CRM.
PluginTypeStatistic	This is a table representing the PluginTypeStatistic entities in Dynamics CRM.
PriceLevel	This is a table representing the PriceLevel entities in Dynamics CRM.
PrincipalAttributeAccessMap	This is a table representing the PrincipalAttributeAccessMap entities in Dynamics CRM.
PrincipalEntityMap	This is a table representing the PrincipalEntityMap entities in Dynamics CRM.

Name	Description
PrincipalObjectAccess	This is a table representing the PrincipalObjectAccess entities in Dynamics CRM.
PrincipalObjectAttributeAccess	This is a table representing the PrincipalObjectAttributeAccess entities in Dynamics CRM.
Privilege	This is a table representing the Privilege entities in Dynamics CRM.
PrivilegeObjectTypeCodes	This is a table representing the PrivilegeObjectTypeCodes entities in Dynamics CRM.
ProcessSession	This is a table representing the ProcessSession entities in Dynamics CRM.
Product	This is a table representing the Product entities in Dynamics CRM.
ProductAssociation	This is a table representing the ProductAssociation entities in Dynamics CRM.
ProductPriceLevel	This is a table representing the ProductPriceLevel entities in Dynamics CRM.
ProductSalesLiterature	This is a table representing the ProductSalesLiterature entities in Dynamics CRM.
ProductSubstitute	This is a table representing the ProductSubstitute entities in Dynamics CRM.
Publisher	This is a table representing the Publisher entities in Dynamics CRM.

Name	Description
PublisherAddress	This is a table representing the PublisherAddress entities in Dynamics CRM.
QuarterlyFiscalCalendar	This is a table representing the QuarterlyFiscalCalendar entities in Dynamics CRM.
Queue	This is a table representing the Queue entities in Dynamics CRM.
QueueItem	This is a table representing the QueueItem entities in Dynamics CRM.
Quote	This is a table representing the Quote entities in Dynamics CRM.
QuoteClose	This is a table representing the QuoteClose entities in Dynamics CRM.
QuoteDetail	This is a table representing the QuoteDetail entities in Dynamics CRM.
RecurrenceRule	This is a table representing the RecurrenceRule entities in Dynamics CRM.
RecurringAppointmentMaster	This is a table representing the RecurringAppointmentMaster entities in Dynamics CRM.
RelationshipRole	This is a table representing the RelationshipRole entities in Dynamics CRM.
RelationshipRoleMap	This is a table representing the RelationshipRoleMap entities in Dynamics CRM.

Name	Description
Report	This is a table representing the Report entities in Dynamics CRM.
ReportCategory	This is a table representing the ReportCategory entities in Dynamics CRM.
ReportEntity	This is a table representing the ReportEntity entities in Dynamics CRM.
ReportLink	This is a table representing the ReportLink entities in Dynamics CRM.
ReportVisibility	This is a table representing the ReportVisibility entities in Dynamics CRM.
Resource	This is a table representing the Resource entities in Dynamics CRM.
ResourceGroup	This is a table representing the ResourceGroup entities in Dynamics CRM.
ResourceGroupExpansion	This is a table representing the ResourceGroupExpansion entities in Dynamics CRM.
ResourceSpec	This is a table representing the ResourceSpec entities in Dynamics CRM.
RibbonCommand	This is a table representing the RibbonCommand entities in Dynamics CRM.
RibbonContextGroup	This is a table representing the RibbonContextGroup entities in Dynamics CRM.

Name	Description
RibbonCustomization	This is a table representing the RibbonCustomization entities in Dynamics CRM.
RibbonDiff	This is a table representing the RibbonDiff entities in Dynamics CRM.
RibbonRule	This is a table representing the RibbonRule entities in Dynamics CRM.
RibbonTabToCommandMap	This is a table representing the RibbonTabToCommandMap entities in Dynamics CRM.
Role	This is a table representing the Role entities in Dynamics CRM.
RolePrivileges	This is a table representing the RolePrivileges entities in Dynamics CRM.
RoleTemplate	This is a table representing the RoleTemplate entities in Dynamics CRM.
RoleTemplatePrivileges	This is a table representing the RoleTemplatePrivileges entities in Dynamics CRM.
RollupField	This is a table representing the RollupField entities in Dynamics CRM.
SalesLiterature	This is a table representing the SalesLiterature entities in Dynamics CRM.
SalesLiteratureItem	This is a table representing the SalesLiteratureItem entities in Dynamics CRM.

Name	Description
SalesOrder	This is a table representing the SalesOrder entities in Dynamics CRM.
SalesOrderDetail	This is a table representing the SalesOrderDetail entities in Dynamics CRM.
SalesProcessInstance	This is a table representing the SalesProcessInstance entities in Dynamics CRM.
SavedQuery	This is a table representing the SavedQuery entities in Dynamics CRM.
SavedQueryVisualization	This is a table representing the SavedQueryVisualization entities in Dynamics CRM.
SdkMessage	This is a table representing the SdkMessage entities in Dynamics CRM.
SdkMessageFilter	This is a table representing the SdkMessageFilter entities in Dynamics CRM.
SdkMessagePair	This is a table representing the SdkMessagePair entities in Dynamics CRM.
SdkMessageProcessingStep	This is a table representing the SdkMessageProcessingStep entities in Dynamics CRM.
SdkMessageProcessingStepImage	This is a table representing the SdkMessageProcessingStepImage entities in Dynamics CRM.
SdkMessageProcessingStepSecureConfig	This is a table representing the SdkMessageProcessingStepSecureConfig entities in Dynamics CRM.

Name	Description
SdkMessageRequest	This is a table representing the SdkMessageRequest entities in Dynamics CRM.
SdkMessageRequestField	This is a table representing the SdkMessageRequestField entities in Dynamics CRM.
SdkMessageResponse	This is a table representing the SdkMessageResponse entities in Dynamics CRM.
SdkMessageResponseField	This is a table representing the SdkMessageResponseField entities in Dynamics CRM.
SemiAnnualFiscalCalendar	This is a table representing the SemiAnnualFiscalCalendar entities in Dynamics CRM.
Service	This is a table representing the Service entities in Dynamics CRM.
ServiceAppointment	This is a table representing the ServiceAppointment entities in Dynamics CRM.
ServiceContractContacts	This is a table representing the ServiceContractContacts entities in Dynamics CRM.
ServiceEndpoint	This is a table representing the ServiceEndpoint entities in Dynamics CRM.
SharePointDocumentLocation	This is a table representing the SharePointDocumentLocation entities in Dynamics CRM.
SharePointSite	This is a table representing the SharePointSite entities in Dynamics CRM.

Name	Description
Site	This is a table representing the Site entities in Dynamics CRM.
SiteMap	This is a table representing the SiteMap entities in Dynamics CRM.
Solution	This is a table representing the Solution entities in Dynamics CRM.
SolutionComponent	This is a table representing the SolutionComponent entities in Dynamics CRM.
StatusMap	This is a table representing the StatusMap entities in Dynamics CRM.
StringMap	This is a table representing the StringMap entities in Dynamics CRM.
Subject	This is a table representing the Subject entities in Dynamics CRM.
Subscription	This is a table representing the Subscription entities in Dynamics CRM.
SubscriptionClients	This is a table representing the SubscriptionClients entities in Dynamics CRM.
SubscriptionManuallyTrackedObject	This is a table representing the SubscriptionManuallyTrackedObject entities in Dynamics CRM.
SubscriptionSyncInfo	This is a table representing the SubscriptionSyncInfo entities in Dynamics CRM.

Name	Description
SubscriptionTrackingDeletedObject	This is a table representing the SubscriptionTrackingDeletedObject entities in Dynamics CRM.
SystemForm	This is a table representing the SystemForm entities in Dynamics CRM.
SystemUser	This is a table representing the SystemUser entities in Dynamics CRM.
SystemUserBusinessUnitEntityMap	This is a table representing the SystemUserBusinessUnitEntityMap entities in Dynamics CRM.
SystemUserLicenses	This is a table representing the SystemUserLicenses entities in Dynamics CRM.
SystemUserPrincipals	This is a table representing the SystemUserPrincipals entities in Dynamics CRM.
SystemUserProfiles	This is a table representing the SystemUserProfiles entities in Dynamics CRM.
SystemUserRoles	This is a table representing the SystemUserRoles entities in Dynamics CRM.
Task	This is a table representing the Task entities in Dynamics CRM.
Team	This is a table representing the Team entities in Dynamics CRM.
TeamMembership	This is a table representing the TeamMembership entities in Dynamics CRM.

Name	Description
TeamProfiles	This is a table representing the TeamProfiles entities in Dynamics CRM.
TeamRoles	This is a table representing the TeamRoles entities in Dynamics CRM.
Template	This is a table representing the Template entities in Dynamics CRM.
Territory	This is a table representing the Territory entities in Dynamics CRM.
TimeZoneDefinition	This is a table representing the TimeZoneDefinition entities in Dynamics CRM.
TimeZoneLocalizedName	This is a table representing the TimeZoneLocalizedName entities in Dynamics CRM.
TimeZoneRule	This is a table representing the TimeZoneRule entities in Dynamics CRM.
TransactionCurrency	This is a table representing the TransactionCurrency entities in Dynamics CRM.
TransformationMapping	This is a table representing the TransformationMapping entities in Dynamics CRM.
TransformationParameterMapping	This is a table representing the TransformationParameterMapping entities in Dynamics CRM.
UnresolvedAddress	This is a table representing the UnresolvedAddress entities in Dynamics CRM.

Name	Description
UoM	This is a table representing the UoM entities in Dynamics CRM.
UoMSchedule	This is a table representing the UoMSchedule entities in Dynamics CRM.
UserEntityInstanceData	This is a table representing the UserEntityInstanceData entities in Dynamics CRM.
UserEntityUISettings	This is a table representing the UserEntityUISettings entities in Dynamics CRM.
UserFiscalCalendar	This is a table representing the UserFiscalCalendar entities in Dynamics CRM.
UserForm	This is a table representing the UserForm entities in Dynamics CRM.
UserQuery	This is a table representing the UserQuery entities in Dynamics CRM.
UserQueryVisualization	This is a table representing the UserQueryVisualization entities in Dynamics CRM.
UserSettings	This is a table representing the UserSettings entities in Dynamics CRM.
WebResource	This is a table representing the WebResource entities in Dynamics CRM.
WebWizard	This is a table representing the WebWizard entities in Dynamics CRM.

Name	Description
WizardAccessPrivilege	This is a table representing the WizardAccessPrivilege entities in Dynamics CRM.
WizardPage	This is a table representing the WizardPage entities in Dynamics CRM.
Workflow	This is a table representing the Workflow entities in Dynamics CRM.
WorkflowDependency	This is a table representing the WorkflowDependency entities in Dynamics CRM.
WorkflowLog	This is a table representing the WorkflowLog entities in Dynamics CRM.
WorkflowWaitSubscription	This is a table representing the WorkflowWaitSubscription entities in Dynamics CRM.

Connecting to MongoDB

MongoDB Overview

The ElastiCube Manager enables easy and quick access to databases, tables and views contained within MongoDB databases.

Sisense provides a MongoDB connector for the ElastiCube. [Click here](#) to download the driver.

Note: Before connecting to MongoDB database with Sisense, please note that MongoDB is an unstructured database, and therefore tables may be flattened with additional tables being created for nested items.

Upgrading to MongoDB v2.0

The latest version of MongoDB provides enhanced security and improved handling for complex MongoDB models.

The latest version of MongoDB makes significant breaking changes, if you use a version of MongoDB prior to version 2.0, and you are satisfied with your service, you can continue without upgrading.

When to Upgrade?

In certain circumstances it is recommended or necessary that you upgrade your driver for MongoDB version 2.0, for example:

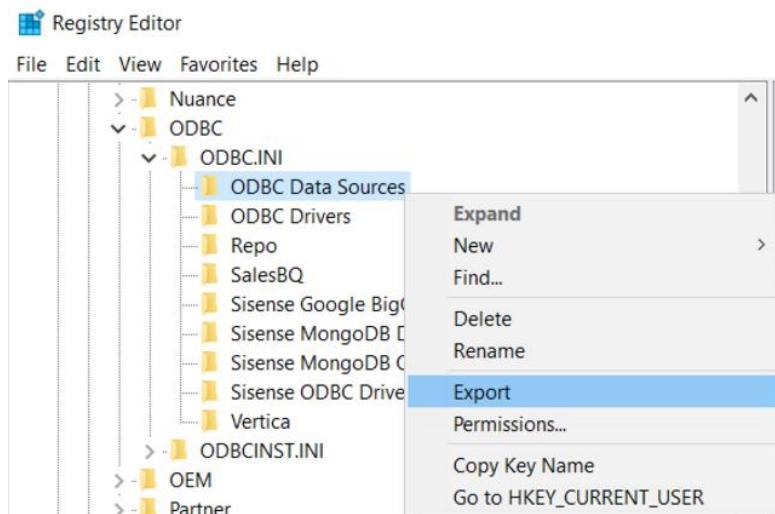
1. You are new to Sisense and MongoDB
2. You want to connect with the Sisense Native REST Connector
3. You experience security-related issues in MongoDB v1.0
4. You have a really complex MongoDB model that v1.0 can't handle

Before Upgrading

When upgrading to MongoDB v2.0, your current DSN names configured locally may be deleted. As a workaround, you can export the ODBC registry entry and then import after completing the upgrade.

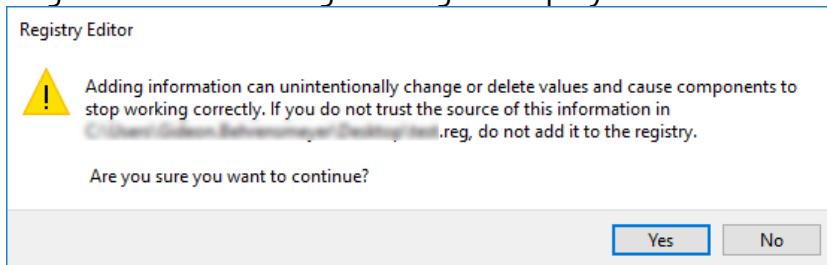
To migrate your ODBC Data Source registry file:

1. In Windows, open the Run box with by pressing the Windows key + r.
2. In the Run line, enter regedit.
3. Click **OK**.
4. Navigate to the ODBC Data Source registry entry at the following location:
HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBC.INI\ODBC Data Sources
5. Right click the entry and select **Export**.



6. Save the registry file on your machine.

7. After installing MongoDB v2, click the exported registry file. The Register Editor warning message is displayed.



8. Click **Yes** to import the entry.

After Upgrading

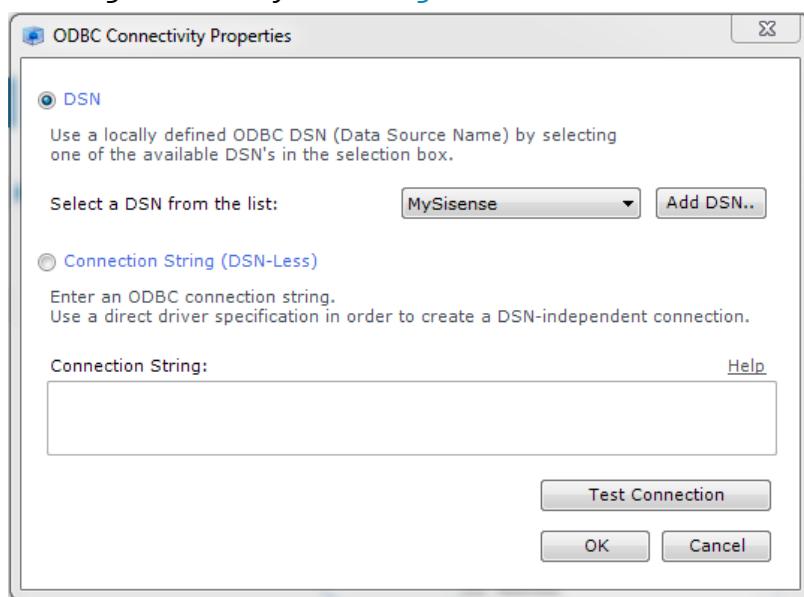
After upgrading to MongoDB v2.0, note the following:

1. The new ODBC schema creation is different than v1.0.
MongoDB v2.0 will not create virtual tables (`_vt_` tables) in the ElastiCube Manager. As a result, your current ElastiCube schema will not match schemas created before upgrading.
Create a new schema using the new v2.0 DSN.
2. The MongoDB v2.0 driver saves a JSON format of the extracted schema, as opposed to the v1.0 driver which uses XML.
3. The MongoDB v2.0 driver stores the schema extracted in MongoDB by default. This allows multiple users to work with the same extracted schema from different servers without extracting a new schema individually each time. However, any changes to the schema affects the schema for everyone else.
4. Storing the schema in the MongoDB means your DB is not **ReadOnly**. Administrators should be aware of this and protect their database with authentication and access control, or communicate this store action and its effects to all users.

Downloading and Connecting to the MongoDB ODBC Driver

To connect to MongoDB, complete the following procedures:

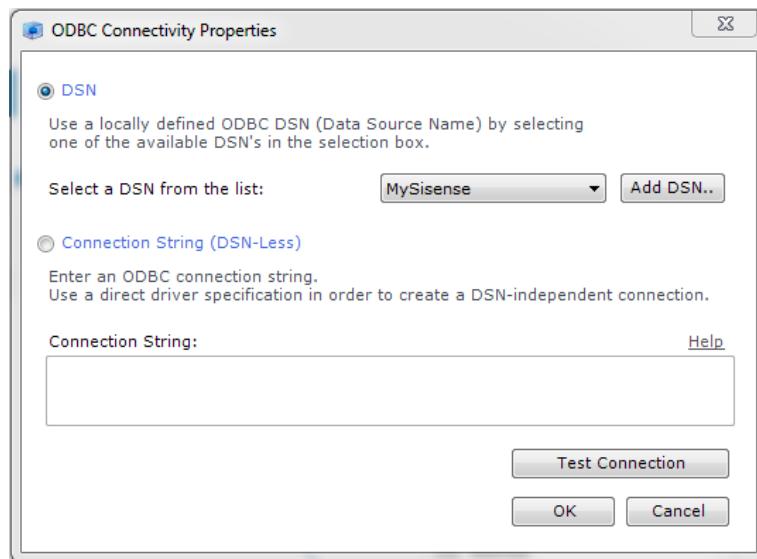
1. [Download](#) and install the MongoDB ODBC Driver.
2. In the ElastiCube Manager, click **Add Data**, and then **Generic ODBC Driver**.
3. Select **DSN**. If a DSN file has already been created, look for it in the drop-down list, and click **Test Connection**.
4. If you need to add a DSN, see the [next section](#). If your connection has been set up, you can click **OK** and review the tables generated by the [MongoDB ODBC Driver](#).



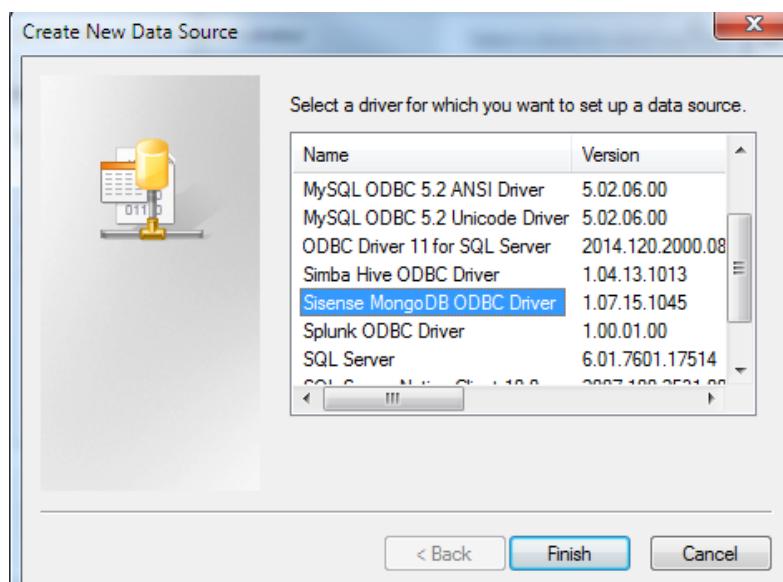
Adding a DSN

If a DSN has not yet been created, you will need to add it as follows:

1. Click **Add DSN**.
Note: To add a DSN you must run the Sisense ElastiCube Manager as an administrator.
2. Select the **System Data Source** option. The created file will apply to all users in a specific machine only. Click **Next**.

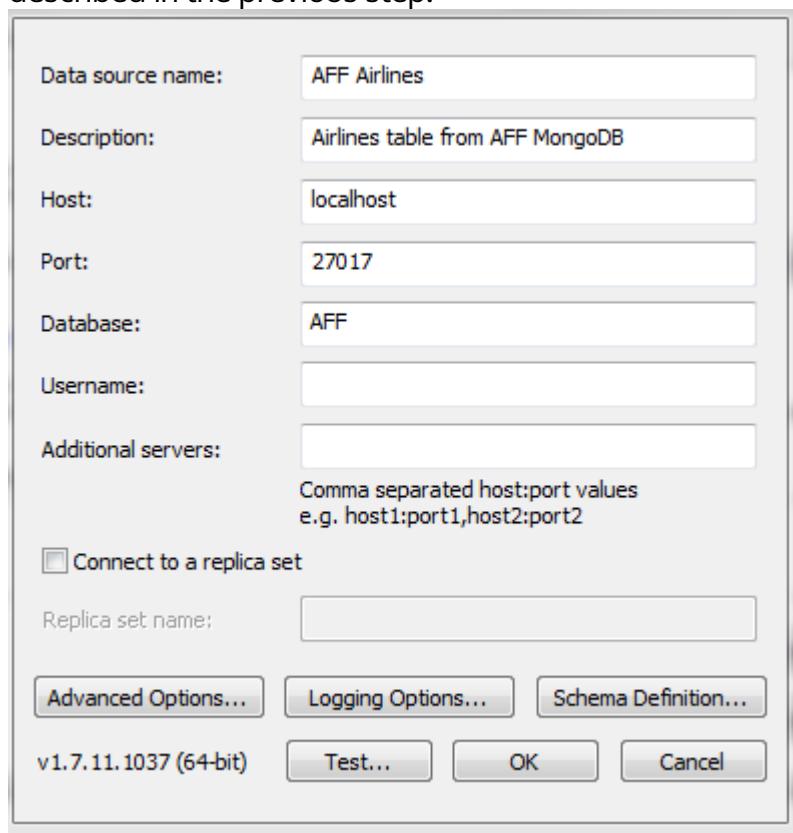


3. Select the previously installed **Sisense MongoDB ODBC Driver**, and click **Next**, and then **Finish**.



4. Open the MongoDB ODBC Driver from the ODBC Data Source Administrator.
5. Provide a **Data source name**. The DSN file will be available thereafter when the user/system data source is chosen.
6. In the **Host** field, enter the DB hosting machine's IP address (if the DB is stored locally, enter *localhost*).
7. In the **Port** field, enter the MongoDB default port which is *27017*.

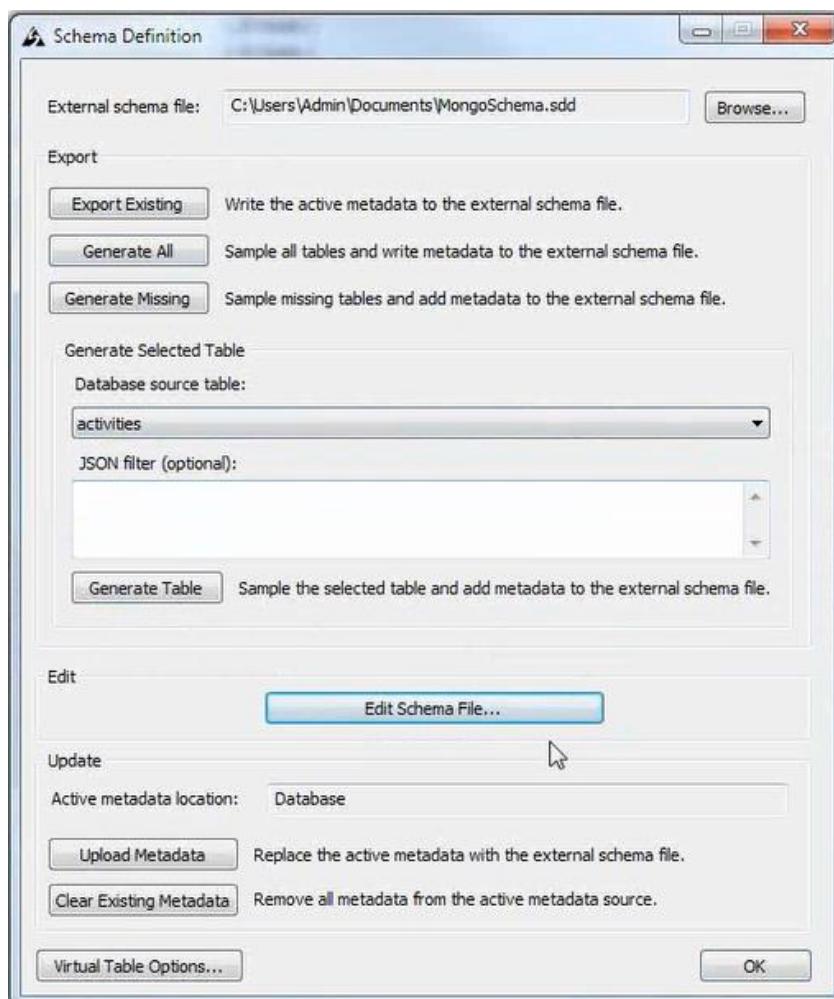
8. In the **Database** field, enter the name that can be found by connecting to the DB using Mongovue/RoboMongo. **Note:** The name is case sensitive.
9. If authentication is required, then enter the **Username**.
10. Use **Additional servers** when connecting to a shared cluster or a replica set. Type in a comma-separated list of additional servers that you need, indicate the port on which a server is listening by appending a colon (:), and the port number to the server name or IP address.
11. Select **Connect to a replica set** if you are connecting to a replica set, and type the name of the replica set in the **Replica set name** field. Make sure that the names of the secondary servers in the replica set are listed in the **Additional servers** field, as described in the previous step.



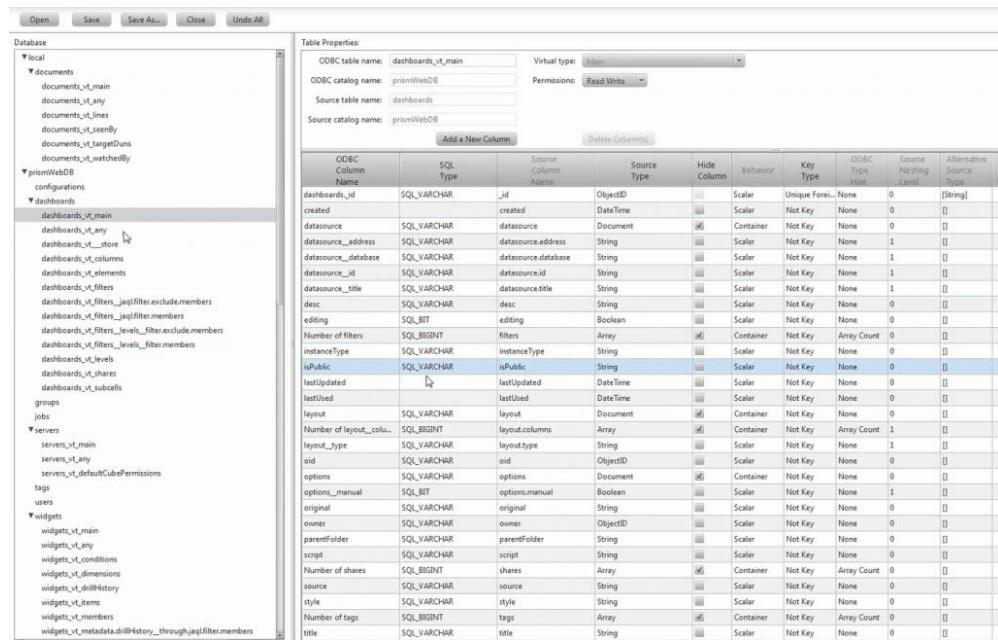
DSN Schema Definition

1. Open the MongoDB ODBC Driver from the ODBC Data Source Administrator.

2. Click **Schema Definition** to review the schema and make changes if needed:
 - ▶ **External schema file:** The driver automatically generates the schema if not manually defined (path to file is displayed). To edit an existing DSN schema, click **Browse** to locate and open the file in the ODBC Data Source Administrator.
 - ▶ **Export** options include:
 - **Export Existing:** Exports metadata that has already been generated.
 - **Generate All:** Exports metadata for all the tables in the database.
 - **Generate Missing:** Exports metadata for tables that have not been included in the schema.

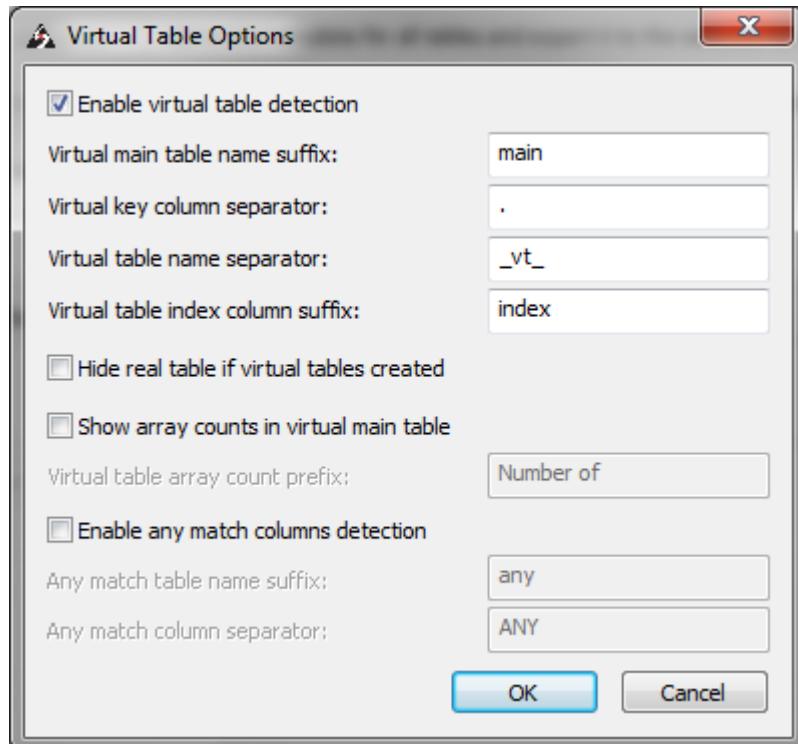


- ▶ **Edit Schema File:** Use this option to edit the schema before updating metadata (change field types, hide columns etc.).



ODBC Column Name	SQL Type	Source Column Name	Source Type	Hide Column	Behavior	Key Type	ODBC Type Hint	Source Nulling Level	Alternative Source Type
dashboards_id	SQL_VARCHAR	_id	ObjectID	<input checked="" type="checkbox"/>	Scalar	Unique Forei.	None	0	[String]
created		created	DateTime	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
datasource	SQL_VARCHAR	datasource	Document	<input checked="" type="checkbox"/>	Container	Not Key	None	0	<input type="checkbox"/>
datasource_address	SQL_VARCHAR	datasource.address	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	1	<input type="checkbox"/>
datasource_database	SQL_VARCHAR	datasource.database	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	1	<input type="checkbox"/>
datasource_id	SQL_VARCHAR	datasource.id	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	1	<input type="checkbox"/>
datasource_title	SQL_VARCHAR	datasource.title	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	1	<input type="checkbox"/>
desc	SQL_VARCHAR	desc	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
editing	SQL_BIT	editing	Boolean	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
Number of filters	SQL_BIGINT	filters	Array	<input checked="" type="checkbox"/>	Container	Not Key	Array Count	0	<input type="checkbox"/>
instanceType		instanceType	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
isPublic	SQL_VARCHAR	isPublic	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
lastUpdated		lastUpdated	DateTime	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
lastUsed		lastUsed	DateTime	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
layout	SQL_VARCHAR	layout	Document	<input checked="" type="checkbox"/>	Container	Not Key	None	0	<input type="checkbox"/>
Number of layout_colu...	SQL_BIGINT	layout.columns	Array	<input checked="" type="checkbox"/>	Container	Not Key	Array Count	1	<input type="checkbox"/>
layout_type	SQL_VARCHAR	layout.type	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	1	<input type="checkbox"/>
oid	SQL_VARCHAR	oid	ObjectID	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
options	SQL_VARCHAR	options	Document	<input checked="" type="checkbox"/>	Container	Not Key	None	0	<input type="checkbox"/>
options_manual	SQL_BIT	options.manual	Boolean	<input checked="" type="checkbox"/>	Scalar	Not Key	None	1	<input type="checkbox"/>
original	SQL_VARCHAR	original	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
owner	SQL_VARCHAR	owner	ObjectID	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
parentFolder	SQL_VARCHAR	parentFolder	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
script	SQL_VARCHAR	script	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
Number of shares	SQL_BIGINT	shares	Array	<input checked="" type="checkbox"/>	Container	Not Key	Array Count	0	<input type="checkbox"/>
source	SQL_VARCHAR	source	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
style	SQL_VARCHAR	style	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>
Number of tags	SQL_BIGINT	tags	Array	<input checked="" type="checkbox"/>	Container	Not Key	Array Count	0	<input type="checkbox"/>
title	SQL_VARCHAR	title	String	<input checked="" type="checkbox"/>	Scalar	Not Key	None	0	<input type="checkbox"/>

- ▶ **Upload Metadata:** Updates metadata after settings are complete. **Current metadata source** displays the source that has been set in **Advanced Options**.
- ▶ **Clear Existing Metadata:** Deletes all the metadata that the driver has generated for the MongoDB instance.
- ▶ **Virtual Tables Options:** Sisense enables virtual tables creation by default. Virtual tables are created when Sisense detects an array within a field of the main table. Virtual tables can be disabled or custom configured. By default, virtual tables will receive the name MAINTABLE_vt_FIELD and the main table will receive the suffix "main". We recommend leaving these settings as is.



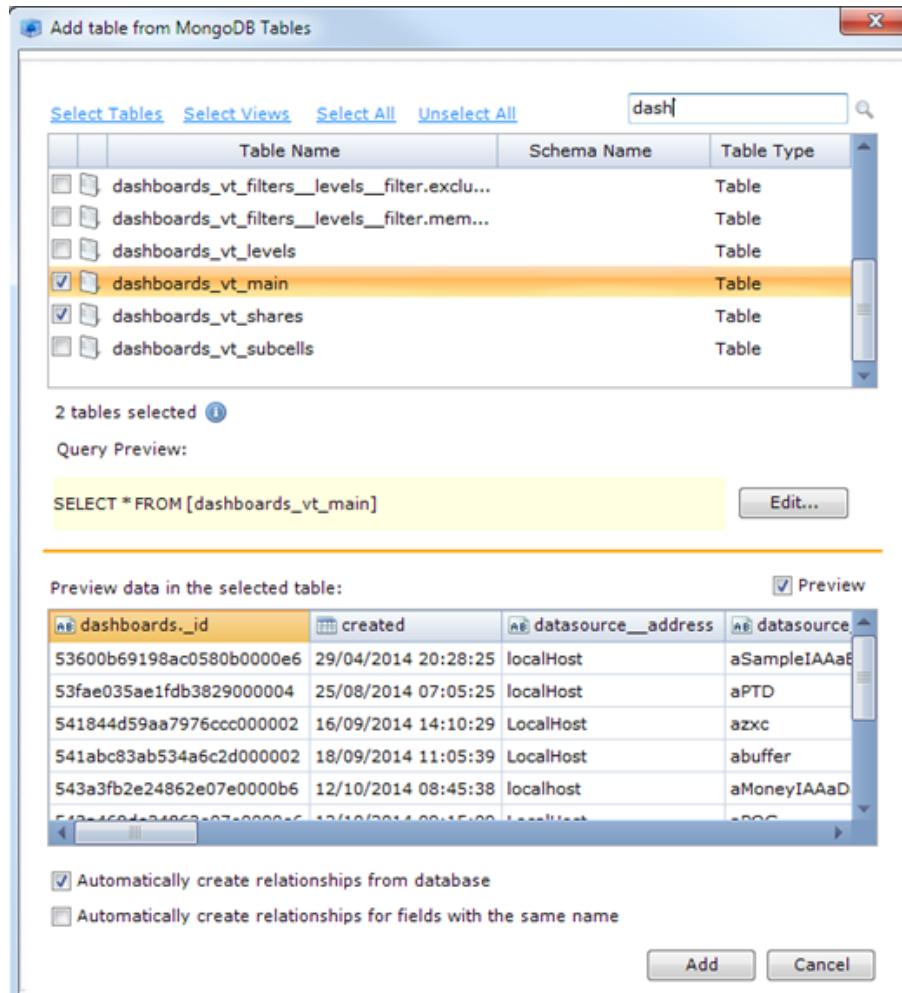
- When done, click **Test Connection**, and **OK**.

Selecting Tables in the ElastiCube Manager

After setting up the DSN, you can preview and customize the query (similar to other ODBC connections) of the imported MongoDB table.

In the ElastiCube Manager You can view the SQL syntax in the **Query Preview** section. Click **Edit** to customize it.

Select the tables created by the Sisense MongoDB ODBC Driver (tables including MAINTABLE_vt_FIELD), as opposed to the table that includes all the raw data (with suffix *main*).

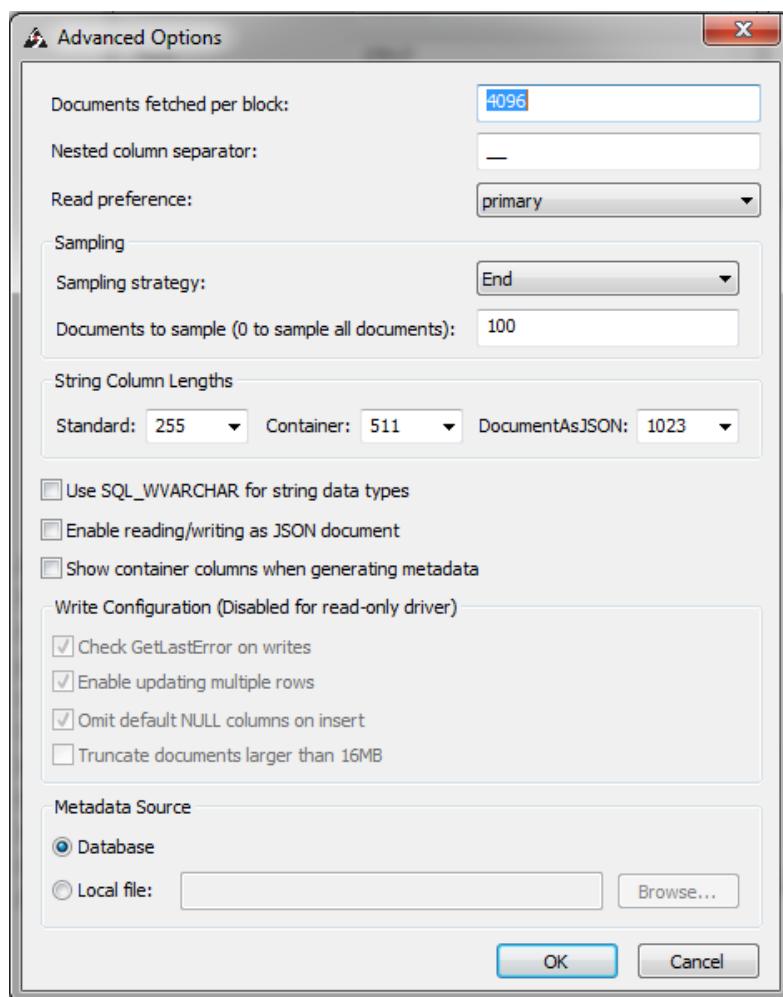


DSN Advanced Options

Click **Advanced Options** if you require additional settings:

- ▶ **Document fetched per block:** Maximum number of documents (rows) that the query returns each time.
- ▶ **Nested column separator:** Characters/strings used to separate names of nested columns.
- ▶ **Read preference:** Specifies how the driver routes read operations to the members of a replica set.

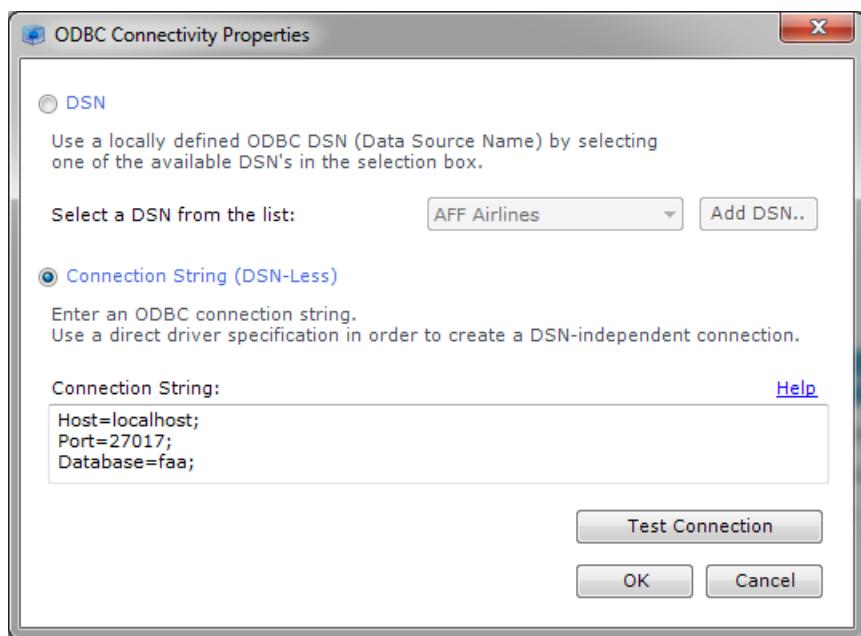
- ▶ **Sampling strategy:** Location in the table to sample rows from when detecting the schema (beginning, end or random position in table). *End* is recommended.
- ▶ **Documents to sample:** Number of documents to sample when detecting schema for a table (0 scans all documents).
- ▶ **Use SQL_WVARCHAR for string data types:** Select this option for the driver to map the MongoDB String data type to SQL_WVARCHAR instead of SQL_VARCHAR.



Connecting to Sisense MongoDB ODBC Driver using a Connection String

If you have a driver already defined, then you can connect to Sisense MongoDB ODBC Driver using strings (without using a DSN).

1. Select **Connection String (DSN-Less)**.



2. Type in the connection details as strings, as follows:

Example with minimum required attributes

```
Driver=Sisense MongoDB ODBC Driver;
Host=localhost;
Port=27017;
Database=aff;
```

Example with with advanced options:

Driver=Sisense MongoDB ODBC Driver

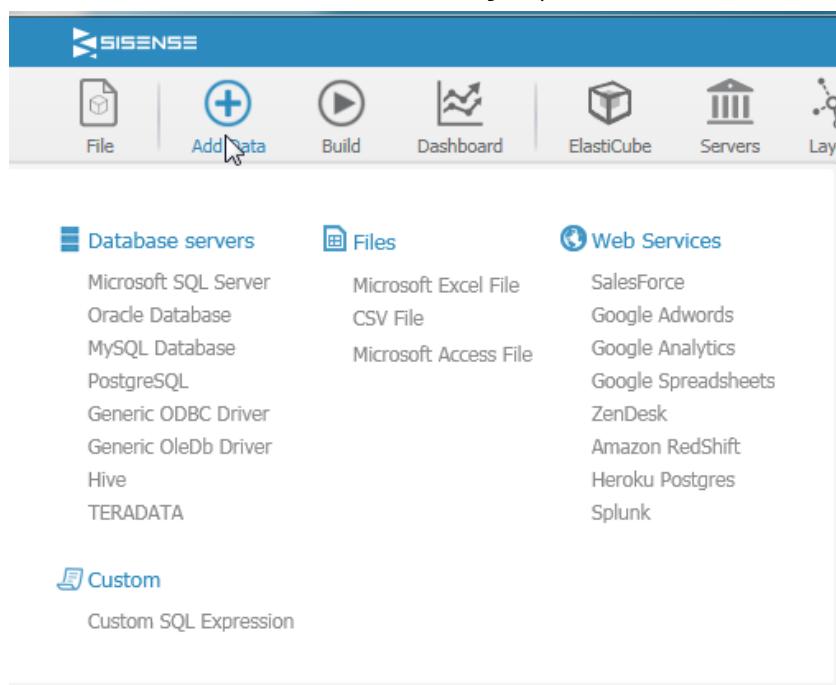
```
Host=192.168.100.100;
Port=27017;
Database=MyDatabase;
UID=MyUsername;
PWD=MyPassword;
RowsFetchedPerBlock=4096;
DefaultStringColumnLength=255;
DefaultContainerColumnLength=511;
UseSqlWVarchar=0;
CacheMetadata=1; VirtualTableDetection=1
```

3. When done, click **Test Connection**, and **OK**.

Connecting to MySQL

The ElastiCube Manager enables easy and quick access to databases, tables and views contained with MySQL databases. The steps below detail how to connect to this type of data source.

1. Click **Add data** on the top menu of the ElastiCube Manager.
2. Under **Database servers**, select **MySQL Database**.



The Connect to MySql window is displayed.

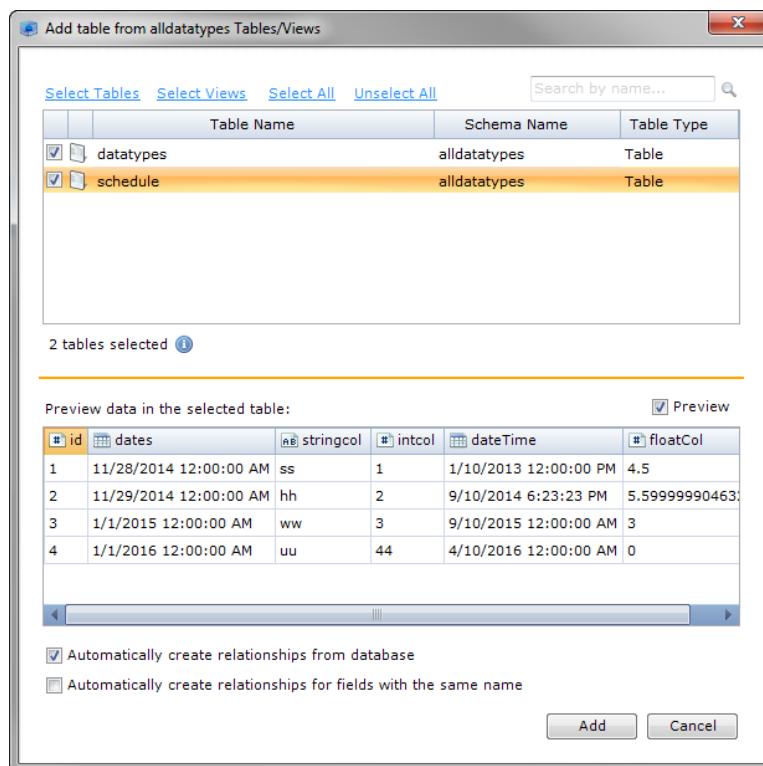
3. **Database server location:** Enter the computer/server IP address of the database. To connect to a database running on your own computer enter **localhost**.
4. Select either **Windows Authentication** if configured with the database or alternatively, **Use the Following User Name & Password**, and enter the database credentials.
5. (Optional) Select **Use SSL** to if you are connecting to an SSL server. Enter the relevant information to configure verification between the client and the MySql server over SSL:

- ▶ In the **Private Key** field, enter the path or click  and navigate to the location of your file containing the your private key.
- ▶ In the **Certificate** field, enter the path or click  and navigate to the location of your PEM file containing the client's certificate.
- ▶ In the **Authority** field, enter the path or click  and navigate to the location of your PEM file containing the trusted SSL certificate authority.

Alternatively, you can use Sisense's ODBC connectors to connect to secure data sources.

6. Click **Connect to server**.
7. From the **Select Database** drop-down list, select the relevant database you want to work with, and click **OK**

All tables and views associated with the database will appear in a new window.



To view a preview of data contained in a particular table, highlight the table or view in the list and click the preview pane below. To preview the table, select the **Preview** checkbox. Enable the checkbox next to each table or view you would like to use.

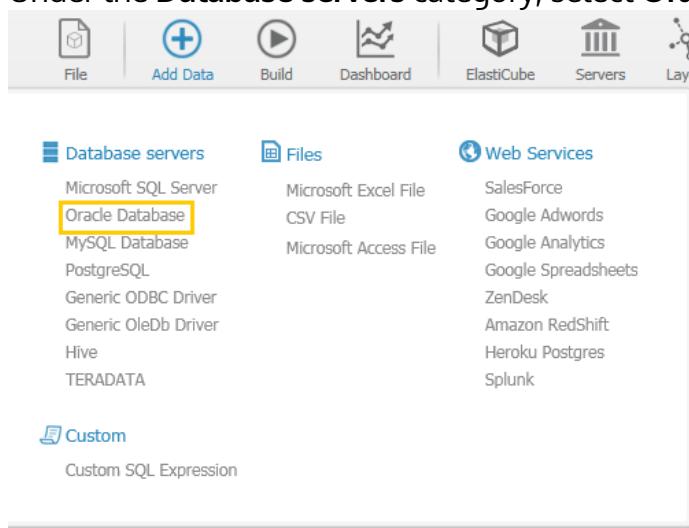
Existing relationships between tables can be automatically replicated in the ElastiCube by selecting the **Automatically create relationships from database** option. Likewise fields with similar names can be linked by selecting the **Automatically create relationships for fields with the same name** option.

8. If you want to customize the data before importing it into the ElastiCube, you can run a custom SQL query to manipulate the data. This can be useful, for example, when you want to import only a portion of the data, rather than all of the data.
9. Once all relevant tables are selected, click **Add**.

Connecting to Oracle

The ElastiCube Manager enables easy and quick access to databases, tables and views contained within Oracle databases. The steps below describe how to connect to this type of data source.

1. Click **Add data** in the top menu of the ElastiCube Manager.
2. Under the **Database servers** category, select **Oracle Database**.



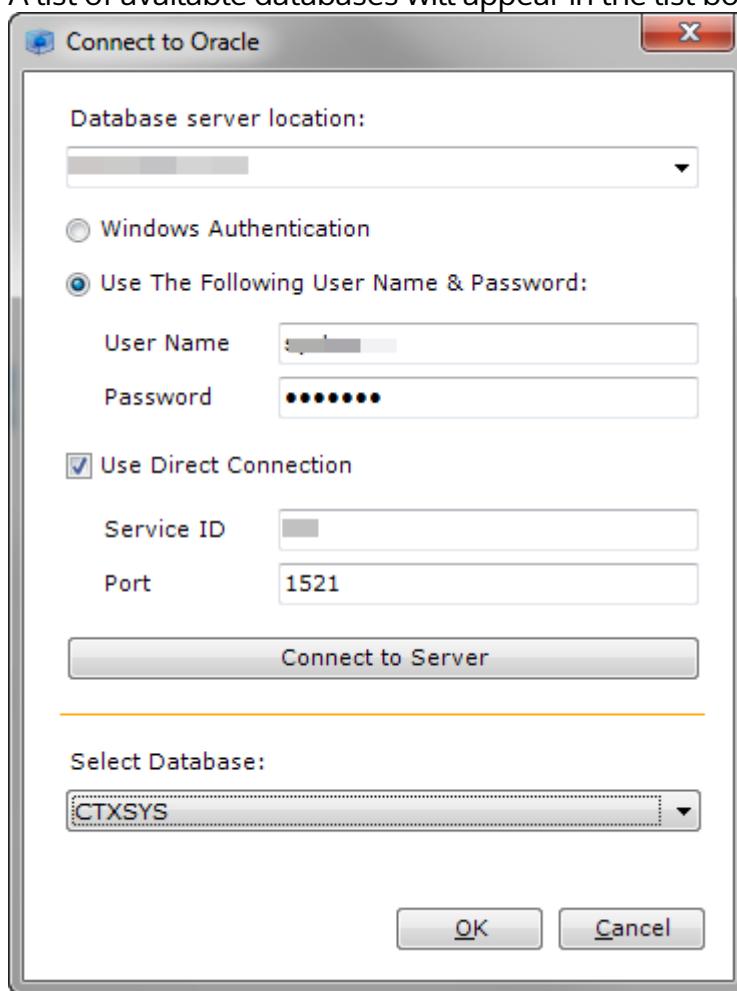
3. You are prompted to enter the following information:
 - ▶ **Database server location:** Enter the computer/server IP address with the database. To connect to a database running on your own computer, type in *localhost*.
 - ▶ **Login details:** Either use your Windows login details if they are configured with the database, or alternatively enter the username and password to connect to the database.
 - ▶ **Use Direct Connection:** You can also use a direct connection to connect. You will need to enter the **Service ID** and **Port** number to connect.
 - As long as you choose to work in direct mode to Oracle (this can be defined in the Oracle provider wizard), no client software needs to be installed. If you want to work with an already installed Oracle client, you can use the

indirect mode, which assumes you have the Oracle client already installed.

- Service ID can be found in the TNSNAMES.ora file. You can find a full description in the following topic: [Service ID](#)
- Note that in the file for each tnsname, you can find the service ID and service name.

4. Click **Connect to Server**.

5. A list of available databases will appear in the list box below.

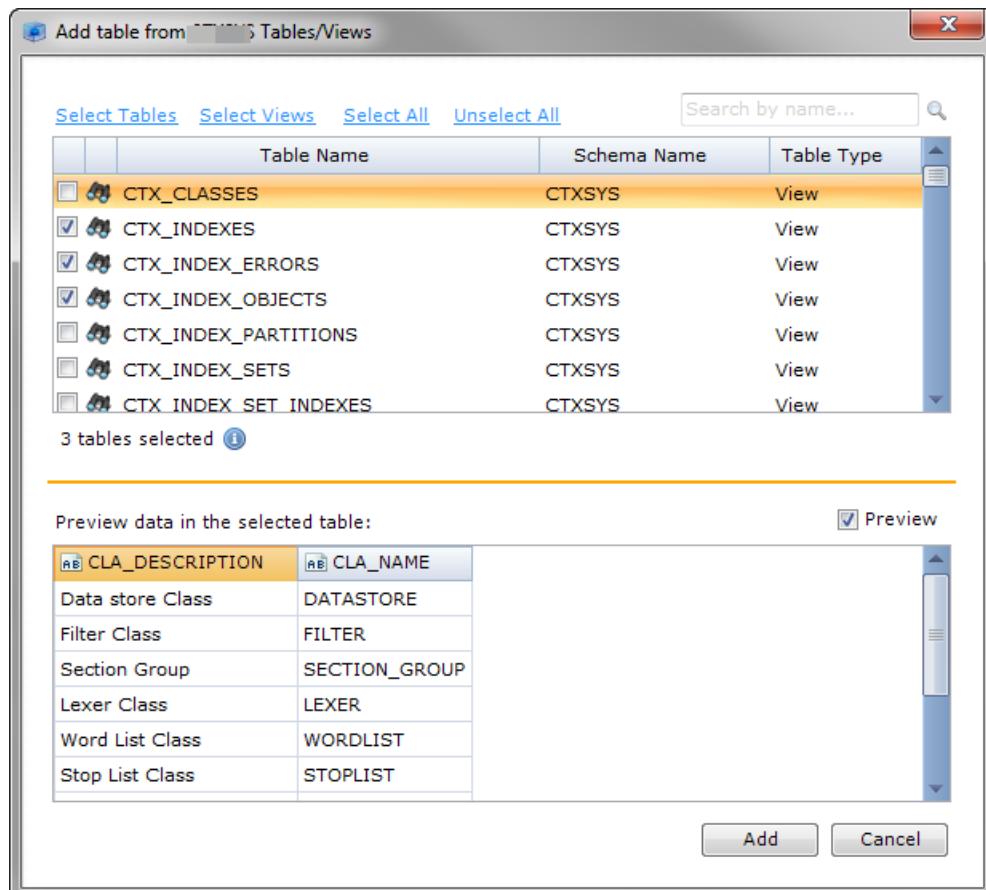


6. Select the relevant database you want to work with and click **OK**.

All tables and views associated with the database will appear in a new window.

To view a preview of data contained in a particular table, highlight the

table or view in the list and click the preview pane below. To preview the table, select the **Preview** checkbox.

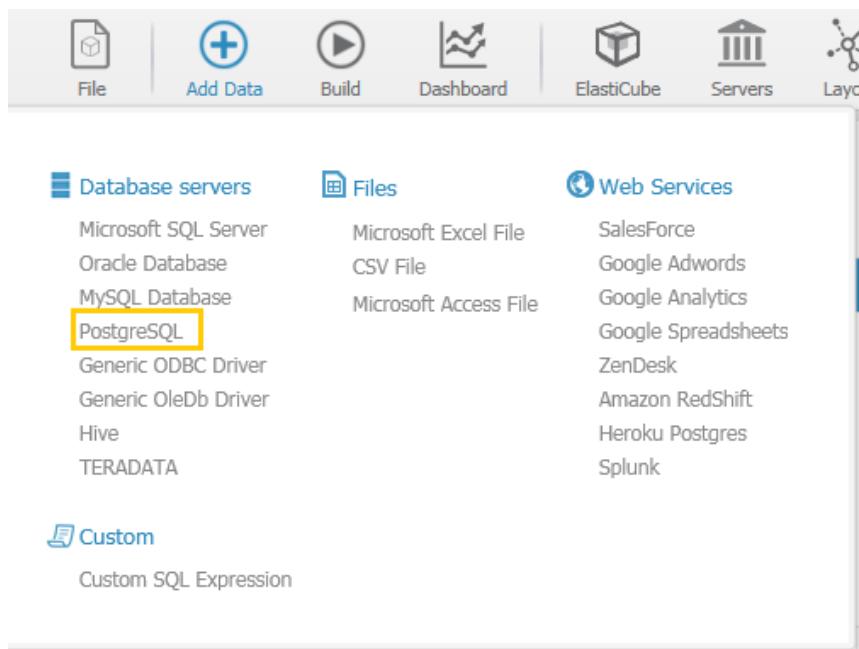


7. Select the check box next to each table or view you want to use.
8. If you want to customize the data before importing it into the ElastiCube, you can run a custom SQL query to manipulate the data. This can be useful, for example, when you want to import only a portion of the data, rather than all of the data. [Click here](#) to learn more.
9. Once all relevant tables are selected, click **Add**.

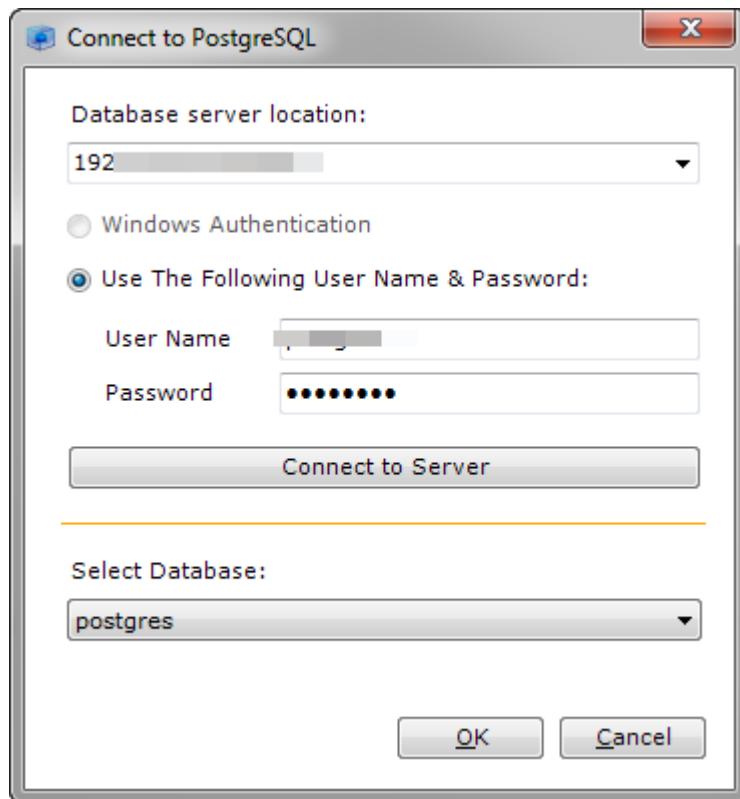
Connecting to PostgreSQL

The ElastiCube Manager enables easy and quick access to databases, tables and views contained within PostgreSQL databases. The steps below describe how to connect to this type of data source.

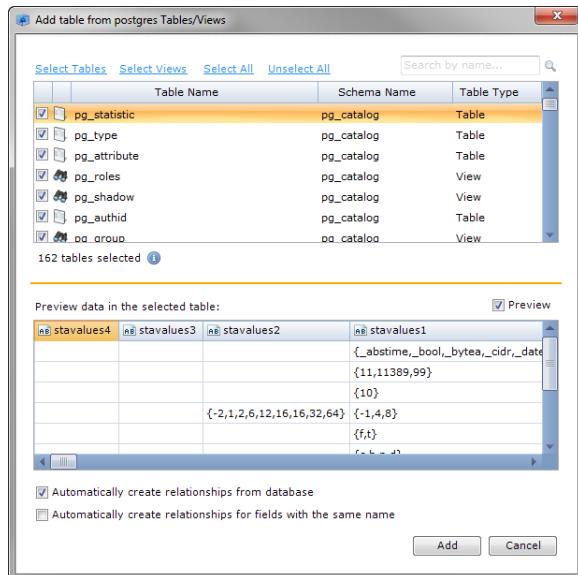
1. Click **Add Data** in the top menu of the ElastiCube Manager.
2. Under the **Database servers** category, select **PostgreSQL**.



3. You will be prompted to enter the following information:
 - ▶ **Database server location:** Enter the computer/server IP address where the database is located. To connect to a database on running on your own computer enter **localhost**.
 - ▶ **Login details:** Either use your Windows login details if they are configured with the database, or alternatively enter the username and password to connect to the database.



4. Click **Connect to Server**.
5. A list of available databases will appear in the drop-down list below.
6. Select the relevant database you want to work with and click **OK**.
7. All tables and views associated with the database will appear in a new window.



To view a preview of data contained in a particular table, highlight the table or view in the list and click the preview pane below. To preview the table, select the **Preview** checkbox.

8. Select the checkboxes next to each table or view you want to use.

Existing relationships between tables can be automatically replicated in the ElastiCube by selecting the **Automatically create relationships from database** checkbox. Likewise fields with similar names can be linked by selecting **Automatically create relationships for fields with the same name**.

9. If you want to customize the data before importing it into the ElastiCube, you can run a custom SQL query to manipulate the data. This can be useful, for example, when you want to import only a portion of the data, rather than all of the data.
10. Once all relevant tables are selected, click **Add**.

Connecting to Quickbooks Desktop

The Sisense Quickbooks Desktop connector is a standalone connector that allows you to import data from Quickbooks Desktop's API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the Quickbooks Desktop APIs. To obtain a connection string, you will need to create a Quickbooks Desktop app.

Once you have connected to Quickbooks Desktop, you can import a variety of tables from the Quickbooks Desktop API.

This section describes how to install the Quickbooks Desktop connector, how to connect to Quickbooks Desktop with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the Quickbooks Desktop Connector

Sisense provides the Quickbooks Desktop connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Quickbooks Desktop connector:

1. [Download](#) the Quickbooks Desktop installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.

The Quickbooks Desktop connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

Connecting to the Quickbooks Desktop REST API

The Quickbooks Desktop connector makes requests to QuickBooks through the Remote Connector. The Remote Connector is an easy-to-use tool that enables developers to access QuickBooks data remotely. The Remote Connector runs on the same machine as QuickBooks and accepts connections through a lightweight, embedded Web server.

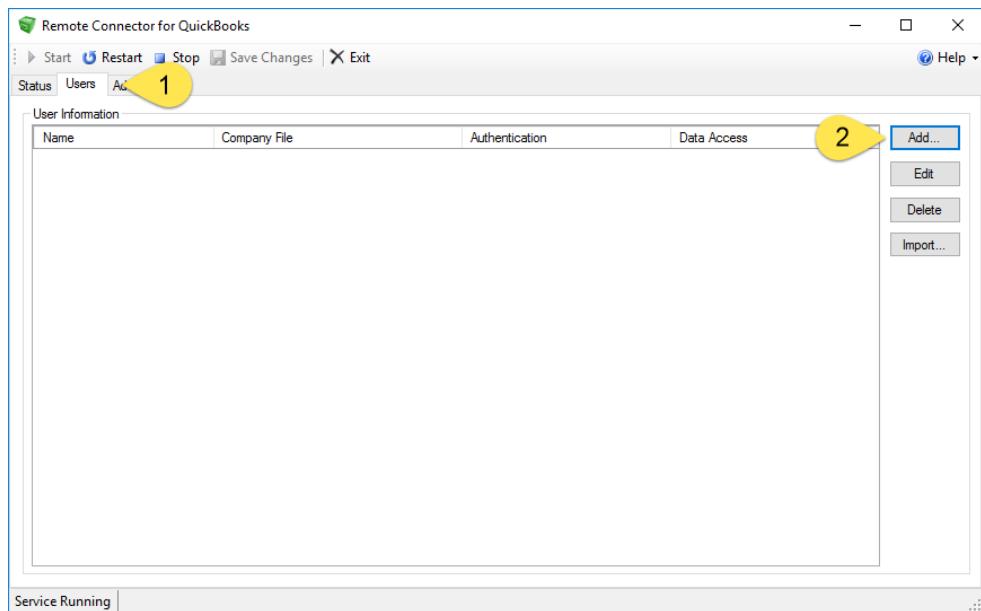
The server supports SSL/TLS, enabling users to connect securely from remote machines. The first time you connect, you will need to authorize the driver with QuickBooks.

The Remote Connector can be used to read and write to QuickBooks in situations where direct COM access to QuickBooks is not available (e.g., ASP.NET, Java, or QuickBooks on a remote machine).

In the Remote Connector, you can define users and their passwords and then use these credentials in the connection string that you create to connect to your QuickBooks Desktop tables.

Follow the procedure below to connect to QuickBooks for the first time through the Remote Connector:

1. Download the Remote Connector
from remoteconnector.com and install the Remote Connector on the machine where QuickBooks is installed.
2. Open the company file you want to connect to in QuickBooks using an administrator account in single-user mode.
3. Open the Remote Connector from the system tray and add a user on the Users tab. Enter a User and Password and select the level of access in the Data Access menu.



Note: The Remote Connector does not use the User and Password properties to access QuickBooks; the User and Password properties authenticate the user to the Remote Connector. Authentication to QuickBooks is handled based on the Application Name property.

4. When you first connect, a dialog will appear in QuickBooks prompting you to authorize the application. After authorizing the application, you can then execute commands to QuickBooks. Specify the URL of the Remote Connector and the User and Password. By default, the Remote Connector connects to the currently open company file.
5. If you want to access QuickBooks when QuickBooks is not running, save the company file information for the user. The Remote Connector will then automatically open QuickBooks in the background with the company file for that user.

Note: If the QuickBooks UI is open, you can only connect to that company file. Additionally, note that the user permissions you run the Remote Connector under must match the user permissions you run

QuickBooks under. The Remote Connector installation process installs the Remote Connector as a service under the current user account.

Adding Quickbooks Desktop Tables to your ElastiCube

Sisense uses connection strings to connect to Quickbooks Desktop and import data into the ElastiCube Manager.

The connection string to connect to Quickbooks Desktop has the following structure:

`jdbc:Quickbooks Desktop:Property1=Value1;Property2=Value2;`

The following is an example of a Quickbooks Desktop connection string:

`jdbc:quickbooks:User=xxxxx;Password=xxxxxxxxxx;URL=http://xxxx`
`xxxxxxxx;`

- ▶ **User:** Set this to username defined in Remote Connector.
- ▶ **Password:** Set this to password defined in Remote Connector.
- ▶ **URL:** The IP address of the remote machine where Remote Connector was installed and the port. The port is defined under Connection Settings in the **Advanced** tab of the Remote Connector.
- ▶ **OAuthSettingsLocation:** The location of the settings file where OAuth values are saved for each user when InitiateOAuth is set to GETANDREFRESH. See Switching between Accounts for more information.

Note: To switch between accounts, you need to delete the file OAuthsettings.txt located.

at

.\\Users\\xxx\\AppData\\Roaming\\CData\\QuickbooksDesktop\\DataProvider.

To add Quickbooks Desktop data:

1. In ElastiCube Manager, click **Add Data** and then, **Quickbooks Desktop**. The Connect to Quickbooks Desktop window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Quickbooks Desktop is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Quickbooks Desktop and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.
6. The tables are displayed in the ElastiCube Manager.

Switching between Accounts

When you connect to the Quickbooks Desktop data source, Sisense saves your OAuth values in the file OAuthsettings.txt file located at .\\Users\\xxx\\AppData\\Roaming\\CData\\Quickbooks Desktop Data Provider on your Sisense server. To connect to the Quickbooks Desktop data source with another user on the same machine, you must delete the OAuthsettings.txt file. Sisense will then generate a new file for that user.

Another option to support multiple users is to define the location and file name of an OAuthsettings file for each unique user in your connection string through the **OAuthSettingsLocation** parameter.

When each user connects to the data source, Sisense generates the OAuth file with the file name you specify in the location you define. In the examples below, two users are allowed to access the Quickbooks

Desktop data source and for each user, Sisense generates a file that contains that user's OAuth values in the location defined in the string.

```
jdbc:QuickbooksDesktop:OAuthSettingsLocation=C:\QuickbooksDes  

ktop\auth\john.txt;OAuthClientId=11276856774486;
```

```
OAuthClientSecret=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhost/;
```

```
jdbc:QuickbooksDesktop:OAuthSettingsLocation=C:\QuickbooksDes  

ktop\auth\sally.txt;OAuthClientId=11276856774486;
```

```
OAuthClientSecret=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhost/;
```

In the example above, two OAuth files are created, one for John and one for Sally in the location C:\QuickbooksDesktop\auth\.

This is useful if you support many users who each need to access the Quickbooks Desktop data source.

Quickbooks Desktop Tables

Quickbooks Desktop's RESTful APIs expose the following Quickbooks Desktop tables that you can import into the ElastiCube Manager through the Sisense Quickbooks Desktop connector:

Available Tables

Name	Description
Accounts	Create, update, delete, and query QuickBooks Accounts. To update Accounts, set the QBXMLVersion to 6.0 or higher.
BillExpenseItems	Create, update, delete, and query QuickBooks Bill Expense Line Items.

Name	Description
BillLineItems	Create, update, delete, and query QuickBooks Bill Line Items.
BillPaymentChecks	Create, update, delete, and query QuickBooks Bill Payment Checks. QBXMLVersion must be set to 6.0 or higher to update a BillPaymentCheck.
BillPaymentChecksAppliedTo	Create, update, delete, and query QuickBooks Bill Payment AppliedTo aggregates. In a Bill Payment, each AppliedTo aggregate represents the Bill transaction to which this part of the payment is being applied.
BillPaymentCreditCards	Create, update, delete, and query QuickBooks Bill Payments.
BillPaymentCreditCardsAppliedTo	Create, update, delete, and query QuickBooks Bill Payment AppliedTo aggregates. In a Bill Payment, each AppliedTo aggregate represents the Bill transaction to which this part of the payment is being applied.
Bills	Create, update, delete, and query QuickBooks Bills.
BuildAssemblies	Insert, Update, delete, and query QuickBooks Build Assembly transactions.
BuildAssemblyLineItems	Create and query QuickBooks Build Assembly transactions.
CheckExpenseItems	Create, update, delete, and query QuickBooks Check Expense Line Items.
Checks	Create, update, delete, and query QuickBooks Check Line Items.
Class	Create, update, delete, and query QuickBooks Classes. QuickBooks requires QBXML version 8.0 or higher for updates to a Class.

Name	Description
CreditCardChargeExpenseItems	Create, update, delete, and query QuickBooks Credit Card Charge Expense Line Items.
CreditCardChargeLineItems	Create, update, delete, and query QuickBooks Credit Card Charge Line Items.
CreditCardCharges	Create, update, delete, and query QuickBooks Credit Card Charges.
CreditCardCreditExpenseItems	Create, update, delete, and query QuickBooks Credit Card Credit Expense Line Items.
CreditCardCreditLineItems	Create, update, delete, and query QuickBooks Credit Card Credit Line Items.
CreditCardCredits	Create, update, delete, and query QuickBooks Credit Card Credits.
CreditMemoLineItems	Create, update, delete, and query QuickBooks Credit Memo Line Items.
CreditMemos	Create, update, delete, and query QuickBooks Credit Memos.
Currency	Create, update, delete, and query QuickBooks Currencies. This table requires QBXML version 8.0 or higher, and you will need to enable multiple Currencies in your QuickBooks company file to use it.
CustomerContacts	Create, update, delete, and query QuickBooks Customer Contacts. This table requires QBXML version 12.0 or higher, and is only available in QuickBooks editions 2016 and above.
CustomerMessages	Create, delete, and query Customer Messages.

Name	Description
CustomerNotes	Create, update, and query QuickBooks Customer Notes. This table requires QBXML version 12.0 or higher.
Customers	Create, update, delete, and query QuickBooks Customers.
CustomerShippingAddresses	Create, update, delete, and query QuickBooks Customer Shipping Addresses. Multiple Customer Shipping Addresses are supported in only QuickBooks 2013 and higher. Additionally, this table requires QBXML version 12.0 or higher.
CustomerTypes	Create, update, delete, and query QuickBooks Customer Types.
DateDrivenTerms	Create, delete, and query QuickBooks Date Driven Terms.
DepositLineItems	Create, update, delete, and query QuickBooks Deposit Line Items. QBXMLVersion must be set to 7.0 or higher to update a deposit.
Deposits	Create, update, delete, and query QuickBooks Deposits. QBXMLVersion must be set to 7.0 or higher to update a deposit.
EmployeeEarnings	Create, update, delete, and query QuickBooks Employee Earnings.
Employees	Create, update, delete, and query QuickBooks Employees.
EstimateLineItems	Create, update, delete, and query QuickBooks Estimate Line Items.
Estimates	Create, update, delete, and query QuickBooks Estimates.
InventoryAdjustmentLineItems	Create and query QuickBooks Inventory Adjustment Line Items.
InventoryAdjustments	Create, query, and delete QuickBooks Inventory Adjustments.

Name	Description
InventorySites	Create, update, delete, and query QuickBooks Inventory Sites. Inventory Sites are only available in QuickBooks Enterprise 2010 and above, and only with the Advanced Inventory add-on.
InvoiceLineItems	Create, update, delete, and query QuickBooks Invoice Line Items.
Invoices	Create, update, delete, and query QuickBooks Invoices.
ItemLineItems	Create, update, delete, and query QuickBooks Item Line Items.
ItemReceiptExpenseItems	Create, update, delete, and query QuickBooks Item Receipt Expense Line Items.
ItemReceiptLineItems	Create, update, delete, and query QuickBooks Item Receipt Line Items.
ItemReceipts	Create, update, delete, and query QuickBooks Item Receipts.
Items	Create, update, delete, and query QuickBooks Items.
JobTypes	Create and query QuickBooks JobTypes.
JournalEntries	Create, update, delete, and query QuickBooks Journal Entries. Note that while Journal Entry Lines can be created with a new Journal Entry, they cannot be added or removed from an existing Journal Entry.
JournalEntryLines	Create, update, delete, and query QuickBooks Journal Entries. Note that while Journal Entry Lines can be created with a new Journal Entry, they cannot be added or removed from an existing Journal Entry.
OtherNames	Create, update, delete, and query QuickBooks Other Name entities.

Name	Description
PaymentMethods	Create, update, delete, and query QuickBooks Payment Methods.
PayrollNonWageItems	Query QuickBooks Non-Wage Payroll Items.
PayrollWageItems	Create and query QuickBooks Wage Payroll Items.
PriceLevelPerItem	Create and query QuickBooks Price Levels Per Item. Only QuickBooks Premier and Enterprise support Per-Item Price Levels. Note that while Price Levels can be added from this table, you may only add Per-Item Price Levels from this table. Price Levels may be deleted from the PriceLevels table. This table requires QBXML version 4.0 or later.
PriceLevels	Create, delete, and query QuickBooks Price Levels. Note that while Price Levels can be added and deleted from this table, you may add only fixed-percentage Price Levels from this table. Per-Item Price Levels may be added via the PriceLevelPerItem table. This table requires QBXML version 4.0 or higher.
PurchaseOrderLineItems	Create, update, delete, and query QuickBooks Purchase Order Line Items.
PurchaseOrders	Create, update, delete, and query QuickBooks Purchase Orders.
ReceivePayments	Create, update, delete, and query QuickBooks Receive Payment transactions. QBXMLVersion must be set to 6.0 or higher to update a ReceivePayment.
ReceivePaymentsAppliedTo	Create, update, and query QuickBooks Receive Payment AppliedTo aggregates. In a Receive Payment, each AppliedTo aggregate represents the transaction to which this part of the payment is being applied. QBXMLVersion must be set to 6.0 or higher to update a ReceivePayment.

Name	Description
SalesOrderLineItems	Create, update, delete, and query QuickBooks Sales Order Line Items.
SalesOrders	Create, update, delete, and query QuickBooks Sales Orders.
SalesReceiptLineItems	Create, update, delete, and query QuickBooks Sales Receipt Line Items.
SalesReceipts	Create, update, delete, and query QuickBooks Sales Receipts.
SalesReps	Create, update, delete, and query QuickBooks Sales Rep entities.
SalesTaxCodes	Create, update, delete, and query QuickBooks Sales Tax Codes.
SalesTaxItems	Create, update, delete, and query QuickBooks Sales Tax Items.
ShippingMethods	Create, update, delete, and query QuickBooks Shipping Methods.
StandardTerms	Create, update, delete, and query QuickBooks Standard Terms.
StatementCharges	Create, update, delete, and query QuickBooks Statement Charges.
TimeTracking	Create, update, delete, and query QuickBooks Time Tracking events.
ToDo	Create, update, delete, and query QuickBooks To Do entries.
TransferInventory	Query and delete QuickBooks Transfer Inventory transactions. Transfer Inventory is available in only QuickBooks Enterprise 2010 and above, and only with the Advanced Inventory add-on.

Name	Description
TransferInventoryLineItems	Create and query QuickBooks Transfer Inventory Line Items. Transfer Inventory is available in only QuickBooks Enterprise 2010 and above, and only with the Advanced Inventory add-on.
Transfers	Create, update, and query QuickBooks transfers. Requires QBXML version 12.0 or higher.
UnitOfMeasure	Create, update, delete, and query QuickBooks units of measure. QuickBooks requires QBXML version 7.0 or higher to use this table.
UnitOfMeasureDefaultUnits	Create and query QuickBooks unit-of-measure default units. QuickBooks requires QBXML version 7.0 or higher to use this table.
UnitOfMeasureRelatedUnits	Create and query QuickBooks unit-of-measure related units. QuickBooks requires QBXML version 7.0 or higher to use this table.
VehicleMileage	Create, update, delete, and query QuickBooks Vehicle Mileage entities. QuickBooks requires QBXML version 6.0 or higher to use this table.
VendorCreditExpenseItems	Create, update, delete, and query QuickBooks Vendor Credit Expense Line Items.
VendorCreditLineItems	Create, update, delete, and query QuickBooks Vendor Credit Line Items.
VendorCredits	Create, update, delete, and query QuickBooks Vendor Credits.
Vendors	Create, update, delete, and query QuickBooks Vendors.
VendorTypes	Create, update, delete, and query QuickBooks Vendor Types.

Name	Description
WorkersCompCodeLines	Query QuickBooks Workers Comp Code entries by line. Requires QBXML Version 7.0 or higher.
WorkersCompCodes	Query QuickBooks Workers Comp Code entries. Requires QBXML Version 7.0 or higher.

Available Views

Name	Description
BillLinkedTransactions	Query QuickBooks Bill Linked Transactions.
CompanyInfo	Query the company information from QuickBooks.
CreditMemoLinkedTransactions	Query QuickBooks Credit Memo Linked Transactions.
DeletedEntities	Query deleted Entities.
DeletedTransactions	Query deleted Transactions.
EstimateLinkedTransactions	Query QuickBooks Estimate Linked transactions.
Host	Query the QuickBooks host process. The Host represents information about the QuickBooks process currently being executed.
InvoiceLinkedTransactions	Query QuickBooks Invoice Linked Transactions.
ItemReceiptLinkedTransactions	Query QuickBooks Item Receipt Linked Transactions.
ItemSites	Create, update, delete, and query QuickBooks Item Sites. Item sites are only available in QuickBooks Enterprise 2010 and above, and only with the Advanced Inventory add-on. This table requires a minimum of QBXML version 10.0.

Name	Description
Preferences	Query information about many of the preferences the QuickBooks user has set in the company file.
PurchaseOrderLinkedTransactions	Query QuickBooks Purchase Order Linked Transactions.
SalesOrderLinkedTransactions	Query QuickBooks Sales Order Linked Transactions.
StatementChargeLinkedTransactions	Query QuickBooks Statement Charge Linked Transactions.
Templates	Query QuickBooks templates.
Transactions	Query QuickBooks transactions. You may search the Transactions using a number of values including Type, Entity, Account, ReferenceNumber, Item, Class, Date, and TimeModified.
VendorCreditLinkedTransactions	Query QuickBooks Vendor Credit Linked Transactions.

Connecting to QuickBooks Online

The Sisense QuickBooks Online connector is a standalone connector that allows you to import data from the QuickBooks Online API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect to Quickbooks Online API through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the QuickBooks Online API. To obtain a connection string, you will need to create a QuickBooks Online app.

Once you have connected to QuickBooks Online, you can import a variety of tables from the QuickBooks Online API.

This section describes how to install the QuickBooks Online connector, how to connect to QuickBooks Online with a connection string, and what tables you can import into the ElastiCube Manager.

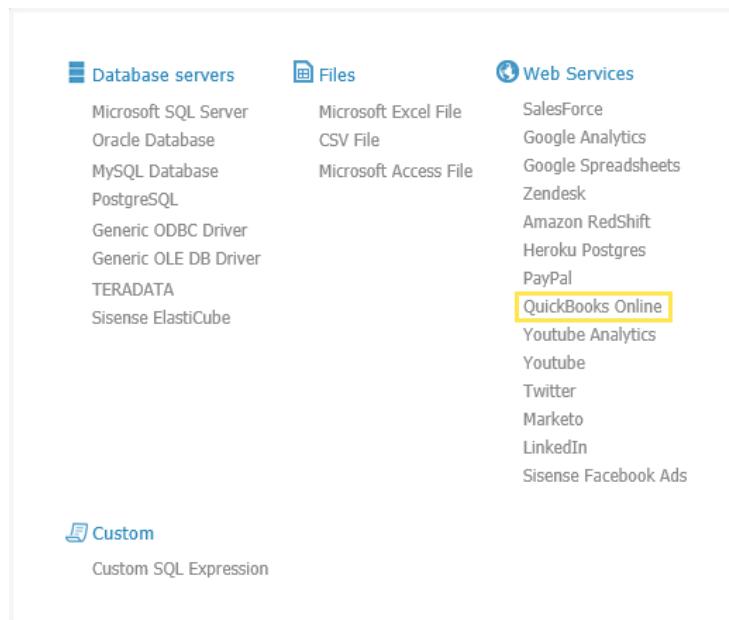
Installing the Quickbooks Online Connector

Sisense provides the Quickbooks Online connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the QuickBooks Online connector:

1. [Download](#) the Quickbooks Online installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.

The Quickbooks Online connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.



Connecting to the Quickbooks Online REST API

To access Quickbooks Online's REST API from Sisense, you must provide valid Oauth Quickbooks Online credentials. These credentials are provided by Quickbooks Online when you create an application.

Creating an App

You can follow the steps below to create an app and obtain the OAuth client credentials, the consumer key and consumer secret. You will need to create an Intuit developer account if you do not already have one.

1. Log in to <http://developer.intuit.com>.
2. Click **My Apps > Create New App > Select APIs**. Select the **Accounting API**.
3. On the Settings tab, define the Launch URL. If you are making a desktop application, set Launch URL to `http://localhost/`. For a Web application, set Launch URL to a page you would like the

user to be returned to after they have granted your application permissions.

Once you have created the app, you will be redirected to a page with information about your app. The **Development tab** contains your Sandbox credentials and the **Production tab** contains credentials for use with a production account. **The consumer key and consumer secret are displayed on the Keys tab.**

Authenticating through Quickbooks Online

After setting the following connection properties, you are ready to connect:

- ▶ **OAuthClientId:** Set this to the consumer key in your app settings.
- ▶ **OAuthClientSecret:** Set this to the consumer secret in your app settings.
- ▶ **CompanyId:** Set this to the ID of the company you want to connect to.
- ▶ **CallbackURL:** Set this to `http://localhost`.
- ▶ **InitiateOAuth:** Set this to `GETANDREFRESH`. You can use `InitiateOAuth` to avoid repeating the OAuth exchange and manually setting the `OAuthAccessToken` and `OAuthAccessTokenSecret`.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver then completes the OAuth process:

1. Extracts the access token from the callback URL and authenticates requests.

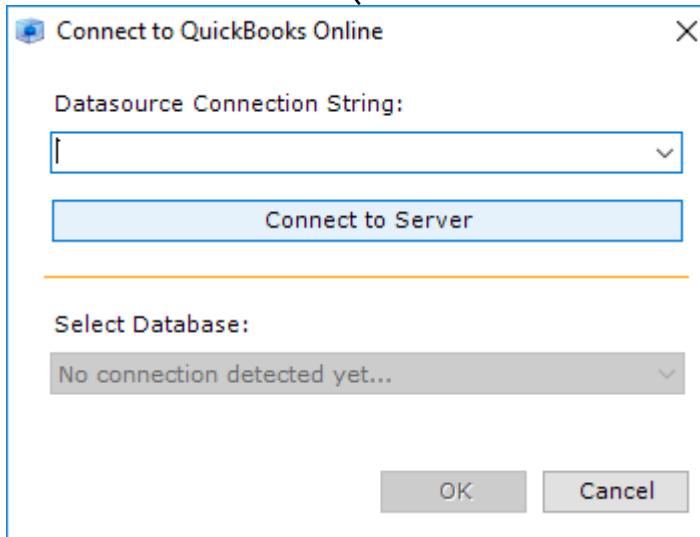
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Adding Quickbooks Online Tables to your ElastiCube

Sisense uses connection strings to connect to Quickbooks Online and import data into the ElastiCube Manager. Each connection string contains authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add Quickbooks Online data:

1. In ElastiCube Manager, click **Add Data** and then, **Quickbooks Online**. The Connect to Quickbooks Online window is displayed.



2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Quickbooks Online is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Quickbooks Online and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.

The tables are displayed in the ElastiCube Manager.

Quickbooks Online Tables

The Sisense Quickbooks Online connector allows you to import the following tables into the ElastiCube Manager.

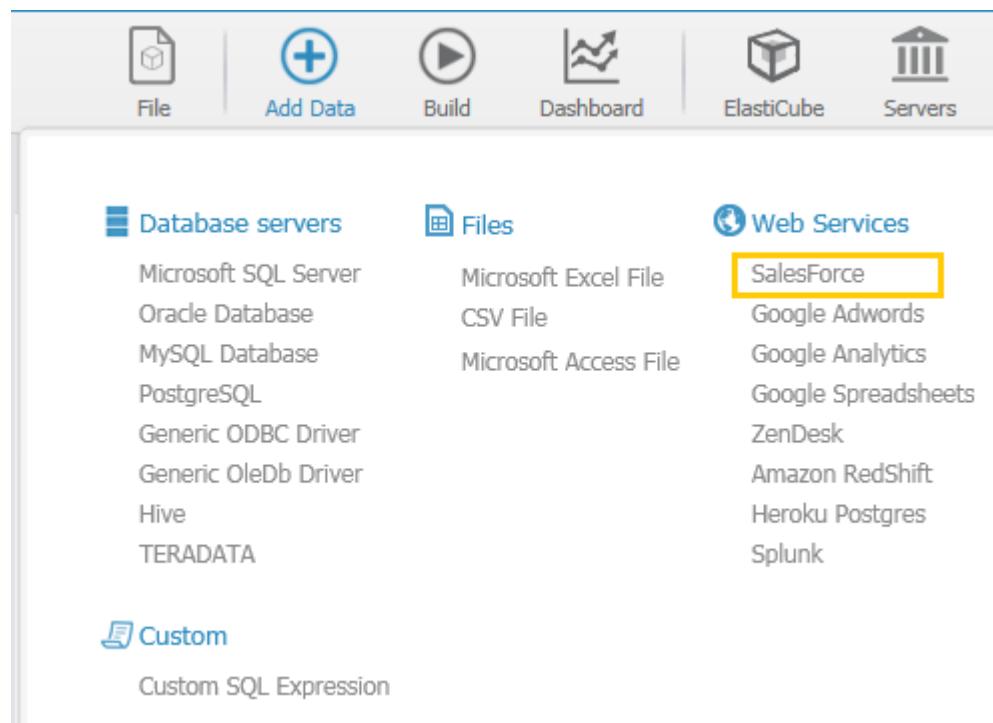
Name	Description
Accounts	Query QuickBooks Accounts.
Attachables	Attachables.
BillLineItems	Query QuickBooks Bill Line Items.
BillPaymentLineItems	Query QuickBooks Bill Payment Line Items.
BillPayments	Query QuickBooks Bill Payments.
Bills	Query QuickBooks Bills.
BudgetDetails	Query QuickBooks BudgetDetails.
Budgets	Query QuickBooks Budgets.
Class	Query QuickBooks Classes.
CompanyInfo	Retrieve information about the QuickBooks company.
CreditMemoLineItems	Query QuickBooks Credit Memo Line Items.
CreditMemos	Query QuickBooks Credit Memos.
Customers	Query QuickBooks Customers.
Departments	Query QuickBooks Departments.

Name	Description
DepositLineItems	Query QuickBooks Deposit Line Items.
Deposits	Query QuickBooks Deposits.
Employees	Query QuickBooks Employees.
Entitlements	Retrieves QuickBooks Entitlements.
EstimateLineItems	Query QuickBooks Estimate Line Items.
EstimateLinkedTransactions	Query QuickBooks Estimate Line Items.
Estimates	Query QuickBooks Estimates.
InvoiceLineItems	Query QuickBooks Invoice Line Items.
InvoiceLinkedTransactions	Query QuickBooks Invoice Line Items.
Invoices	Query QuickBooks Invoices.
Items	Query QuickBooks Items.
JournalEntries	Query QuickBooks Journal Entries.
JournalEntryLineItems	Query QuickBooks Journal Entry Line Items.
PaymentLineItems	Query QuickBooks Payment Line Items.
PaymentMethods	Query QuickBooks Payment Methods.
Payments	Query QuickBooks Payments.
Preferences	Query QuickBooks Preferences. The Preferences table contains settings for company-wide preferences, which affect all users.

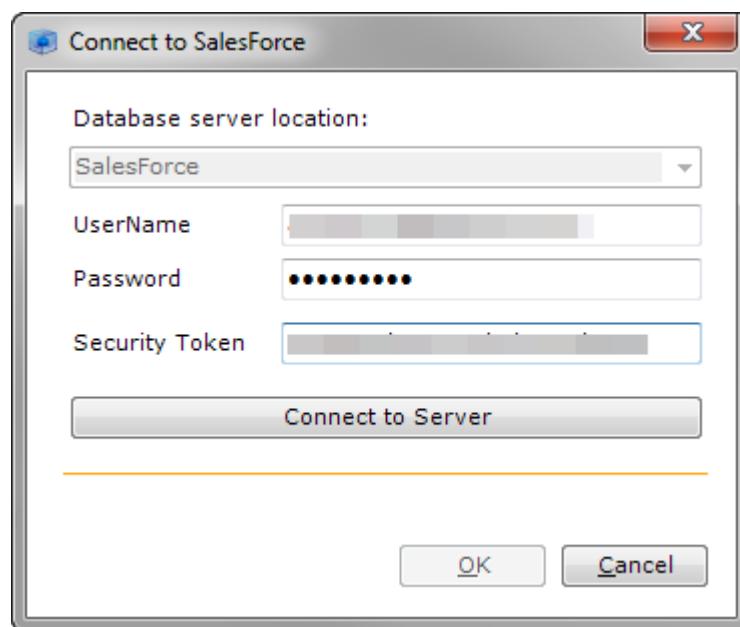
Name	Description
PurchaseLineItems	Query QuickBooks Purchase Line Items.
PurchaseOrderLineItems	Query QuickBooks Purchase Order Line Items.
PurchaseOrders	Query QuickBooks Purchase Orders.
Purchases	Query QuickBooks Purchases.
RefundReceiptLineItems	Query QuickBooks RefundReceiptLineItems.
RefundReceipts	Query QuickBooks RefundReceipts.
SalesReceiptLineItems	Query QuickBooks Sales Receipt Line Items.
SalesReceipts	Query QuickBooks Sales Receipts.
TaxCodes	Query QuickBooks Sales Tax Codes.
TaxRates	Query QuickBooks Tax Rates.
Terms	Query QuickBooks Terms.
TimeActivities	Query QuickBooks Time Activities.
Transfers	Query QuickBooks Transfers
VendorCredits	Query QuickBooks Vendor Credits.
Vendors	Query QuickBooks Vendors.

Connecting to SalesForce

1. The ElastiCube Manager enables easy and quick access to databases, tables and views contained within SalesForce. The steps below describe how to connect to this type of data source.
2. If you need to connect to your SalesForce Sandbox environment, read [this article](#) in our support forum.
3. **Note:** *Sisense only supports SalesForce Enterprise and above.* Before getting started you will need a *SalesForce Security Token*. To generate a token, open SalesForce and click Setup>'My Personal Information'>'Reset Security Token', and then click on the button labelled 'Reset Security Token'. The token will be sent to your email address.
4. **Before you begin**
5. SalesForce can be customized in various ways, therefore its recommend to first review your deployment, and understand in which entity (account, contact etc.) and how the required data is stored. Once understood, connect to SalesForce and import a few tables to review the data. For a general outline of SalesForce entities see [SalesForce Schema](#) and [Table Description](#).
6. Click **Add Data** in the top menu of the ElastiCube Manager.
7. Under the **Web Services** category, select **SalesForce**.

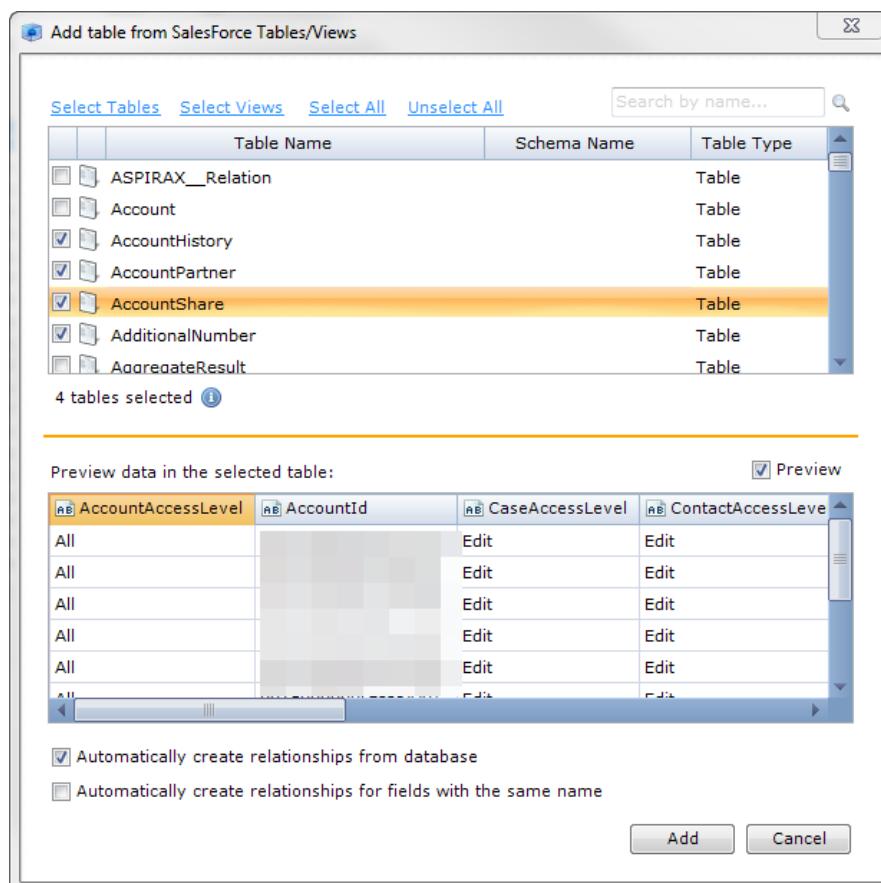


8. Enter the username and password to connect to SalesForce.
9. Enter the security token requested from SalesForce (see note above).



10. Click **Connect to Server**, then click **OK**.

11. All tables and views associated with SalesForce will appear in a new window.
12. To preview the data contained in a particular table, highlight the table or view in the list and click the preview pane below. To preview the table, select the Preview checkbox.
13. Select the checkboxes next to each table or view you want to use.
14. Existing relationships between tables can be automatically replicated in the ElastiCube by selecting the **Automatically create relationships from database** checkbox. Likewise fields with similar names can be linked by selecting the **Automatically create relationships for fields with the same name** checkbox.





SISENSE

15. Once all relevant tables are selected, click **Add**.

Connecting to ServiceNow

The Sisense ServiceNow connector is a standalone connector that allows you to import data from ServiceNow's API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the ServiceNow APIs. To obtain a connection string, you will need to register a ServiceNow app.

Once you have connected to ServiceNow, you can import a variety of tables from the ServiceNow API.

This section describes how to install the ServiceNow connector, how to connect to ServiceNow with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the ServiceNow Connector

Sisense provides the ServiceNow connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the ServiceNow connector:

1. [Download](#) the ServiceNow installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The ServiceNow connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

Connecting to the ServiceNow REST API

To access ServiceNow's REST API from the ElastiCube Manager, you must provide valid Oauth ServiceNow credentials through a connection string. These credentials are provided by ServiceNow when you create an application.

After you receive your credentials from ServiceNow, you can create the connection string and provide Sisense with it to connect to your data.

Registering an App

Follow the steps below to obtain the OAuth client credentials, the OAuthClientId and OAuthClientSecret:

1. In the Navigator, click **System OAuth> Application Registry**.
2. Click **New** and then click **Create** an OAuth API Endpoint for External Clients.
3. Select the **Active** check box and enter the details for your app.

Authenticating through ServiceNow

After setting the following properties, you are ready to connect by providing the following details in a connection string:

- ▶ **InitiateOAuth:** Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken connection property.
- ▶ **OAuthClientId:** Set this to the Client Id in your app settings.
- ▶ **OAuthClientSecret:** Set this to the Client Secret in your app settings.
- ▶ **Password:** Set this to your password.

- ▶ **Username:** Set this to your username.
- ▶ **Instance:** Set this to your instance.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver completes the OAuth process.

Adding ServiceNow Tables to your ElastiCube

Sisense uses connection strings to connect to ServiceNow and import data into the ElastiCube Manager. Each connection string contains authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add ServiceNow data:

1. In ElastiCube Manager, click **Add Data** and then, **ServiceNow**.
The Connect to ServiceNow window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. ServiceNow is displayed in the Select Database list.
4. Click **OK**. Sisense connects to ServiceNow and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**. The tables are displayed in the ElastiCube Manager.

ServiceNow Tables

ServiceNow's RESTful APIs expose the following ServiceNow tables that you can import into the ElastiCube Manager through the Sisense ServiceNow connector:

Available Tables

Name	Description
ast_contract	The ServiceNow table ast_contract.
ast_license_base	The ServiceNow table ast_license_base.
change_request	The ServiceNow table change_request.
cmdb_ci	The ServiceNow table cmdb_ci.
cmdb_metric	The ServiceNow table cmdb_metric.
cmn_building	The ServiceNow table cmn_building.
cmn_context_help	The ServiceNow table cmn_context_help.
cmn_cost_center	The ServiceNow table cmn_cost_center.
cmn_department	The ServiceNow table cmn_department.
cmn_location	The ServiceNow table cmn_location.
cmn_map_page	The ServiceNow table cmn_map_page.
cmn_notif_device	The ServiceNow table cmn_notif_device.
cmn_notif_device_variable	The ServiceNow table cmn_notif_device_variable.
cmn_notif_grmember	The ServiceNow table cmn_notif_group.
cmn_notif_group	The ServiceNow table cmn_notif_message.
cmn_notif_message	The ServiceNow table cmn_notif_message.
cmn_notif_service_provider	The ServiceNow table cmn_notif_service_provider.

Name	Description
cmn_other_schedule	The ServiceNow table cmn_other_schedule.
cmn_relative_duration	The ServiceNow table cmn_relative_duration.
cmn_schedule	The ServiceNow table cmn_schedule.
cmn_schedule_blackout	The ServiceNow table cmn_schedule_blackout.
cmn_schedule_condition	The ServiceNow table cmn_schedule_condition.
cmn_schedule_maintenance	The ServiceNow table cmn_schedule_maintenance.
cmn_schedule_page	The ServiceNow table cmn_schedule_page.
cmn_schedule_span	The ServiceNow table cmn_schedule_span.
cmn_timeline_page	The ServiceNow table cmn_timeline_page.
cmn_timeline_page_style	The ServiceNow table cmn_timeline_page_style.
cmn_timeline_sub_item	The ServiceNow table cmn_timeline_sub_item.
diagrammer_action	The ServiceNow table diagrammer_action.
expert_panel	The ServiceNow table expert_panel.
item_option_new	The ServiceNow table item_option_new.
question	The ServiceNow table question.
sc_category	The ServiceNow table sc_category.
sc_cat_item	The ServiceNow table sc_cat_item.
sla	The ServiceNow table sla.

Name	Description
sysauto	The ServiceNow table sysauto.
sysauto_script	The ServiceNow table sysauto_script.
syslog	The ServiceNow table syslog.
sysrule	The ServiceNow table sysrule.
system_db_object	The ServiceNow table system_db_object.
system_dictionary	The ServiceNow table system_dictionary.
system_documentation	The ServiceNow table system_documentation.
system_import_set_row	The ServiceNow table system_import_set_row.
system_script_client	The ServiceNow table system_script_client.
system_ui_policy	The ServiceNow table system_ui_policy.
system_ui_policy_action	The ServiceNow table system_ui_policy_action.
task	The ServiceNow table task.
v_field_creator	The ServiceNow table v_field_creator.

Connecting to Sisense ElastiCubes

ElastiCubes can also be connected to a data source using the Sisense ElastiCube connector or the Sisense ODBC, allowing you to import tables across different ElastiCubes. This can be useful for several use cases, such as:

- ▶ Separating custom SQL logic that is used for heavy data transformation from the main ElastiCube for easier management and shorter build times.
- ▶ Combining tables that are based on different refresh schedules. As every ElastiCube has its own build schedule, it is possible to import a table from an ElastiCube that doesn't refresh as often.

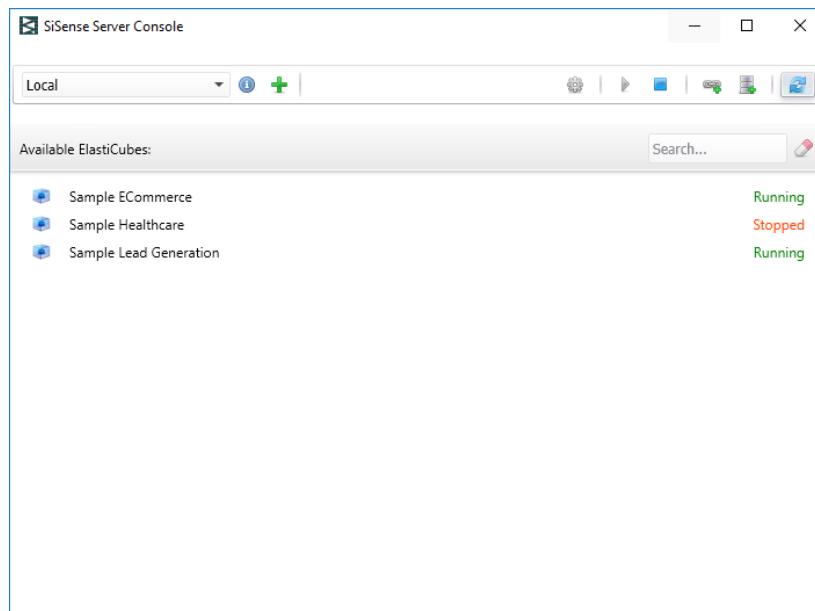
This section describes how to connect to your Sisense ElastiCubes through both methods:

- ▶ Sisense ElastiCube Connector
- ▶ ODBC Driver

Sisense ElastiCube Connector

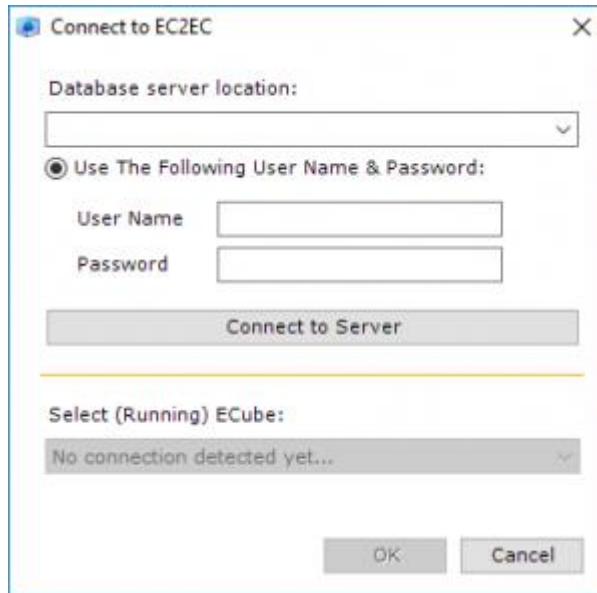
The native Sisense ElastiCube Connector enables you to import data from other running ElastiCubes and provides faster performance over the Sisense ODBC. The Sisense ElastiCube Connector does not support SSL connections, however, you can import ElastiCube through SSL connections via the ODBC driver described below.

Note: The first time you connect to an ElastiCube, the ElastiCube must be running. For subsequent connections, the connector automatically activates the ElastiCube. You can verify and activate your ElastiCubes from the Sisense Server Console in the ElastiCube Manager Admin.



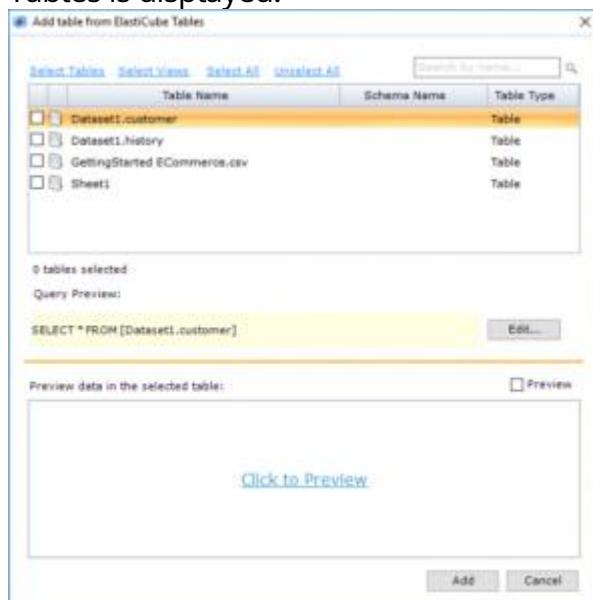
To import data from ElastiCubes:

1. In ElastiCube Manager, click **Add Data** and then, **Ecube Database**. The Connect to ElastiCube window is displayed.



2. From the Database server location list, enter the address of your ElastiCube.
OR
If you have previously entered the address, select the relevant address.

3. In the **User Name** and **Password** fields, enter your Sisense credentials for the ElastiCube.
4. Click **Connect to Server**. The running ElastiCubes at the location you defined are displayed in the Select (Running) ECube list.
5. Select the relevant ElastiCube. The Add table from ElastiCube Tables is displayed.



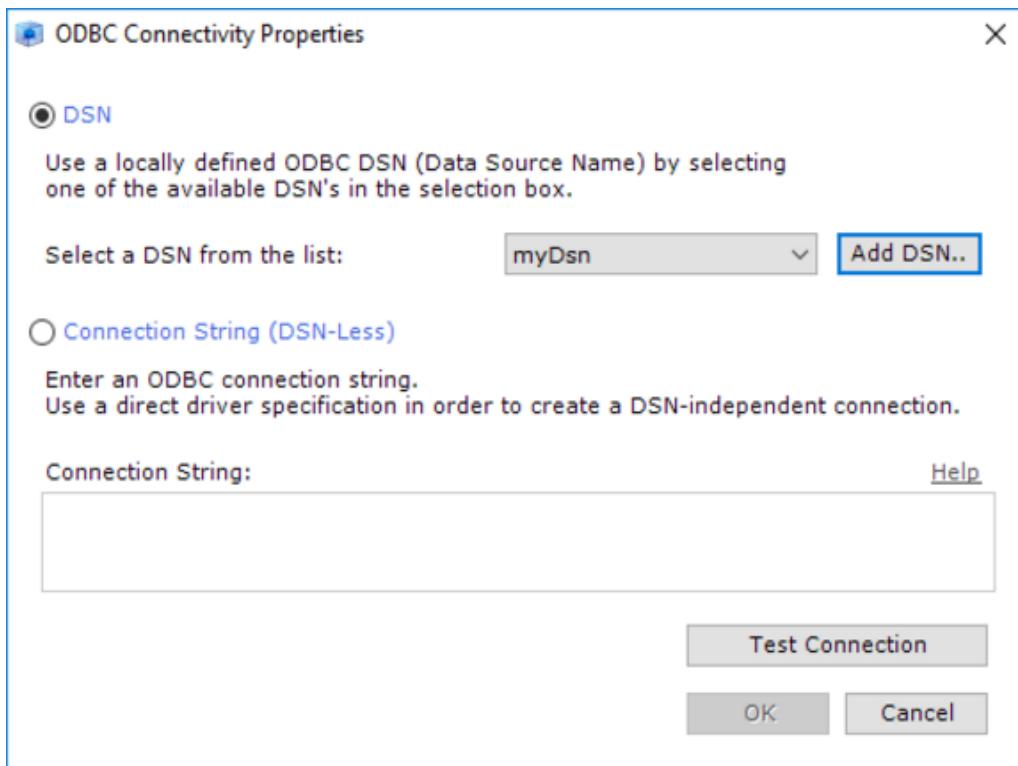
6. Select the relevant tables to import and click **Add**. The tables are displayed in the ElastiCube Manager.

Sisense ODBC Driver

The following topics explain how you can import ElastiCube data from another ElastiCube.

Downloading and Connecting the Sisense ODBC Driver

1. Download and install the Sisense ODBC Driver .
2. In ElastiCube Manager, click **Add Data** and then, **Generic ODBC Driver**.
3. Select DSN. If a DSN file has already been created, select it in the dropdown list, and click Test Connection. If you need to add a DSN, see Adding a DSN. If your connection has been set up, you can click OK and review the tables generated by the Sisense ODBC driver.



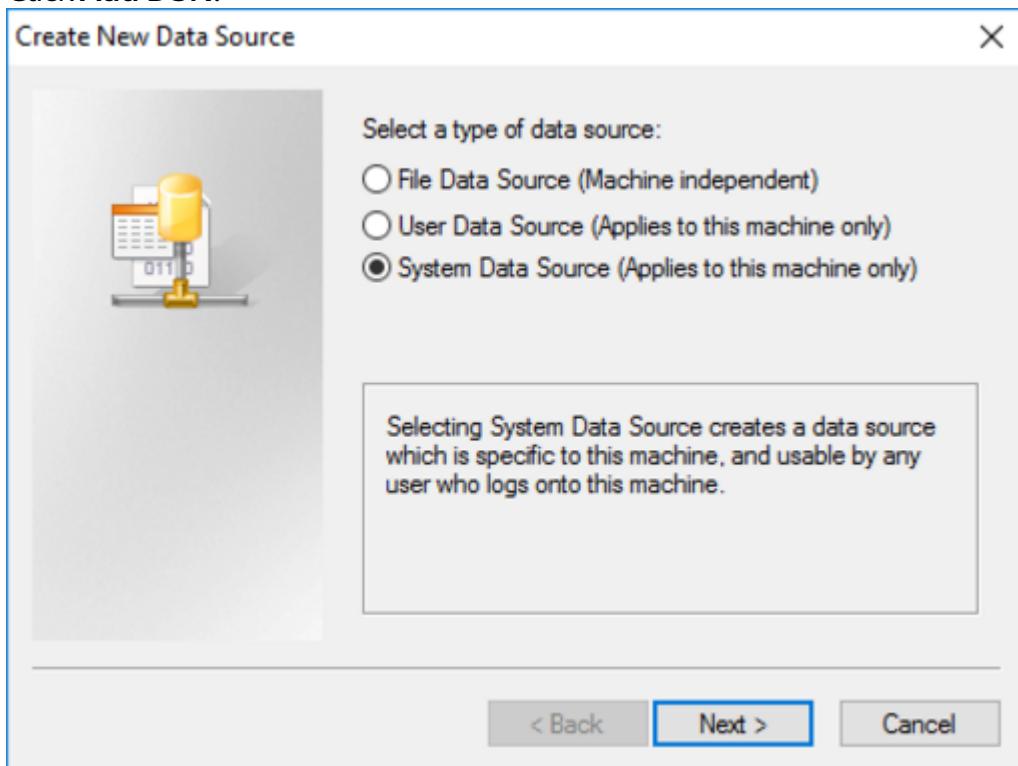
Adding a DSN

If a DSN has not yet been created, you must add the DSN manually.

Note: To add a DSN you must run the Sisense ElastiCube Manager as an administrator.

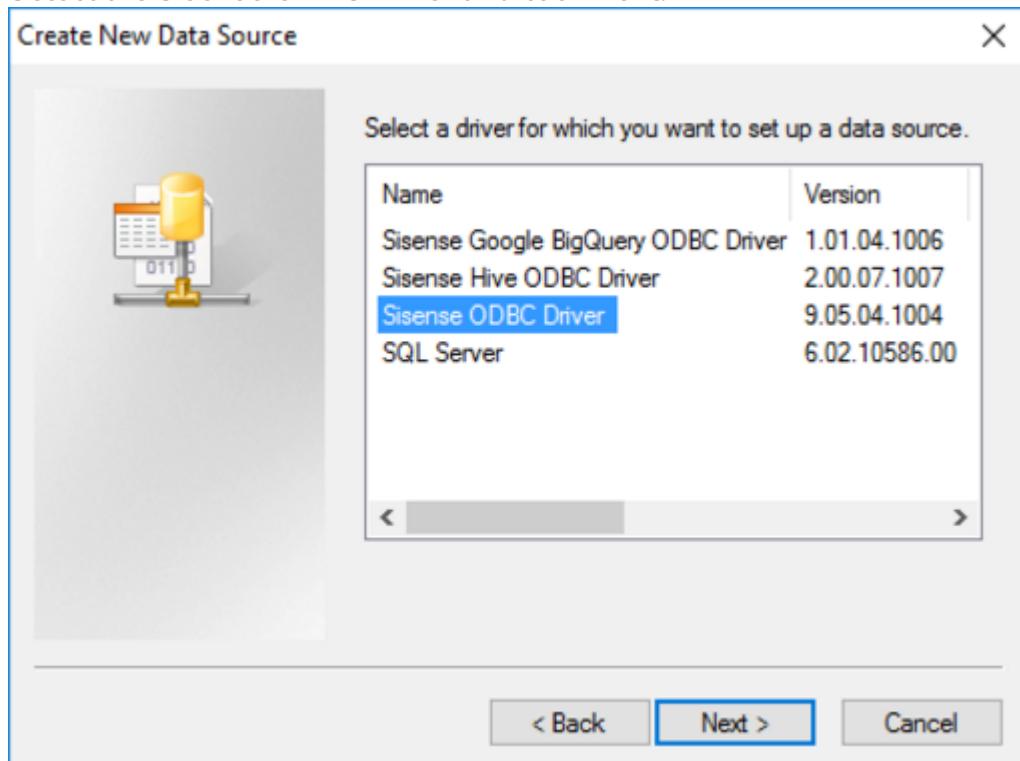
To add a DSN:

1. Click **Add DSN**.

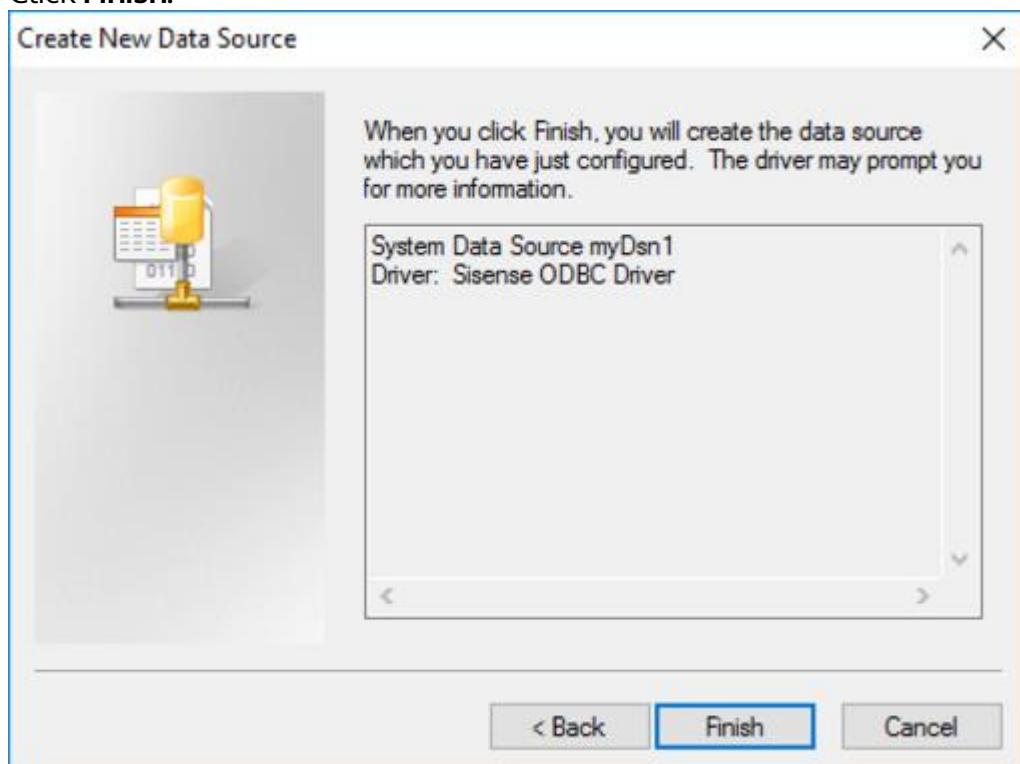


2. Select the System Data Source option. The created file will apply to all users in a specific machine only. Click **Next**.

3. Select the Sisense ODBC Driver and click **Next**.



4. Click **Finish**.



5. In the New Sisense Data Source window, do the following:

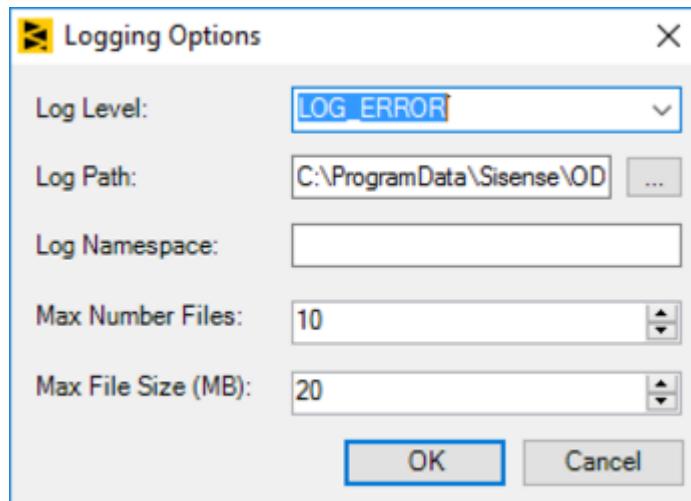
Field	Procedure
Data Source Name	Enter your name for the DSN that is to be displayed in the DSN dropdown list.
Description	Enter your description of the DSN.
Server	Enter the IP address of your ElastiCube server.
Use secure connection (SSL)	Select to enable a secure SSL connection.
ElastiCube	<p>Select the ElastiCube you want to export.</p> <p>OR</p> <p>If this is your first time configuring a DSN for the Sisense ODBC, click the ElastiCube list. The Login to Sisense window is displayed.</p> <p>Enter your credentials and click OK. Wait several seconds while Sisense returns a list of your ElastiCubes.</p> <p>After the list is displayed in the ElastiCube list, select the ElastiCube you want to import</p>
Log	(Optional) To configure logging behavior for the driver, click Logging Options . For more information, see Logging Options.

6. Click **Test**. A pop window is displayed that indicates the connection was successful.
7. Click **OK**.
8. In the ODBC Connectivity Properties window, click **Test Connection**.

9. Click **OK** to save and close the ODBC Connectivity Properties window.

Logging Options

You can configure logging options by clicking **Log** in the New Sisense Data Source window. This displays the Logging Options window.



The Sisense ODBC driver provides tracing functionality, which you can activate to help troubleshoot issues.

Important: Only enable logging long enough to capture an issue. Logging decreases performance and can consume a large quantity of disk space.

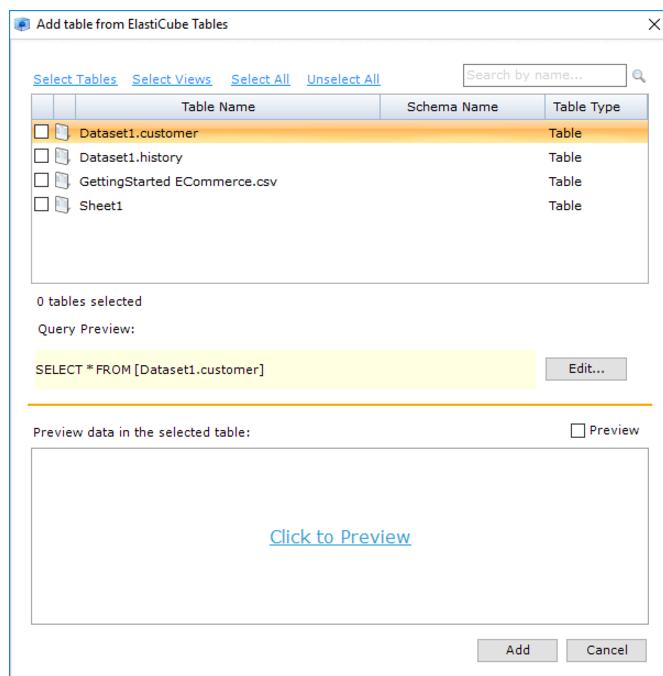
The driver allows you to set the amount of detail included in log files. The table below lists the logging levels provided by the Sisense ODBC Driver, in order from least verbose to most verbose.

In the Logging Options window, you can configure the following information:

Field	Description
Log Level	<p>Select the Log Level. There are seven possible options:</p> <p>LOG_OFF: Disables all logging.</p> <p>LOG_FATAL: Logs very severe error events that lead the driver to abort.</p> <p>LOG_ERROR: Logs error events that might still allow the driver to continue running.</p> <p>LOG_WARNING: Logs potentially harmful situations.</p> <p>LOG_INFO: Logs general information that describes the progress of the driver.</p> <p>LOG_DEBUG: Logs detailed information that is useful for debugging the driver.</p> <p>LOG_TRACE: Logs more detailed information than the DEBUG level.</p>
Log Path	<p>Enter the full path to the folder where you want to save log files.</p> <p>OR</p> <p>Click Browse and select the folder where you want to save log files.</p>
Log Namespace	Enter the filename for your log.
Max Number Files	Enter the maximum number of logs that can be created before the oldest log files are overwritten by newer logs.
Max File Size (MB)	Enter the maximum number of megabytes your log can reach before the log is overwritten by a newer log.

Adding ElastiCubes Tables to your Project

After setting up the DSN, the Add table from ElastiCube Tables window is displayed.



From this window, you add your ElastiCube tables to your ElastiCube Manager. In addition, you can view the SQL syntax in the Query Preview section and click **Edit** to customize it.

To add ElastiCube Tables to your Project:

1. Connect to your ElastiCube tables via the Sisense ODBC Tool.
2. Add your data source.
3. In the Add table from ElastiCube Tables window, select the table you want to add to the ElastiCube Manager.
4. Click **Add**. The selected tables are added to your ElastiCube Manager.

Connecting to Snowflake

Through the ElastiCube Manager, you can connect and integrate your Snowflake data warehouse quickly and easily to generate and analyze your data. Snowflake provides an ODBC driver for connecting to Snowflake using ODBC-based client applications such as Sisense.

Downloading and Connecting the Snowflake ODBC Driver

1. Download and install the Snowflake ODBC Driver. For more information, click [here](#).
2. In ElastiCube Manager, click **Add Data** and then, **Generic ODBC Driver**.
3. Select **DSN**. If a DSN file has already been created, select it in the dropdown list, and click **Test Connection**. If you need to add a DSN, see Adding a DSN. If your connection has been set up, you can click **OK** and review the tables generated by the Snowflake ODBC driver.



OR

Select Connection String (DSN-Less) and enter your Connection String. Click **Test Connection**. For more information, see Connecting without a DSN.

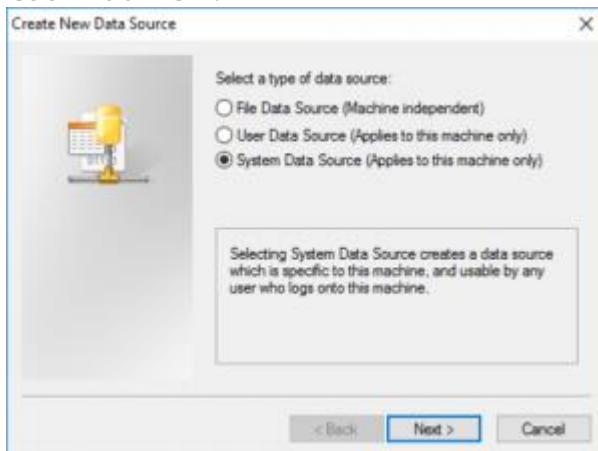
Adding a DSN

If a DSN has not yet been created, you must add the DSN manually.

Note: To add a DSN you must run the Sisense ElastiCube Manager as an administrator.

To add a DSN:

1. Click **Add DSN**.



2. Select the **System Data Source** option. The created file applies to all users in a specific machine only. Click **Next**.
3. Select the **Snowflake ODBC Driver**, and click **Next**.
4. Click **Finish**.

5. In the Snowflake ODBC Driver DSN Setup window, enter the following details:



Field	Description
User	Snowflake user name.
Password	Snowflake password.
Server	Specifies the domain name for your account provided to you by Snowflake.
Database	Specifies the default database to use for sessions initiated by the driver.
Schema	Specifies the default schema to use for sessions initiated by the driver.
Warehouse	Specifies the default warehouse to use for sessions initiated by the driver.
Role	Specifies the default role to use for sessions initiated by the driver. The specified role should be a role that has been assigned to the specified user for the driver. If the specified role has not been assigned to the user, the role is not used for sessions initiated by the driver.
Tracing	The level of detail to be logged in the driver trace files: 0 = Disable tracing

Field	Description
	1 = Fatal only error tracing
	2 = Error tracing
	3 = Warning tracing
	4 = Info tracing
	5 = Debug tracing
	6 = Detailed tracing

When entering parameters, note the following:

- ▶ Data Source, User and Server are the only parameters required to create a DSN.
- ▶ The Password field accepts a value, but does *not* store the value. This is a security precaution to ensure passwords are never stored directly in the driver.
- ▶ All other parameters in the dialog are optional.
- 6. Click **OK**.

Adding Snowflake Tables to your Project

After setting up the DSN, the Add table from Snowflake Tables window is displayed.

From this window, you add your Snowflake tables to your ElastiCube Manager. In addition, you can view the SQL syntax in the Query Preview section and click **Edit** to customize it.

To add Snowflake Tables to your Project:

1. Connect to Snowflake via the Sisense ODBC Tool.
2. Add your data source.

3. In the Add table from Snowflake Tables window, select the table you want to add to the ElastiCube Manager.
4. Click **Add**. The selected tables are added to your ElastiCube Manager.

Connecting to Splunk

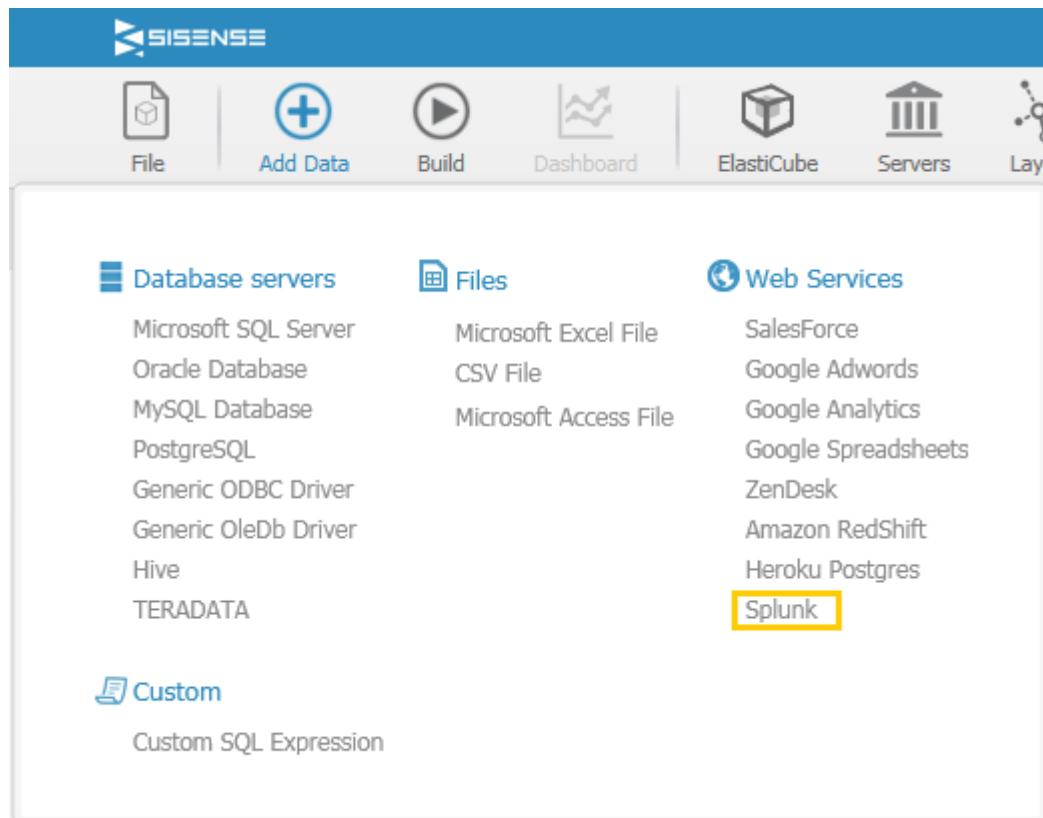
The Splunk SDK for C# v1.0.x is deprecated, and has been replaced by the Splunk SDK for C# 2.0. Unlike the Splunk SDK for C# 1.0.x, the Splunk SDK for C# 2.0 abides by .NET guidelines, as well as FxCop and StyleCop rules.

In addition, the API client in Splunk SDK for C# 2.0 is a Portable Class Library (PCL), and supports cross-platform development.

Version 2 of the SDK is incompatible with version 1, and applications built with Splunk SDK for C# version 1.0.x will not recompile using Splunk SDK for C# version 2.0. See [Migrating from Splunk SDK for C# v1.0.x](#) for more information.

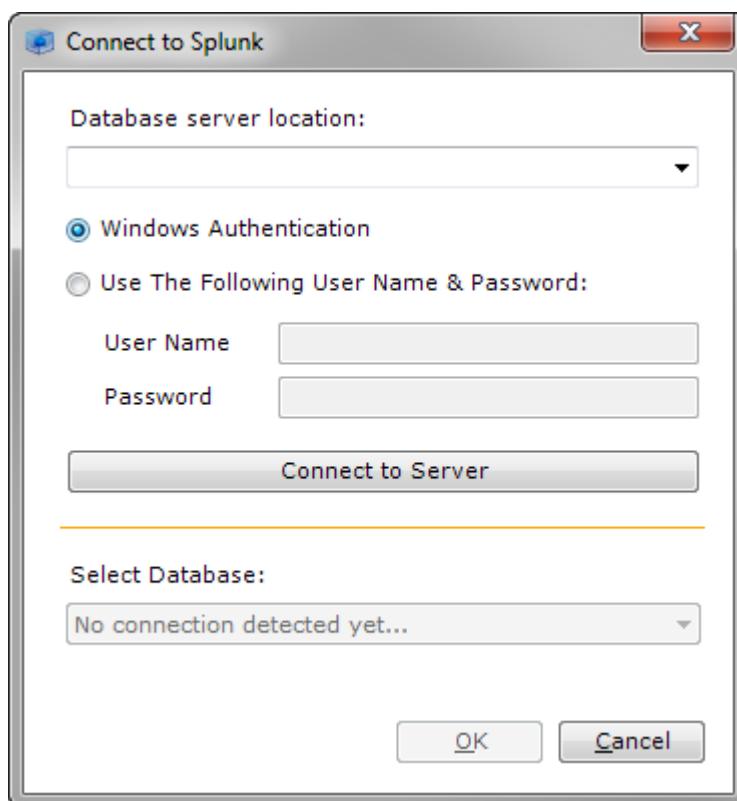
The ElastiCube Manager enables easy and quick access to Splunk. The steps below detail how to connect to this type of data source.

1. Click **Add data** in the top menu of the ElastiCube Manager.
2. Under the **Database Servers** category, select **Splunk**.



3. You will be prompted to enter the following information:

4. **Database server location:** Enter the computer/server IP address which holds the Splunk instance.
5. Select either **Windows Authentication** if configured with the database or alternatively, **Use the Following User Name & Password**, and enter the database credentials.



6. Click **Connect to Server**.
7. A list of available Splunk instances will appear in the list box below.
8. Select the relevant Splunk instance you want to work with and click **OK**.
9. All saved searches associated with the Splunk instance will appear in a new window.
10. To preview data contained in a particular Splunk search, highlight the search in the list and click in the **Preview** pane. To preview the search, select the **Preview** checkbox.

11. Select the checkbox next to each table or view you want to use.
12. Fields with similar names can be linked by selecting the **Automatically create relationships for fields with the same name** checkbox.
13. Once all relevant tables are selected, click **Add**.

Connecting to Stripe

The Sisense Stripe connector is a standalone connector that allows you to import data from Stripe's API into the ElastiCube Manager.

After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the Stripe APIs. To obtain a connection string, you will need to create a Stripe app.

Once you have connected to Stripe, you can import a variety of tables from the Stripe API.

This section describes how to install the Stripe connector, how to connect to Stripe with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the Stripe Connector

Sisense provides the Stripe connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Stripe connector:

1. [Download](#) the Stripe installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The Stripe connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

Connecting to the Stripe REST API

To access Stripe's REST API from the ElastiCube Manager, you must provide valid Oauth Stripe credentials through a connection string.

These credentials are provided by Stripe when you create an application.

After you receive your credentials from Stripe, you can create the connection string and provide Sisense with it to connect to your data.

Registering an App

Follow the steps below to obtain the OAuth client credentials, the OAuthClientId and OAuthClientSecret:

1. Log into your Stripe dashboard and click **Your Account > Account Settings > Connect > Platform Settings**.
2. Enter a name, description, and other information to be displayed to users when they log in to grant permissions to your app.
3. If you are making a desktop application, set the Redirect URL to <http://localhost:33333> or a different port number of your choice.

If you are making a Web application, set the Redirect URL to a page on your Web app you would like the user to be returned to after they have authorized your application.

Authenticating through Stripe

After setting the following properties, you are ready to connect:

- ▶ **OAuthClientId:** Set this to the client Id in the Connect section of your account settings.
- ▶ **OAuthClientSecret:** Set this to a secret key in the API Keys section of your account settings.

- ▶ **CallbackURL:** Set this to the Redirect URL in the Connect section of your account settings.
- ▶ **InitiateOAuth:** Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken connection property.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver completes the OAuth process.

1. Extracts the access token from the callback URL and authenticates requests.
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Adding Stripe Tables to your ElastiCube

Sisense uses connection strings to connect to Stripe and import data into the ElastiCube Manager. Each connection string contains a authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add Stripe data:

1. In ElastiCube Manager, click **Add Data** and then, **Stripe**. The Connect to Stripe window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Stripe is displayed in the Select Database list.

4. Click **OK**. Sisense connects to Stripe and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.
6. The tables are displayed in the ElastiCube Manager.

Stripe Tables

Stripe's RESTful APIs expose the following Stripe tables that you can import into the ElastiCube Manager through the Sisense Stripe connector:

Available Tables

Name	Description
Accounts	Create, update, delete, and query the Accounts you manage in Stripe.
BankAccounts	Create, update, delete, and query the available Bank Accounts in Stripe.
BankAccountTokens	Insert and query the available Bank Account Tokens in Stripe.
Cards	Create, update, and query the available Cards in Stripe.
CardTokens	Create and query the available Card Tokens in Stripe.
Charges	Create, update, and query the available Charges in Stripe.
Customers	Create, update, delete, and query the available Customers in Stripe.
Disputes	Update and query the available Disputes in Stripe.
Refunds	Query the available refunds in Stripe.
TransferReversals	Query the available categories in Stripe.
Transfers	Query the available transfers in Stripe.

Available Views

Name	Description
AvailableBalance	Query the Available Balance in Stripe.
BalanceHistory	Query Balance History in Stripe.
CountrySpecs	Query the available Country Specs in Stripe.
Events	Query the available events in Stripe.
Orders	Query the available orders in Stripe.
PendingBalance	Query the available balance in Stripe.
Products	Query the available products in Stripe.
Skus	Query the available SKUs in Stripe.

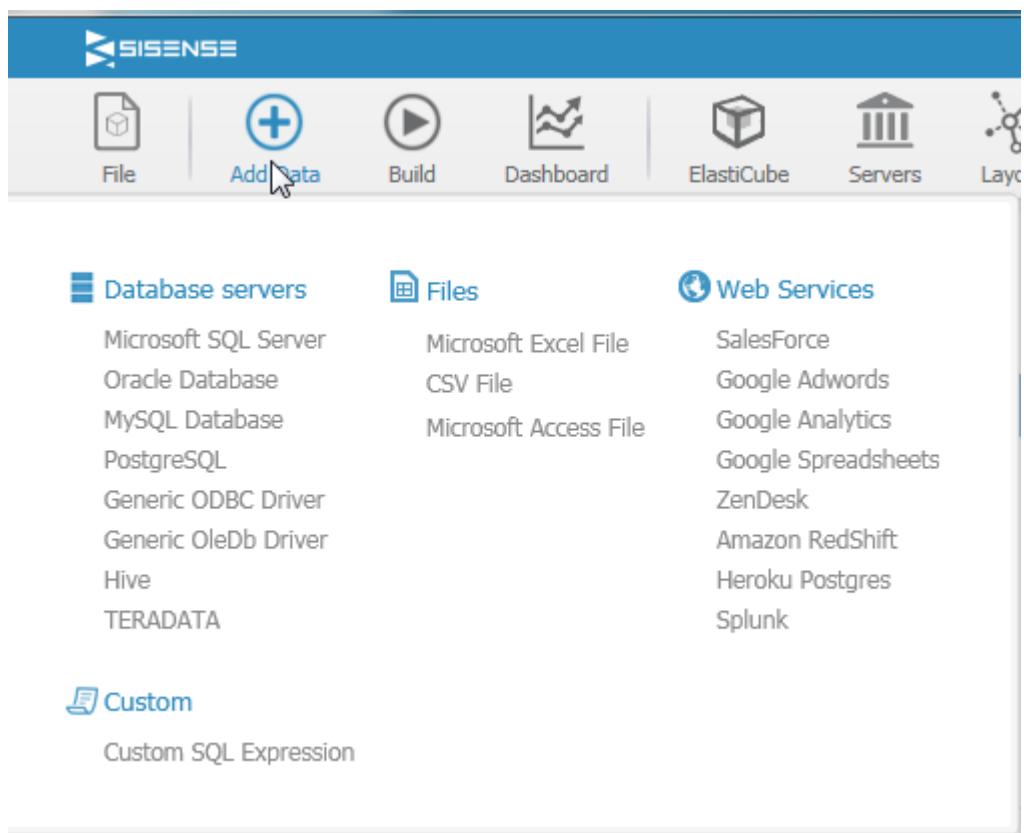
Limitations

1. Not all fields can be filtered.
2. When a field cannot be filtered, you can not perform an accumulative build on that field.

Connecting to SQL Server

The ElastiCube Manager enables easy and quick access to databases, tables and views contained with Microsoft SQL Server databases. The steps below detail how to connect to this type of data source.

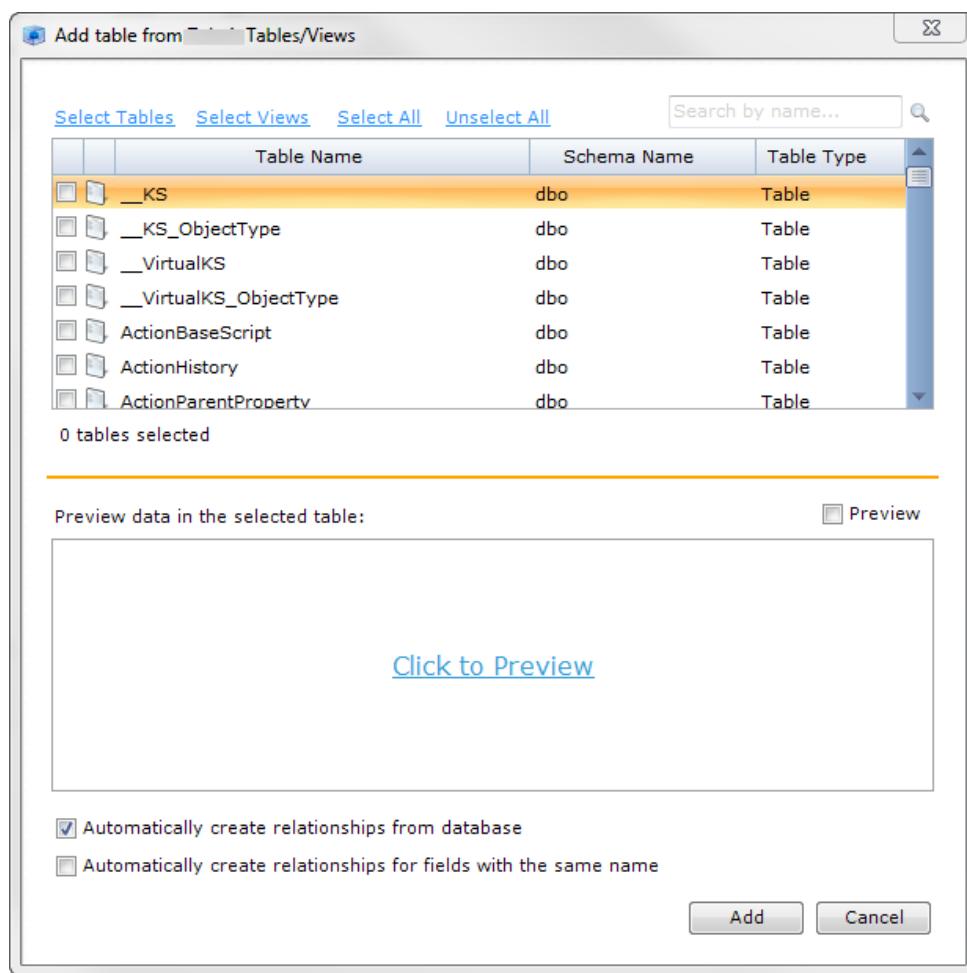
1. Click **Add data** on the top menu of the ElastiCube Manager.
2. Under **Database servers**, select **Microsoft SQL Server**.



The screenshot shows the ElastiCube Manager's main interface. At the top, there's a blue header bar with the Sisense logo and several navigation icons: File, Add data (which is highlighted with a cursor), Build, Dashboard, ElastiCube, Servers, and Layout. Below the header is a large central area divided into three columns. The left column is titled 'Database servers' and lists: Microsoft SQL Server, Oracle Database, MySQL Database, PostgreSQL, Generic ODBC Driver, Generic OleDb Driver, Hive, and TERADATA. The middle column is titled 'Files' and lists: Microsoft Excel File, CSV File, and Microsoft Access File. The right column is titled 'Web Services' and lists: SalesForce, Google Adwords, Google Analytics, Google Spreadsheets, ZenDesk, Amazon RedShift, Heroku Postgres, and Splunk. At the bottom of the central area, there's a section titled 'Custom' with the text 'Custom SQL Expression'.

3. The Connect to SQL Server window is displayed.
4. **Database server location:** Enter the computer/server IP address of the database. To connect to a database running on your own computer enter **localhost**.
5. Select either **Windows Authentication** if configured with the database or alternatively, **Use the Following User Name & Password**, and enter the database credentials.

6. (Optional) Select **Encrypt Connection** to configure the driver to encrypt all communication with the SQL Server instance before sending it over the network, and then select **Trust Server Certificate**.
7. Click **Connect to server**
8. From the **Select Database** dropdown list, select the database you want to work with, and click **OK**.
9. All tables and views associated with the database will appear in a new window.



10. To view a preview of data contained in a particular table, highlight the table or view in the list and click the preview pane below. To preview the table, select the **Preview** checkbox. Enable the checkbox next to each table or view you would like to use.

11. Existing relationships between tables can be automatically replicated in the ElastiCube by selecting the **Automatically create relationships from database** option. Likewise fields with similar names can be linked by selecting the **Automatically create relationships for fields with the same name** option.
12. If you want to customize the data before importing it into the ElastiCube, you can run a custom SQL query to manipulate the data. This can be useful, for example, when you want to import only a portion of the data, rather than all of the data.
13. Once all relevant tables are selected, click **Add**.

Connecting to Twitter

The Sisense Twitter connector is a standalone connector that allows you to import data from Twitter's API into the ElastiCube Manager.

After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to Twitter's API. To obtain a connection string, you will need to create a Twitter app.

Once you have connected to Twitter, you can import a variety of tables from the Twitter API.

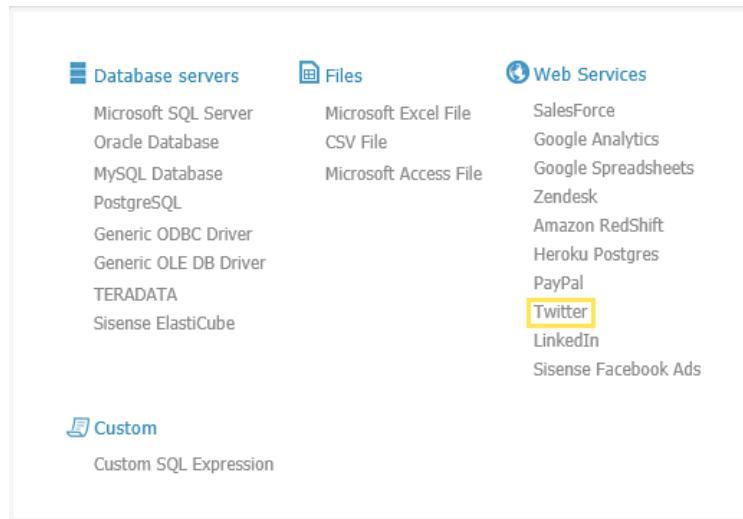
This section describes how to install the Twitter connector, how to connect to Twitter with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the Twitter Connector

Sisense provides the Twitter connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Twitter connector:

1. [Download](#) the Twitter installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The Twitter connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.



Connecting to the Twitter REST API

To access Twitter's REST API from Sisense, you must provide valid OAuth Twitter credentials. These credentials are provided by Twitter when you create an application.

Creating an App

You can follow the steps below to obtain the OAuth client credentials:

1. Log in to <https://dev.twitter.com/apps>.
2. Click **Create New App**.
3. Enter the name, description, and website to be displayed to users when they connect.
4. Define the callback URL setting. If you are making a Web app, set the callback URL to the URL you would like users to be returned to after they have granted your application privileges. If you are making a desktop application, set the callback URL to <http://localhost/>.

Once you have created the app, you will be redirected to a page with information about your app.

If you intend to communicate with Twitter only as the currently authenticated user, then you can obtain the OAuthAccessToken and OAuthAccessTokenSecret directly from this page on Twitter. The OAuthAccessToken and OAuthAccessTokenSecret are listed under the OAuth Settings in the Your Access Token section. You can then connect by setting these connection string properties and do not need to continue to the other steps.

Otherwise, if you need to generate access tokens for other user accounts besides the one you used to create the app on Twitter, use the consumer key and consumer secret in the following steps.

Accessing Ads Data from Twitter

Before requesting data from Twitter, you must apply for access to your data from Twitter by submitting this [request form](#). As part of your request, you must provide your APP ID. Click [here](#) for instructions on how to find your APP ID.

After you have created a Twitter app, you can manage it [here](#). This is useful if you need to set permissions so you can access the relevant data.

Authenticating through Twitter

After setting the following connection properties, you are ready to connect:

- ▶ **OAuthClientId:** Set this to the consumer key in your app settings.
- ▶ **OAuthClientSecret:** Set this to the consumer secret in your app settings.
- ▶ **CallbackURL:** Set this to the callback URL you specified in your app settings.

- ▶ **InitiateOAuth:** Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken and OAuthAccessTokenSecret.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver then completes the OAuth process:

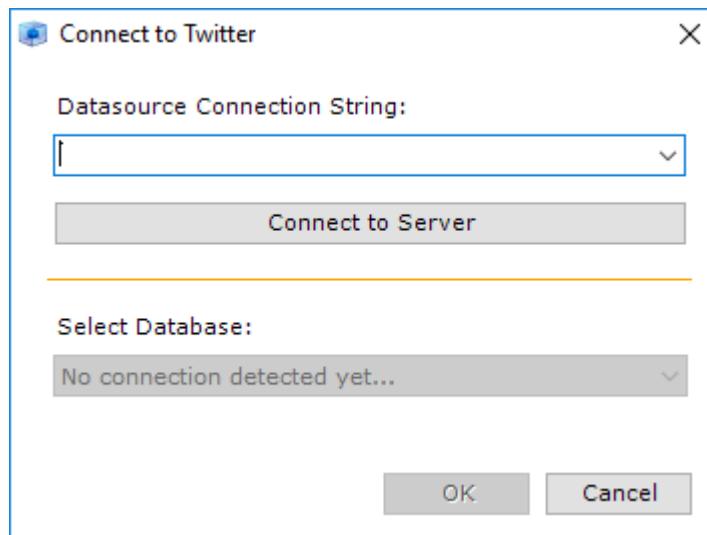
1. Extracts the access token from the callback URL and authenticates requests.
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Adding Twitter Tables to your ElastiCube

Sisense uses connection strings to connect to Twitter and import data into the ElastiCube Manager. Each connection string contains a authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add Twitter data:

1. In ElastiCube Manager, click **Add Data** and then, **Twitter**. The Connect to Twitter window is displayed.



2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Twitter is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Twitter and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**. The tables are displayed in the ElastiCube Manager.

Twitter Tables

The Sisense Twitter connector allows you to import the following tables into the ElastiCube Manager.

Name	Description
DirectMessagesReceived	Query and delete direct messages received by the authenticated user.
DirectMessagesSent	Send direct messages and query messages sent by the authenticated user.

Name	Description
Favorites	Create, delete, and query a list of favorite tweets of the authenticated user and allow the user to favorite new tweets or remove existing favorites.
Following	Create, delete, and query a list of users that the current Twitter account is following, otherwise known as friends.
Tweets	Create, delete, and query status updates and tweets from the authenticated user.
AccountSettings	Query account settings about the currently authenticated user.
AdAccounts	Retrieve all of the advertising-enabled accounts the authenticating user has access to.
AdAvailableAudiences	Query available audiences for this account.
AdCampaigns	Retrieve details for some or all campaigns associated with the current account.
AdFundingInstruments	Retrieve some or all funding instruments associated with the account specified in the path.
AdInsights	Provide qualitative information about various Twitter audiences.
AdLineItems	Retrieve some or all funding instruments associated with the account specified in the path.
AdPromotedTweets	Retrieve all of the advertising-enabled tweets the authenticating user has access to.
AdStats	Query information about all available metrics for a certain entity.
Followers	Query a list of users following the current Twitter account.

Name	Description
ListMembers	Query the members of a specified list.
Lists	Query Twitter list information based on a set of criteria.
ListSubscribers	Query the subscribers to a specified list.
Mentions	Query a list of retweets of the authenticated user.
Retweets	Query a list of retweets of the authenticated user.
Trends	Query the daily trending topics from Twitter.
Users	Query a list of users based on the SearchTerms, Id, or Screen_Name.

Connecting to Xero Accounting

The Sisense Xero connector is a standalone connector that allows you to import data from Xero's API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the Xero APIs. To obtain a connection string, you will need to create a Xero app.

Once you have connected to Xero, you can import a variety of tables from the Xero API.

This section describes how to install the Xero connector, how to connect to Xero with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the Xero Connector

Sisense provides the Xero connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the Xero connector:

1. [Download](#) the Xero installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.

The Xero connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.

Connecting to the Xero REST API

To access Xero's REST API from the ElastiCube Manager, you must provide valid Oauth Xero credentials through a connection string.

These credentials are provided by Xero when you register an application.

After you receive your credentials from Xero, you can create the connection string and provide Sisense with it to connect to your data.

Registering an App

Follow the steps below to obtain the OAuth client credentials, the OAuthClientId and OAuthClientSecret:

1. Log in to the Xero Developer Portal.
2. Click **My Applications > Add Application**.
3. Select the **Public** option.
4. Enter a name for your application and the URL of your company.
This information is displayed to users when they connect.
5. If you are making a Desktop application, set the Callback Domain to 'localhost'.
If you are making a Web application, set the Callback Domain to the domain name of the URL where the user returns with the token that verifies that they have granted your app access.
6. After you click **Save**, you are shown your OAuth credentials, the consumer key, and consumer secret.

Authenticating through Xero

After setting the following connection properties, you are ready to connect:

- ▶ **OAuthClientId:** Set this to the consumer key in your app settings.
- ▶ **OAuthClientSecret:** Set this to the consumer secret in your app settings.

- ▶ **XeroAppAuthentication:** Set this to the type of your application. Allowed values: PUBLIC or PARTNER. Default: PUBLIC.
- ▶ **InitiateOAuth:** Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken connection property.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver then completes the OAuth process:

1. Extracts the access token from the callback URL and authenticates requests.
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Adding Xero Tables to your ElastiCube

Sisense uses connection strings to connect to Xero and import data into the ElastiCube Manager. Each connection string contains authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add Xero data:

1. In ElastiCube Manager, click **Add Data** and then, **Xero**. The Connect to Xero window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. Xero is displayed in the Select Database list.
4. Click **OK**. Sisense connects to Xero and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.
The tables are displayed in the ElastiCube Manager.

Xero Tables

Xero's RESTful APIs expose the following Xero tables that you can import into the ElastiCube Manager through the Sisense Xero connector:

Available Tables

Name	Description
<="" b="">	
Accounts	Create, delete, and query accounts for a Xero organisation.
BankTransactions	Create, update, delete, and query bank transactions for a Xero organisation.
BankTransfers	Usage information for the operation BankTransfers.rsd.
ContactGroups	Create, update, and query contact groups for a Xero organisation.
Contacts	Create, update, and query contacts for a Xero organisation.
CreditNotes	Create, update, delete, and query credit notes for a Xero organisation.
Employees	Create, update, and query employees for a Xero organisation.
ExpenseClaims	Create, update, and query expense claims for a Xero organisation.
Invoices	Create, update, delete, and query invoices for a Xero organisation.
Items	Create, update, delete, and query items for a Xero organisation.
LinkedTransactions	Create, update, and query LinkedTransactions for a Xero organisation.
ManualJournals	Create, update, and query manual journals for a Xero organisation.

Name	Description
Payments	Create and query payments for a Xero organisation.
PurchaseOrders	Create, update, delete, and query purchase orders for a Xero organisation.
Receipts	Create, update, delete, and query receipts for a Xero organisation.
TaxRates	Usage information for the operation TaxRates.rsd.
Payroll US Data Model	
Employees	Create, update, and query employees for a Xero organisation.
PayItemsBenefits	Create, update, and query PayItems for a Xero organisation.
PayItemsDeductions	Create, update, and query PayItems for a Xero organisation.
PayItemsEarnings	Create, update, and query PayItems for a Xero organisation.
PayItemsReimbursements	Create, update, and query PayItems for a Xero organisation.
PayItemsTimeOff	Create, update, and query PayItems for a Xero organisation.
PayRuns	Create, update, and query payruns for a Xero organisation.
PaySchedules	Create, update, and query PaySchedules for a Xero organisation.
Paystubs	Create, update, and query pay stubs for a Xero organisation.
WorkLocations	Create, update, and query work locations for a Xero organisation.
Payroll AUS Data Model	
Employees	Create, update, and query employees for a Xero organisation.

Name	Description
LeaveApplications	Create, update, and query Leave Applications for a Xero organisation.
PayItemsDeductions	Create, update, and query PayItems for a Xero organisation.
PayItemsEarnings	Create, update, and query PayItems for a Xero organisation.
PayItemsLeave	Create, update, and query PayItems for a Xero organisation.
PayItemsReimbursements	Create, update, and query PayItems for a Xero organisation.
PayrollCalendars	Create, update, and query Payroll Calendars for a Xero organisation.
PayRuns	Create, update, and query payruns for a Xero organisation.
SuperFunds	Retrieve, add and update Payroll Super Funds in a Xero organisation.

Available Views

Name	Description
Accounting Data Model	
AgedPayablesByContact	Query organisation data for a Xero organisation.
AgedReceivablesByContact	Query organisation data for a Xero organisation.
BalanceSheet	Query organisation data for a Xero organisation.
BankSummary	Query organisation data for a Xero organisation.
BrandingThemes	Query branding themes for a Xero organisation.
BudgetSummary	Query organisation data for a Xero organisation.
Currencies	Query currencies for a Xero organisation.

Name	Description
Journals	Query the line items in journals for a Xero organisation.
Organisation	Query organisation data for a Xero organisation.
Overpayments	Create and query Overpayments for a Xero organisation.
Prepayments	Create and query Prepayments for a Xero organisation.
ProfitAndLoss	Query organisation data for a Xero organisation.
RepeatingInvoices	Usage information for the operation RepeatingInvoices.rsd.
TrackingCategories	Query tracking categories for a Xero organisation.
TrialBalance	Query organisation data for a Xero organisation.
Users	Query users for a Xero organisation.
Payroll US Data Model	
TimeOffBalances	Retrieve, add and update an Employee's TimeOff balance in a Xero organisation.
Timesheets	Create, update, and query timesheets for a Xero organisation.
Payroll AUS Data Model	
LeaveBalances	Retrieve, add and update an Employee's Leave balance in a Xero organisation.

Limitations

To access the AgedPayablesByContact and AgedRecievablesByContact tables, you need to specify a particular ContactID, the unique ID of the Contacts table, in the select query

using a WHERE clause. This allows you to retrieve data for a specific contact. These are limitations of the Xero API, as these reports are specific to each contact. Here's an example:

```
SELECT due, paid, total FROM AgedPayablesByContact WHERE  
ContactID='xyz'  
SELECT duedate, reference FROM AgedRecievableByContact WHERE  
ContactID='1234'
```

Due this limitation, you cannot build an ElastiCube with these tables without custom SQL.

Daily Limit

There is a daily limit of 1000 API calls against a single Xero organisation in a rolling 24-hour period.

In addition to the daily limit, a single access token can only be used up to 60 times in a rolling 60-second period.

Encountering a Rate Limit

If you encounter a rate limit, the Xero API will return an HTTP 503 (Service Unavailable) error, with the following message:
"oauth_problem=rate limit exceeded".

Note: If you encounter a rate limit, do not continue to make requests, as this may continue to add to your limitation. If necessary, you may need to queue requests.

Token Limitations

Each access token will only last for 30 minutes. If you want longer access to the organisation, you need the user to re-authorize your application.

Connecting to YouTube

The Sisense YouTube connector is a standalone connector that allows you to import data from the YouTube API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect to the YouTube API through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to YouTube's API. To obtain a connection string, you will need to create a YouTube app.

Once you have connected to YouTube, you can import a variety of tables from the YouTube API.

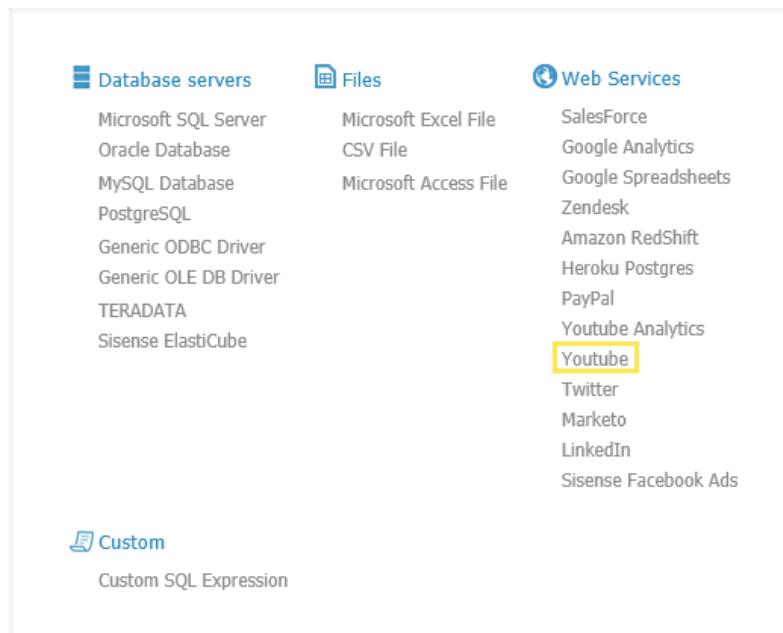
This section describes how to install the YouTube connector, how to connect to YouTube with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the YouTube Connector

Sisense provides the YouTube connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the YouTube connector:

1. [Download](#) the YouTube installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The YouTube connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.



Connecting to the YouTube REST API

To access YouTube's REST API from the ElastiCube Manager, you must provide valid Oauth YouTube credentials through a connection string. These credentials are provided by YouTube when you create an application.

After you receive your credentials from YouTube, you can create the connection string and provide Sisense with it to connect to your data.

Creating an App

You can follow the procedure below to register an app and obtain the OAuth client credentials, the OAuthClientId and OAuthClientSecret:

1. Log in to the Google Developers Console.
2. Click **Create Project** or select an existing project.
3. In the API Manager, click **Credentials > Create Credentials > OAuth Client Id** and then click **Configure Consent**

Screen to customize the information displayed to users when they connect.

4. If you are connecting from a desktop application, click **Other** in the Application Type section. If you are connecting from a Web application, click the Web Application option. In the Authorized Redirect URIs box, enter the URL you want to be used as a trusted redirect URL, where the user will return with the token that verifies that they have granted your app access.
5. Click **Create**. The OAuthClientId and OAuthClientSecret are displayed.
6. Click **Library** and enable the YouTube Data API.

Authenticate to YouTube

After setting the following connection properties, you are ready to connect:

- ▶ **InitiateOAuth**: Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken connection property.
- ▶ **OAuthClientId**: Set this to the Client Id in your app settings.
- ▶ **OAuthClientSecret**: Set this to the Client Secret in your app settings.
- ▶ **CallbackURL**: Set this to `http://localhost`.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver then completes the OAuth process:

1. Extracts the access token from the callback URL and authenticates requests.

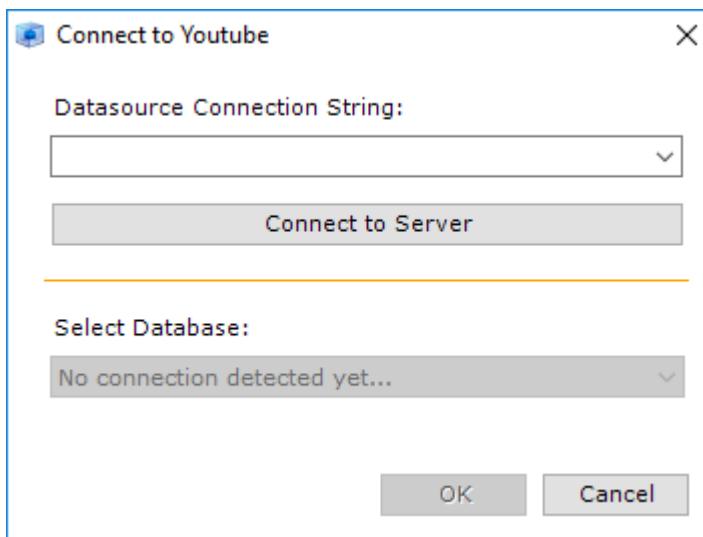
2. Refreshes the access token when it expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Adding YouTube Tables to your ElastiCube

Sisense uses connection strings to connect to YouTube and import data into the ElastiCube Manager. Each connection string contains authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add YouTube data:

1. In ElastiCube Manager, click **Add Data** and then, **YouTube**. The Connect to YouTube window is displayed.



2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. YouTube is displayed in the Select Database list.
4. Click **OK**. Sisense connects to YouTube and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.

6. The tables are displayed in the ElastiCube Manager.

YouTube Tables

The Sisense YouTube connector allows you to import the following tables into the ElastiCube Manager.

Name	Description
Activities	Query YouTube Activities. An Activity resource contains information about an action that a particular Channel, or User, has taken on YouTube.
Captions	Query YouTube Captions. A Caption resource represents a YouTube caption track.
Channels	Query YouTube Channels.
ChannelSections	Query YouTube ChannelSections. A ChannelSection contains information about a set of videos that a channel has chosen to feature.
Comments	Query YouTube Comments.
CommentThreads	Query YouTube CommentThreads. A CommentThread record contains information about a YouTube comment thread, a top-level comment and replies, if any exist, to that comment.
GuideCategories	Query YouTube GuideCategories. A GuideCategory resource identifies a category that YouTube algorithmically assigns based on a content of a channel or other indicators, such as the popularity of the channel.
Languages	Query YouTube i18nLanguages. An i18nLanguage resource identifies an application language that the YouTube website supports. The application language can also be referred to as a UI language.

Name	Description
PlayListItems	Query YouTube PlayListItems. A PlayListItem resource identifies another YouTube entity, such as a Video, that is included in a PlayList. In addition, the PlayListItem record contains details about how that entity is used in that PlayList.
PlayLists	Query YouTube PlayLists. A PlayList is a collection of videos that can be viewed sequentially and shared with other users.
Regions	Query YouTube i18nRegions. An i18nRegion resource identifies a geographic area that a YouTube user can select as the preferred content region. The content region can also be referred to as a content locale.
Subscriptions	Query YouTube User Subscriptions. A Subscription notifies a User when new Videos are added to a Channel, or when another user takes one of several actions on YouTube, such as uploading a Video, rating a Video, or commenting on a Video.
VideoAbuseReportReasons	Query YouTube VideoAbuseReportReasons. A VideoAbuseReportReason resource contains information about a reason that a Video would be flagged for containing abusive content.
VideoCategories	Query YouTube VideoCategories. A VideoCategory resource identifies a category that has been or could be associated with uploaded Videos.
Videos	Query YouTube Videos.

Connecting to YouTube Analytics

The Sisense YouTube Analytics connector is a standalone connector that allows you to import data from the YouTube Analytics API into the ElastiCube Manager. After you have downloaded and installed the connector, you can connect to the YouTube Analytics API through a connection string you provide Sisense in the ElastiCube Manager. The connection string is used to authenticate users who connect to the YouTube Analytics API. To obtain a connection string, you will need to create a YouTube Analytics app.

Once you have connected to YouTube Analytics, you can import a variety of tables from the YouTube Analytics API.

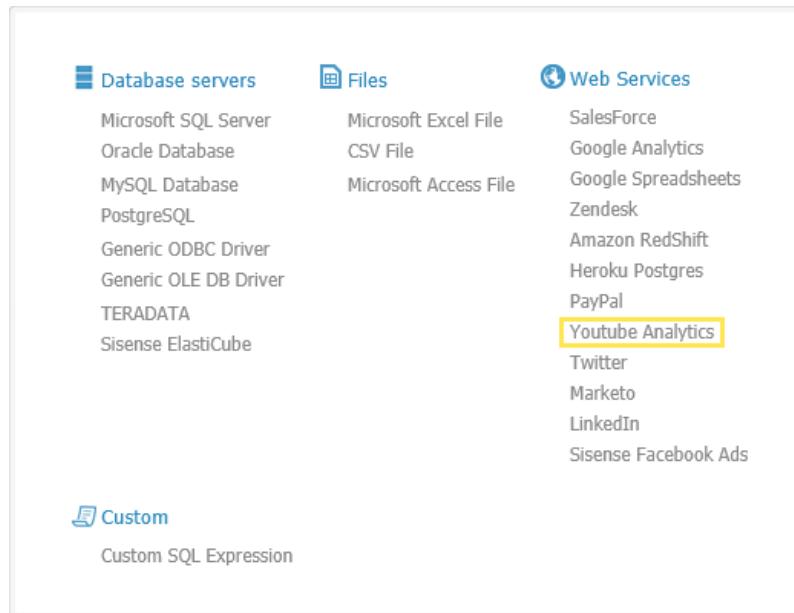
This section describes how to install the YouTube Analytics connector, how to connect to YouTube Analytics with a connection string, and what tables you can import into the ElastiCube Manager.

Installing the YouTube Analytics Connector

Sisense provides the YouTube Analytics connector as a standalone connector that you can download and add to your list of default Sisense connectors.

To install the YouTube Analytics connector:

1. [Download](#) the YouTube Analytics installation file.
2. Open the installation file and click **Install**.
3. After the installation process is complete, click **Close**.
4. The YouTube Analytics connector is displayed in the ElastiCube Manager under **Add Data > Web Services**.



Connecting to the YouTube Analytics REST API

To access YouTube Analytics' REST API from the ElastiCube Manager, you must provide valid OAuth YouTube Analytics credentials through a connection string. These credentials are provided by YouTube Analytics when you create an application.

After you receive your credentials from YouTube Analytics, you can create the connection string and provide Sisense with it to connect to your data.

Creating an App

You can follow the procedure below to register an app and obtain the OAuth client credentials, the Client Id and Client Secret:

1. Log in to the Google API Console.
2. Select a project or create a new one.
3. In the API Manager menu, click **Credentials > Create Credentials > OAuth Client Id > Other.**

4. Click **Create**. The OAuthClientId and OAuthClientSecret are displayed.
5. Click **Library** > YouTube Analytics API > Enable API.

Authenticating through YouTube Analytics

After setting the following properties, you are ready to connect:

- ▶ **OAuthClientId**: Set this to the Client Id in your app settings.
- ▶ **OAuthClientSecret**: Set this to the Client Secret in your app settings.
- ▶ **ChannelId**: Set this to the YouTube Analytics Channel (Profile). This can be set to the Id of the channel. If not specified, MINE will be used.
- ▶ **ContentOwnerId**: Set this to the Id of the content owner.
- ▶ **CallbackURL**: Set this to `http://localhost`.

When you connect the driver opens the OAuth endpoint in your default browser. Log in and grant permissions to the application. The driver completes the OAuth process.

Adding YouTube Analytics Tables to your ElastiCube

Sisense uses connection strings to connect to YouTube Analytics and import data into the ElastiCube Manager. Each connection string contains authentication parameters that the data source uses to verify your identity and what information you can export to Sisense. To learn more, see Connection String Parameters.

To add YouTube Analytics data:

1. In ElastiCube Manager, click **Add Data** and then, **YouTube Analytics**. The Connect to YouTube Analytics window is displayed.
2. In **Datasource Connection String**, enter your connection string.
3. Click **Connect to Server**. YouTube Analytics is displayed in the Select Database list.
4. Click **OK**. Sisense connects to YouTube Analytics and displays a list of tables available for you to import.
5. Select the relevant tables and click **Add**.

The tables are displayed in the ElastiCube Manager.

YouTube Analytics Tables

The Sisense YouTube Analytics connector allows you to import the following tables into the ElastiCube Manager.

Name	Description
GroupItems	Create, delete, and query Items that compose a Group.
Groups	Create, update, delete, and query YouTube Analytics Groups.
AdPerformanceReports	<p>Query Ad Performance Reports. Ad Performance Reports provide impression-based metrics for ads that ran during video playbacks. These metrics account for each ad impression, and each video playback can yield multiple impressions.</p> <p>To access AdPerformanceReports, in the WHERE clause, you should specify at least the video, group or a supported combination of uploaderType and claimedStatus.</p> <p>Optionally, the following filters can be specified: country, continent, subcontinent</p>

Name	Description
	<p>For dimensions, the 'adType' type is required. Optionally, 'day' dimension can be added. An example query similar to yours would be (notice, AdPerformanceReports view does not have a channel column, instead you have to use a video filter. This is a limitation of the YouTube Analytics API):</p> <pre>SELECT AdType, Day, GrossRevenue, StartDate, EndDate FROM AdPerformanceReports WHERE video='vid123' and continent = '019'</pre>
Demographics	Query YouTubeAnalytics Demographics grouped by age groups and gender.
Devices	Query YouTubeAnalytics Devices or Operating System statistics. This table allows you to aggregate viewing statistics based on the manner in which viewers reached your video content. For example, you can identify the number of views that occurred on mobile devices or game consoles.
EngagementReports	Query Engagement Reports info. For example, how many times a user shared a video.
TimeBasedReports	Query time-based info on views, subscribers, etc. For example, how many new subscribers were gained at a specific time.
TopVideos	Query the TopVideos for a channel or the TopPlaylists if you are a content owner. You can also use this view to query playback detail and geographic filters.
TrafficSources	Query TrafficSources. Statistics are based on the manner in which viewers reached your video and playlist content.

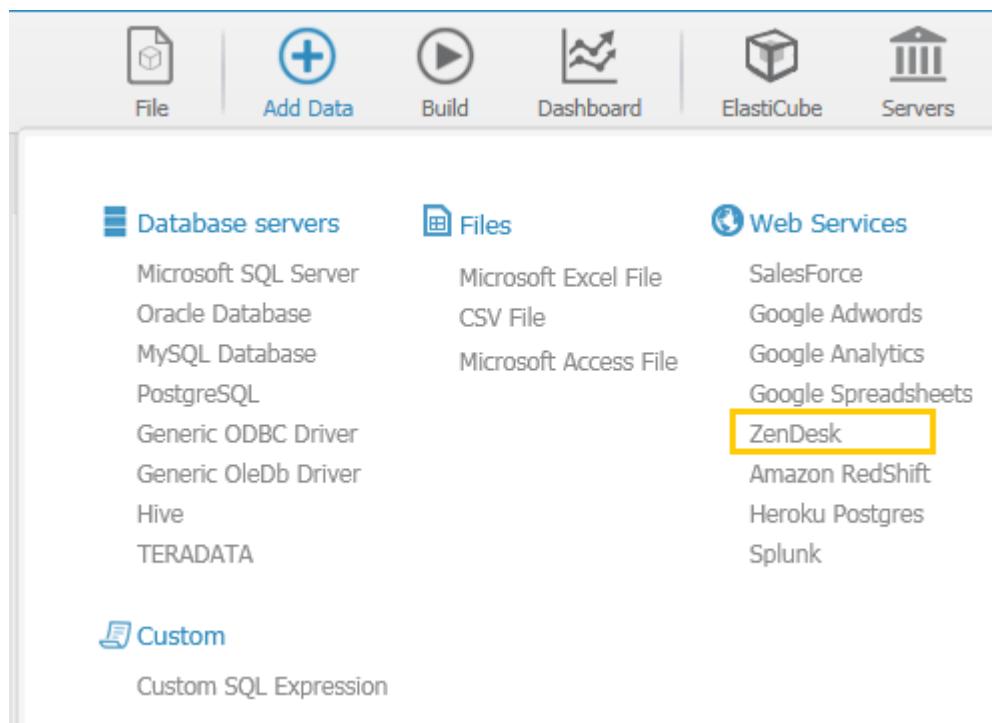
Connecting to Zendesk

The ElastiCube Manager enables easy and quick access to tables and views contained within Zendesk databases.

Note: Only non-archived tickets are supported.

The steps below describe how to connect to this type of data source.

1. Click **Add Data** in the top menu of the ElastiCube Manager.
2. Under the **Web Services** category, select **Zendesk**.



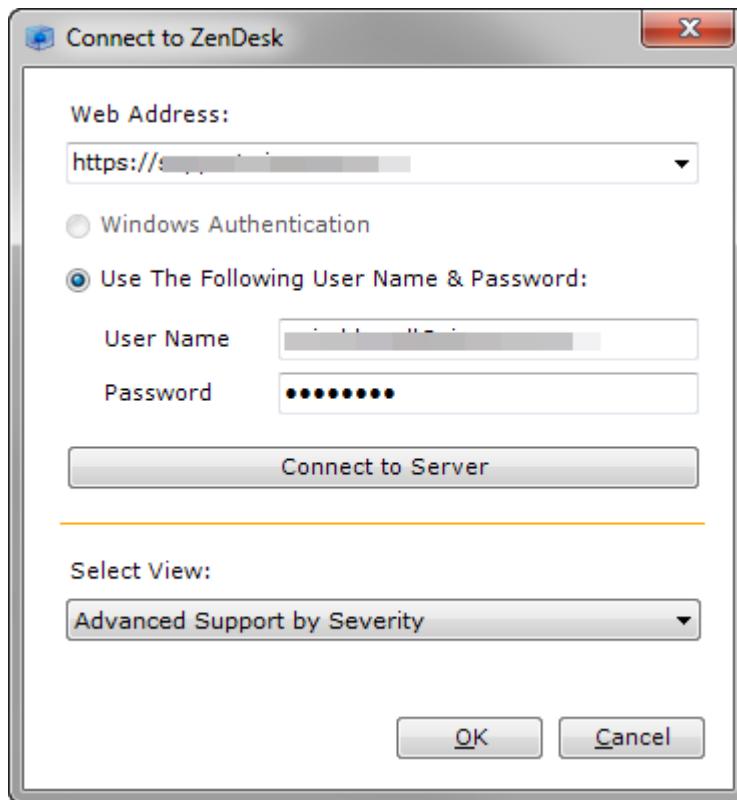
3. You will be prompted to enter the following information:

Web Address: Enter the web address where your Zendesk deployment is hosted, for

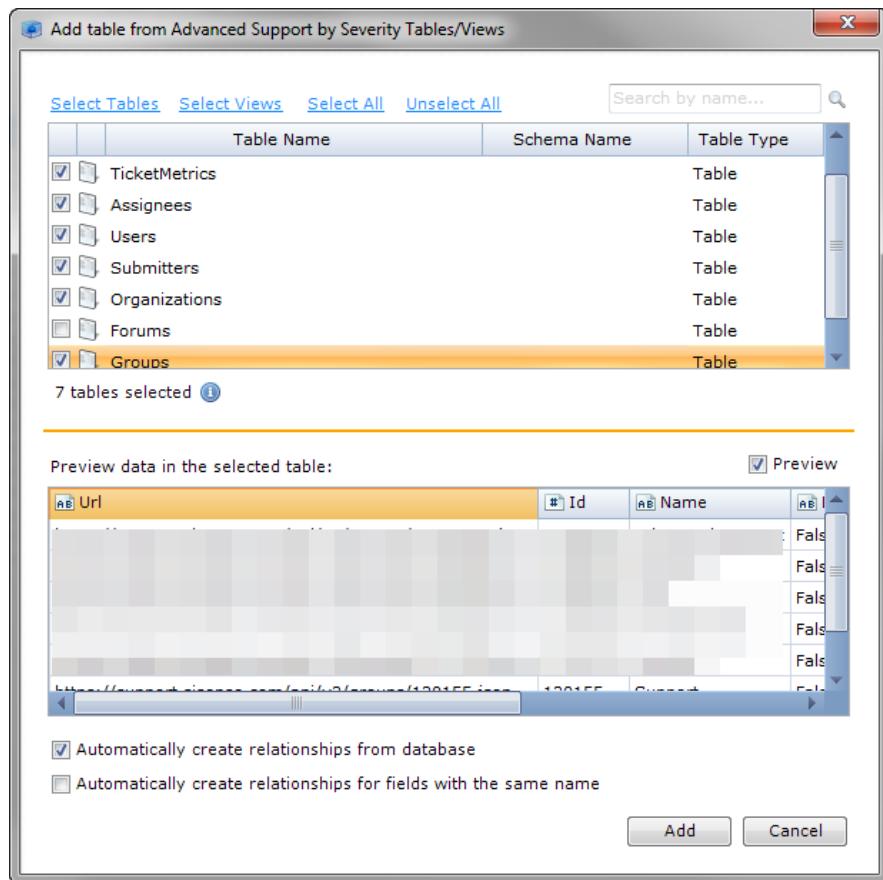
example: *http://support.yourcompany.com*. **Note:** You must include *http://* in your web address.

Login details: Either use your Windows login details if they are configured with Zendesk, or alternatively enter the username and password used to connect to Zendesk.

4. Click **Connect to Server**.
5. A list of available views will appear in the list box below.



6. Select the relevant view you want to work with and click **OK**.
All tables associated with the Zendesk view will appear in a new window.
7. To preview data contained in a particular table, highlight the table or view in the list and click the preview pane below. To preview the table, select the **Preview** checkbox.

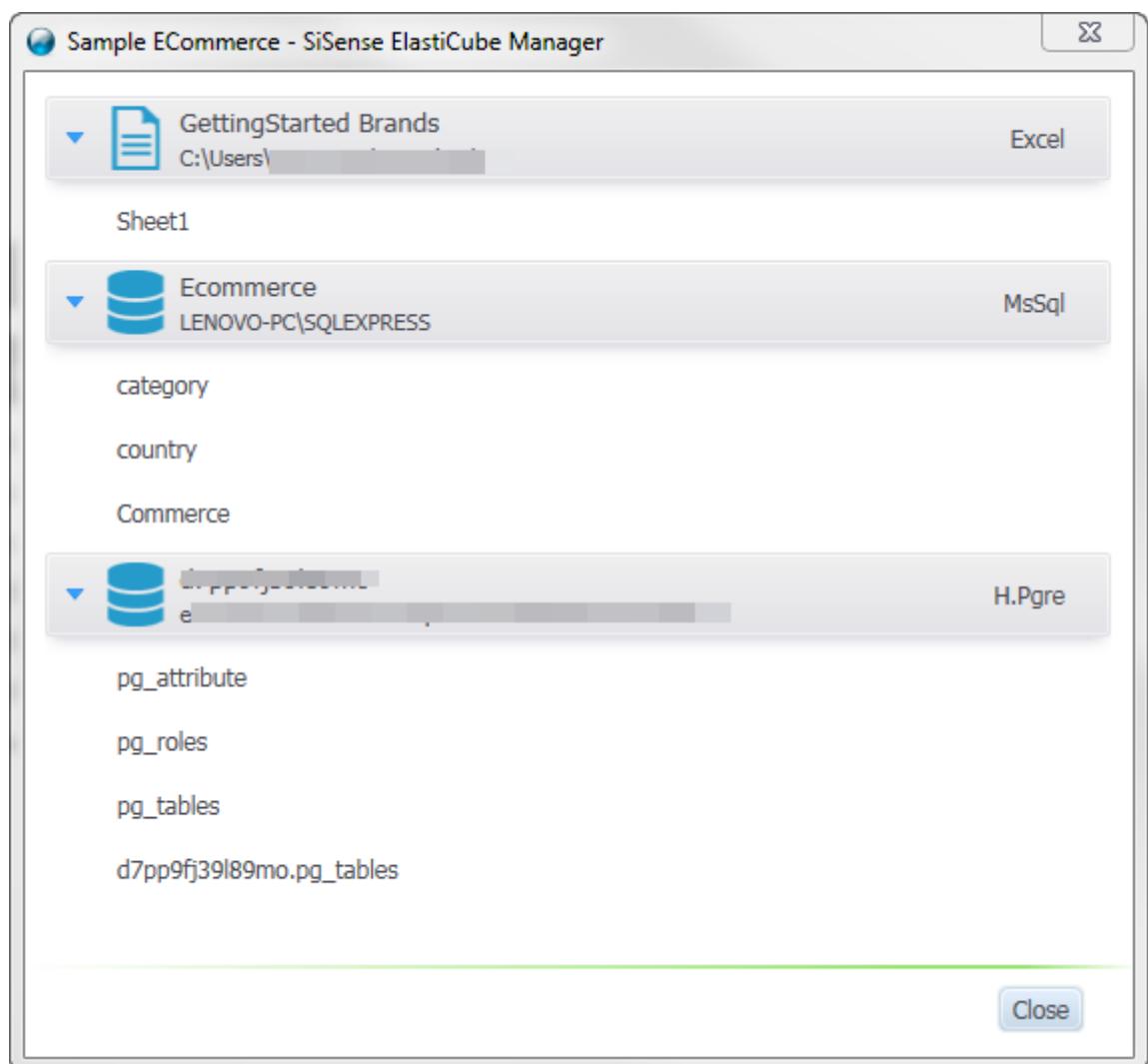


8. Select the checkbox next to each table or view you want to use.
 9. Existing relationships between tables can be automatically replicated in the ElastiCube by selecting the **Automatically create relationships from database** checkbox. Likewise fields with similar names can be linked by selecting the **Automatically create relationships for fields with the same name** checkbox.
 10. Once all relevant tables are selected, click **Add**.

Changing Connectivity Settings for Data Sources

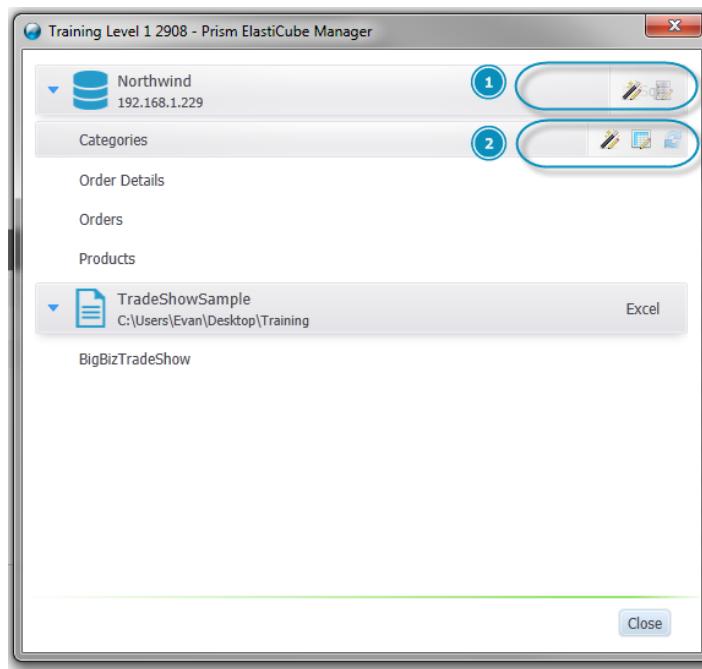
This section describes how to change the connectivity settings for an existing data source. You can update the source of the data, on both the database and table levels. For example, a database or data file may have been moved to a new location, or updated, and the ElastiCube needs to be updated accordingly.

1. In the top menu, Click **ElastiCube**, and under **Build Settings**, select **Change Connectivity Settings...**
2. A list of all existing data sources and associated data tables appear.



3. Hover over the data source to reveal the following options:
 - ▶ **Change Source Provider:** Click to select a data source connection type, for example, change MS SQL to an Oracle database. When done, click **Close**.
 - ▶ **Change Source Database:** Click to edit the login details to the database as well as connections to associated tables.
4. Hover over the data table to reveal the following options:

- ▶ **Change Source Provider:** Click to change the data source connection type, for example, from MS SQL to an Oracle database.
 - ▶ **Change Source Table:** Click to change the source tables using the current connection settings.
 - ▶ **Refresh:** Refresh the tables' fields based on current settings.
Note: The data source name, fields and field types must remain the same for existing widgets in the dashboard to be able to reference the data correctly.
5. To remove a data source, you must first delete all related tables from the ElastiCube schema. After deleting all tables, the data source will no longer appear.



Connection String Parameters

Sisense uses connection strings to connect to some data sources and import data into the ElastiCube Manager.

Connection strings have the following structure:

```
jdbc:DataSourceName:Property1=Value1;Property2=Value2;
```

The following is an example of a Exact connection string:

```
jdbc:exactonline:OAuthClientId=xxxxxxxxxxxxxx;OAuthClientSecret=xxx  
xxxxxxxxxx;Region='United  
States';Division=xxxx;InitiateOAuth=GETANDREFRESH;OAuthCallback  
URL=localhost:12345;
```

This page lists the mandatory and optional parameters required by the data source you want to connect to. Mandatory parameters must be defined and included in your connection string whereas optional parameters are only recommended by Sisense.

In addition, some parameters are required by specific data sources. This section describes specific parameters for the following web services::

- ▶ Exact
- ▶ Facebook
- ▶ HubSpot
- ▶ Quickbooks Online
- ▶ ServiceNow
- ▶ Xero
- ▶ YouTube Analytics

Mandatory Properties

- ▶ **OAuthClientId:** The client ID assigned when you register your application with an OAuth authorization server.
- ▶ **OAuthClientSecret:** Set this to the App Secret in your app settings.
- ▶ **CallbackURL:** The OAuth callback URL to return to when authenticating. This value must match the callback URL you specify in your app settings. Set this to `http://localhost`.
- ▶ **InitiateOAuth:** Set this to `GETANDREFRESH`. You can use `InitiateOAuth` to avoid repeating the OAuth exchange and manually setting the `OAuthAccessToken` connection property.
- ▶ **Timeout:** The value in seconds until the timeout error is thrown, cancelling the operation. Set to 0 to disable time outs.

Optional Parameters

- ▶ **OAuthSettingsLocation:** The location of the settings file where OAuth values are saved for each user when `InitiateOAuth` is set to `GETANDREFRESH`. See [Switching between Accounts](#).
- ▶ **LogFile:** Creates a log file. The value of this parameter is the location on your drive where the log file is to be created. To determine how much detail is included in the log file, see the parameter `Verbosity` below.
- ▶ **Verbosity:** The verbosity level that determines the amount of detail included in the log file. Set the verbosity from 1–5.
 - ▶ 1 will log the query, the number of rows returned by it, the start of execution and the time taken, and any errors.
 - ▶ 2 will log everything included in Verbosity 1, cache queries, and HTTP headers.
 - ▶ 3 will additionally log the body of the HTTP requests.

- ▶ 4 will additionally log transport-level communication with the data source. This includes SSL negotiation.
- ▶ 5 will additionally log communication with the data source and additional details that may be helpful in troubleshooting problems. This includes interface commands.

Switching between Accounts

When you connect to the Facebook data source, Sisense saves your OAuth values in the file OAuthsettings.txt file located at .\Users\xxx\AppData\Roaming\CDATA\Facebook Data Provider on your Sisense server. To connect to the Facebook data source with another user on the same machine, you must delete the OAuthsettings.txt file. Sisense will then generate a new file for that user.

Another option to support multiple users is to define the location and file name of an OAuthsettings file for each unique user in your connection string through the **OAuthSettingsLocation** parameter.

When each user connects to the data source, Sisense generates the OAuth file with the file name you specify in the location you define. In the examples below, two users are allowed to access the Facebook data source and for each user, Sisense generates a file that contains that user's OAuth values in the location defined in the string.

```
jdbc:facebook:OAuthSettingsLocation=C:\facebook\auth\john.txt;O
AuthClientId=11276856774486;OAuthClientSecret
=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;
Version=2.8;CallbackURL=http://localhost/;
```

```
jdbc:facebook:OAuthSettingsLocation=C:\facebook\auth\sally.txt;OAuthClientId=11276856774486;OAuthClientSecret=064c70d78567jm2b7e7e4224fad;InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhost/;
```

In the example above, two OAuth files are created, one for John and one for Sally in the location C:\facebook\auth\.

This is useful if you support many users who each need to access the Facebook data source.

Data Source Parameters

This section lists parameters that are required to be inside connection strings

Exact

Region: Set this to the region of the Exact Online service you want to connect to.

Division: Set this to the division of the Exact Online administration.

The following is an example of an Exact connection string:

```
jdbc:exactonline:OAuthClientId=xxxxxxxxxxxxxx;OAuthClientSecret=xxxx  
xxxxxxxxxx;Region='United  
States';Division=xxxx;InitiateOAuth=GETANDREFRESH;OAuthCallback  
URL=localhost:12345;
```

Facebook

Rate Limitations

- ▶ **ThrottleSeconds:** Indicates how many seconds to wait until sending another insight request when the ThrottleMaxPercent has been met.
- ▶ **ThrottleMaxPercent:** The threshold set for throttling further insight requests. The defaults for ThrottleSeconds and ThrottleMaxPercent are 10 and 95.
- ▶ **RetryWaitTime:** The amount of time in milliseconds to wait before retrying to query the Facebook API.
- ▶ **MaximumRequestRetries:** The maximum amount of request retries.

The following is an example of a Facebook connection string:

```
jdbc:facebookOAuthClientId=xxxxxxxxxxxxxx;OAuthClientSecret=xxxxxx
xxxxxxxxxxxx;
InitiateOAuth=GETANDREFRESH;Version=2.8;CallbackURL=http://localhos
t;Timeout=0;
```

Optional Parameters

- ▶ **Aggregate Format:** The format aggregate or collection columns should return in.
- ▶ **Authenticate As Page:** The name or Id of a page to authenticate as when making requests to Facebook.
- ▶ **Auto Cache:** The AutoCache property controls automatic caching of data.
- ▶ **Cache Connection:** The connection string for the cache database.
- ▶ **Cache Driver:** The database driver to be used to cache data.
- ▶ **Cache Location:** Specifies the path to the cache when caching to a file.

- ▶ **Cache Metadata:** Whether or not to cache the table metadata. The cache will be stored in memory unless CacheConnection or CacheLocation are set.
- ▶ **Callback URL:** The OAuth callback URL to return to when authenticating. This value must match the callback URL you specify in your app settings.
- ▶ **Firewall Password:** A password, if authentication is required to connect through a firewall.
- ▶ **Firewall Port:** The TCP port for the firewall FirewallServer —see the description of the FirewallServer option for details.
- ▶ **Firewall Server:** Specify a firewall name or IP address to authenticate requested connections, if necessary.
- ▶ **Firewall Type:** The type of firewall to connect through.
- ▶ **Firewall User:** The user name to authenticate with the firewall.
- ▶ **Initiate OAuth:** Set this property to initiate the process to obtain or refresh the OAuth access token when you connect.
- ▶ **Location:** A path to the directory that contains the schema files defining tables, views, and stored procedures.
- ▶ **Logfile:** A path to the log file.
- ▶ **Max Log File Size:** A string specifying the maximum size in bytes for a log file (ex: 10MB). When the limit is hit, a new log is created in the same folder with the date and time appended to the end.
- ▶ **OAuth Access Token:** The access token for connecting using OAuth.
- ▶ **OAuth Client Id:** The client Id assigned when you register your application with an OAuth authorization server.

- ▶ **OAuth Client Secret:** The client secret assigned when you register your application with an OAuth authorization server.
- ▶ **OAuth Settings Location:** The location of the settings file where OAuth values are saved when InitiateOAuth is set to GETANDREFRESH or REFRESH.
- ▶ **Offline:** Use offline mode to get the data from the cache instead of the live source.
- ▶ **Other:** The other parameters necessary to connect to a data source, such as username and password, when applicable.
- ▶ **Pagesize:** The maximum number of results to return per page from Facebook.
- ▶ **Proxy Auth:** Scheme The proxy server authorization scheme (default: BASIC).
- ▶ **Proxy Auto Detect:** This indicates whether to use the default system proxy settings or not. Set ProxyAutoDetect to FALSE to use custom proxy settings. This takes precedence over other proxy settings.
- ▶ **Proxy Password:** A password, if authenticating with a proxy server.
- ▶ **Proxy Port:** The TCP port for the proxy ProxyServer (default: 80).
- ▶ **Proxy Server:** If a proxy server is given, then the HTTP request is sent to the proxy instead of the specified server.
- ▶ **Proxy SSL Type:** The SSL type to use when connecting to the proxy server (default: AUTO).
- ▶ **Proxy User:** A user name, if authentication is to be used for the proxy.

- ▶ **Pseudo Columns:** Indicates whether or not to include pseudo columns as columns to the table.
- ▶ **Query Cache:** The time duration, in seconds, for which the in-memory cached data is reused.
- ▶ **Readonly:** You can use this property to enforce read-only access to Facebook from the provider.
- ▶ **RTK:** The runtime key used for licensing.
- ▶ **Search Terms:** Default SearchTerms if none are specified. Used for some tables, such as Users, where SearchTerms may be specified.
- ▶ **SSL Server Cert:** The certificate to be accepted from the server when connecting using TLS/SSL.
- ▶ **Support Enhanced SQL:** If set to true, the provider will cache the entire table in memory and issue the current query against the memory cache. Allows queries with aggregate functions that are otherwise unsupported. The entire memory cache is discarded after the query is executed, unlike QueryCache.
- ▶ **Tables:** Comma-separated list of tables to be listed. For example: Tables=TableA,TableB,TableC.
- ▶ **Target:** A default target if none is specified. Used for some tables, such as Comments, where a target may be specified.
- ▶ **Timeout:** The value in seconds until the timeout error is thrown, canceling the operation.
- ▶ **Verbosity:** The verbosity level that determines the amount of detail included in the log file.
- ▶ **Version:** The Facebook Graph API version to use.

HubSpot

- ▶ **HubId:** Set this to the Hub Id of the HubSpot account you are connecting to.

The following is an example of a HubSpot connection string:

```
jdbc:hubspot:OAuthClientId=xxxxxxxxxxxxxx;OAuthClientSecret=xxxxxxxxxx  
xxxxxxxxxx;InitiateOAuth=GETANDREFRESH;  
HubId=xxxxxxxx;CallbackURL=http://localhost;Timeout=0;
```

Quickbooks Online

- ▶ **CompanyId:** Set this to the ID of the company you want to connect to.

The following is an example of a Quickbooks Online connection string:

```
jdbc:quickbooksonline:OAuthClientId=xxxxxxxxxxxx;OAuthClientSecret=xx  
xxxxxxxxxx;CompanyId=xxxxxxxxxxxxxx;  
CallbackURL=http://localhost/;InitiateOAuth=GETANDREFRESH;Timeout=0  
;
```

ServiceNow

- ▶ **Password:** Set this to your password.
- ▶ **Username:** Set this to your username.
- ▶ **Instance:** Set this to your instance.

The following is an example of a ServiceNow connection string:

```
jdbc:servicenow:InitiateOAuth=GETANDREFRESH;Instance=xxxxxxxxxx  
;OAuthClientId=xxxxxxxxxxxx;  
OAuthClientSecret=xxxxxxxxxx;Username=xxxxxxxxxx;Password=xxxxxx  
xxxxxxxxxx;
```

Xero

- ▶ **XeroAppAuthentication:** Set this to the type of your application. Allowed values: PUBLIC or PARTNER. Default: PUBLIC.

The following is an example of a Xero connection string:

```
jdbc:xero:OAuthclientId=xxxxxxxxxxxxxx;OAuthClientSecret=xxxxxxxxxx  
xxx;  
InitiateOAuth=GETANDREFRESH;XeroAppAuthentication=PUBLIC;
```

YouTube Analytics

- ▶ **ChannelId:** Set this to the YouTube Analytics Channel (Profile). This can be set to the Id of the channel. If not specified, MINE will be used.
- ▶ **ContentOwnerId:** Set this to the Id of the content owner.

The following is an example of a YouTube Analytics connection string:

```
jdbc:youtubeanalytics:OAuthClientId=xxxxxxxxxxxxxxxxxxxxxx;  
OAuthClientSecret=xxxxxxxxxxxxxx;InitiateOAuth=GETANDREFRESH;C  
allbackURL=http://localhost;
```

Working with Data

Searching for Tables and Fields

Through the ElastiCube Search tool, you can locate field tables and fields.

In the Search tool, the results returned are filtered by the characters you enter into the **Search** field. When you click any of the results, the table or field is highlighted in the ElastiCube Manager.

To find a table or field in your ElastiCube:

1. At the top-right of the schema, click the magnifying glass icon.
2. In the search box, begin typing the letters or name of the table/field you are searching. Results are displayed before the **Search** field.
3. Click on a result to highlight the relevant field or table.

Creating and Removing a Relationship between Tables

Relationships specify the connection between tables and enable you to pull data together in meaningful ways. For example, order information is more useful when you know which customer placed each order.

However you don't need to store both the customer and order information in the same table. Customer and order data can be stored in two related tables with a relationship specified between the two tables to view each order and its corresponding customer information.

In practical terms relationships allow you to query and combine data from multiple tables in your dashboard.

To manually create a relationship between tables in the ElastiCube Manager:

1. Locate the field within each table to specify the relationship.
For example, two tables may both have customer ID.
2. Ensure both fields that you are using to create the relationship are of the same type, for example decimal fields.

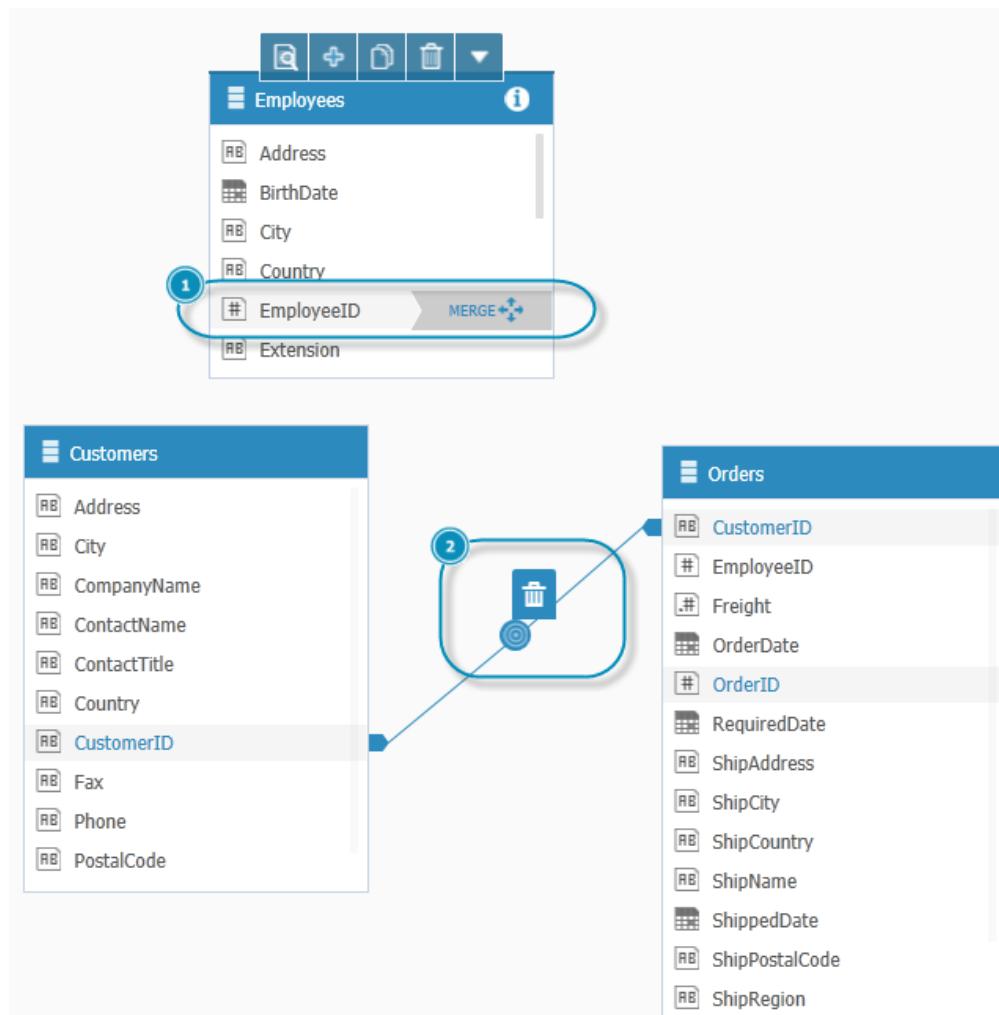
To check or change the field type, click the relevant field and a menu will appear listing the field type. Click on the drop down menu to change the field type.

Make sure the data contained within the field is compatible with the field type. For example, text data needs to have a field type of text.

3. On the right side of the field click and hold down the 'merge icon' and simultaneously drag the link to the corresponding table until the mouse cursor is over the related field.
The link will become green and a 'Drop to merge' message will appear when a relationship can be created. Let go the mouse button and a link between the two fields will appear.

To remove an existing relationship:

1. Click on any part of the existing relationship. The line will become highlighted.
2. An icon will appear, click **Click to remove current selection**.



Item Name	Description
1.	Manually create relationship
2.	Remove existing relationship

Previewing Data from a Table

You can view the data contained in a table. This includes custom fields, tables and ETL processes.

To review the underlying table data:

1. Hover over the table name until the preference icons appear.

2. Click the Show table data icon (#1 in the image below).

Note: To view underlying table data, you will first need to build your ElastiCube.

Item Name	Description
1.	Show table data
2.	Add New Custom Field
3.	Duplicate Table
4.	Delete Table
5.	Additional Preferences

Managing Tables and Fields

The ElastiCube Manager makes it easy to view and modify table and field details.

This section describes the options available for tables and fields.

Table Options

To review table metadata:

- ▶ Hover over the information icon next to the table name.
the following information is displayed:
 - Table name
 - Database Name
 - Server Address
 - Source Type (Excel, MySQL, Google AdWords, etc.)
 - Number of rows

To rename a table:

1. Double-click the header of the table that you want to rename.

2. Type in a new name or use the cursor keys to edit specific components of the existing table name.
3. Click away from the table to save any changes.

To duplicate a table:

1. Hover over the table name until the preference icons appear.
2. Click the duplicate table icon (#3 in the diagram below).

To delete a table:

Delete a table when a data source is no longer available or required. You will need to rebuild the ElastiCube for the changes to apply.

1. Hover over the table name until the preference icons appear.
2. Click the remove table icon (#4 in the diagram below).

To add a custom field to your table:

1. Place the mouse cursor above the table name until the preference icons appear.
2. Click the add new custom field icon (#2 in the diagram below).

To preview data:

1. Hover over the table name until the preference icons appear.
2. Click the preview icon (#1 in the diagram below).

To make your table invisible:

You can make your table invisible, for example, if you're using data from the table in another table, and do not need the table fields visible to the dashboard designer.

1. Hover over the table name until the preference icons appear.
2. Click the Additional Preferences icon (#5 in the diagram below).

3. Clear the **Visible** option.

Image Name	Description
1.	Preview
2.	Add New Custom Field
3.	Duplicate Table
4.	Delete Table
5.	Additional Preferences

Note: Tables with accumulative build settings are highlighted with a dark blue color.

Field Options

To view options available for fields, click on the field name or icon.

The following options are available:

- ▶ To edit the field name, click on the pencil icon.
- ▶ **Type:** Select the field type, for example, integer, date-time, or text.
- ▶ **Import:** Clear this option if you do not want the data associated with this field imported as part of the ElastiCube build.
- ▶ **Invisible:** Select this option if you want to import the data associated with this field, but do not want this field available to dashboard designers and viewers.
- ▶ **Indexed:** Indexes the field data for quicker processing in subsequent builds.
- ▶ **Duplicate:**Duplicates the field in the table. This is useful if you need the same field with some modifications.

Handling Relationship Cycles

Good relationships between tables are the key to pulling in data in logical ways. ElastiCubes are usually created from more than one table. To perform calculations on fields that reside in separate tables, you must define a relationship between the tables so the calculation mechanism knows how to navigate from one field to another.

When a single possible path exists between two fields, there is no issue. For example, performing a query involving the Customer Name and Amount field on the ElastiCube schema below can yield only one result, since there is only one possible path leading from the Customer Name field to the Amount field.

Relationship Cycles

In some cases, more than one path exists leading from one field to another. Sometimes this is due to poor database design, and other times it is just a realistic necessity due to the way the data is structured. In such cases, there is no absolute way to determine the required path to take to calculate results.

For example, in the ElastiCube schema below, the Customer Name and Amount fields now have multiple paths between them:

- ▶ Customer ---> Sales History (over the Customer ID field)
- ▶ Customer ---> Commercial (over the Customer ID field) ---> Sales History (over the Business ID field)
- ▶ Customer ---> Private (over the Customer ID field) ---> Sales History (over the Customer ID field)

Automatically Dealing with Relationship Cycles

Theoretically, any one of these possible paths could be the correct one in terms of the results you are trying to calculate. When the calculation engine encounters numerous possible paths, it picks the shortest path, which will have less impact performance-wise on query processing. Below are some general guidelines.

- ▶ The calculation engine will always prefer paths that have no many-to-many relationships. If all possible paths contain many-to-many relationships, the path with the least number of this type of relationship is preferred.
- ▶ Similarly, the engine will prefer one-to-one relationships over one-to-many relationships.
- ▶ If at the end of the first selection process more than one possible path still exists, the engine will prefer the path containing the least number of tables.
- ▶ Finally, if numerous paths are still possible, one is picked at random.

Breaking Relationship Cycles in the ElastiCube Manager

To ensure the calculation engine takes the path you want it to take, the cycle must be broken. There are a few techniques to do this depending on the required results.

Removing Relationships

One option is to remove one of the relationships in the cycle. For example, removing the Customer ID relationship from the Sales History table will eliminate the cycle. This approach is only applicable when the removed relationship is not used by other queries.

Duplicating Table Elements

Duplicating table elements is another method to break relationship cycles. In the example below, the Sales History table was duplicated and renamed *Private Customers Sales History*. In addition, the Customer ID relationship was removed from the original Sales History table. This results in the creation of two separate Amount fields, one for commercial customers residing in the original Sales History table, and one for private customers residing in the new Private Customers Sales History. As the Customer ID relationship was removed from the original table, whichever field you use will determine the correct calculation path.

Breaking Relationship Cycles in the Sisense Web App

You can also use the Sisense web app to invoke a specific path to be used during query execution (as opposed to the solutions mentioned above that are defined on the schema level, in the ElastiCube Manager).

Each widget in Sisense has an associated data layout. If you have more than one path connecting corresponding fields between two tables, you can enforce a specific path by filtering the widget, and selecting just the field that you want to use. For example if the *Commerce* table is connected to *Brands* via the *Brand* and *Brand ID* fields, then add a filter to the widget and select *Brand ID* to use that path in the query.

Many-to-Many Relationships

In databases, a Many-to-Many relationship exists when the value in each field used to create a relationship between tables is included multiple times in each table.

The problem with Many-to-Many relationships is that it can cause duplications in the returned datasets, which can result in incorrect results and might consume excessive computing resources. This section provides solutions and workarounds to common scenarios with many-to-many relationships.

Example: A hotel may have a table with reservation data and a table with payment data. Both tables include the name of the guest. A guest can have multiple reservations under their name as well as multiple payments recorded on their name. If a relationship exists between the reservation and payment tables based on the guest's name, a many-to-many relationship is created, as the guest's name appears multiple times in each table.

Summary of relationship types

- ▶ **One-to-One relationship** - In this scenario both sides of the relationship have unique values for every row.
- ▶ **One-to-Many relationship** - In this scenario one side of the relationship will contain unique values for every row, but the other side of the relationship will contain duplicate values for any or all of the corresponding values in the first table.
- ▶ **Many-to-Many relationship** - In this scenario, both sides of the relationship will hold duplicated values, causing excessive calculations for every query run against it.

There are several methods to resolve and bypass a many-to-many relationship. The solution depends on the business model and the logic of the business questions at hand. The following solutions differ by business logic and the schema at hand; each solution can be applied to each schema respectively.

The following sections cover:

- ▶ Testing your schema to see if it includes many-to-many relationships
- ▶ Understanding which scenario best fits your current schema
- ▶ According to your schema logic, applying the respective solution

To check if a relationship is Many-to-Many, you need to check the cardinality of the relationship, and determine the number of unique and duplicate values on each side of the relationship.

Testing if a Relationship is Many-to-Many

Many-to-Many relationships occur when two tables are joined on a field containing duplicate values on both tables. For example the same guest may have multiple reservations and multiple payments at a hotel, thus joining on the guest between the reservation and payment table would result in a M2M relationship.

When testing, if you get the same value for both the unique and duplicate values, then there is no duplication, and this will either be a One-to-Many or a One-to-One relationship. If the number of duplicate values is larger than the number of unique values, then this side of the relationship has duplicated values, and you will need to investigate the other side of the relationship. If the other side of the relationship

yields unique values, this is a one-to-many relationship. If not, you have a many-to-many relationship.

Use the following SQL statement to test for potential M2M relationships:

1. In the Elasticube Manager, open the relevant ecube file.
2. Click **Add Data > Custom SQL Expression**.
3. Enter and adjust the SQL statement below.

```
SELECT [Do I have duplications?]

FROM (
    SELECT distinct_count(t1.col1)<>count(t1.col1) AS [Do I have
duplications?]

    FROM [Table1] t1

    UNION all

    SELECT distinct_count(t2.col2)<>count(t2.col2)

    FROM [Table2] t2 AS temp

    GROUP BY [Do I have duplications?]
```

4. In the top right of the expression editor window, click **Parse SQL Expression**. If the expression parses successfully, click **Preview result table**.
5. If the returned result is 'True' in both lines, a many-to-many relationship exists, and will need to be considered in the ElastiCube design.

If the two values are equal, all guest IDs appear only once, making all values unique. Even if the other side of the relationship has duplicate

values for guest ID, this is till a One-To-Many relationship, where the unique values are on the reservations side, and the duplicate values are on the Payments side.

If there are more than two tables connected to this relationship, that is, if there are more than two tables merged on the same field, a few more options exist. The solution for the single many-to-many relationship will be a sub-problem of this scenario. In this case, you'll need to run the test on every table to see the uniqueness or duplication of the merged fields.

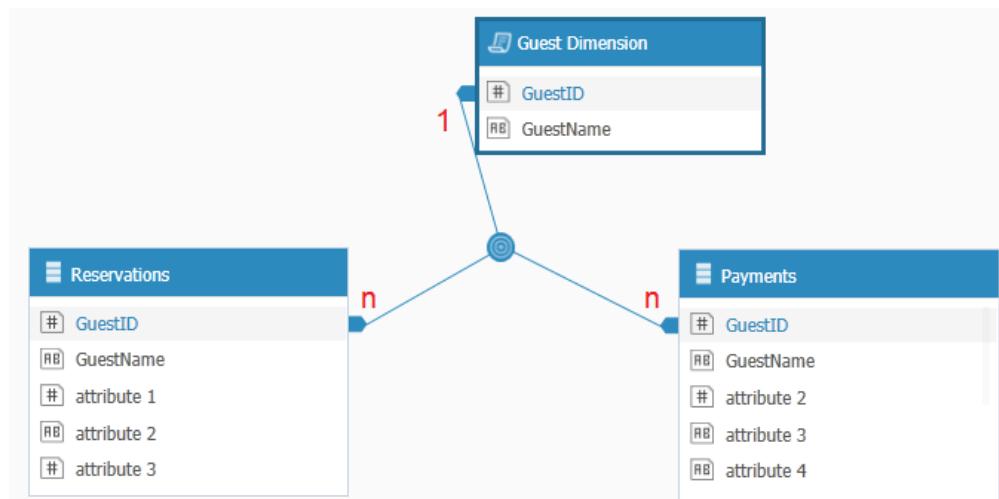
Two Tables with One Relationship

This section describes two possible workarounds when you have a schema that includes two tables with one relationship:

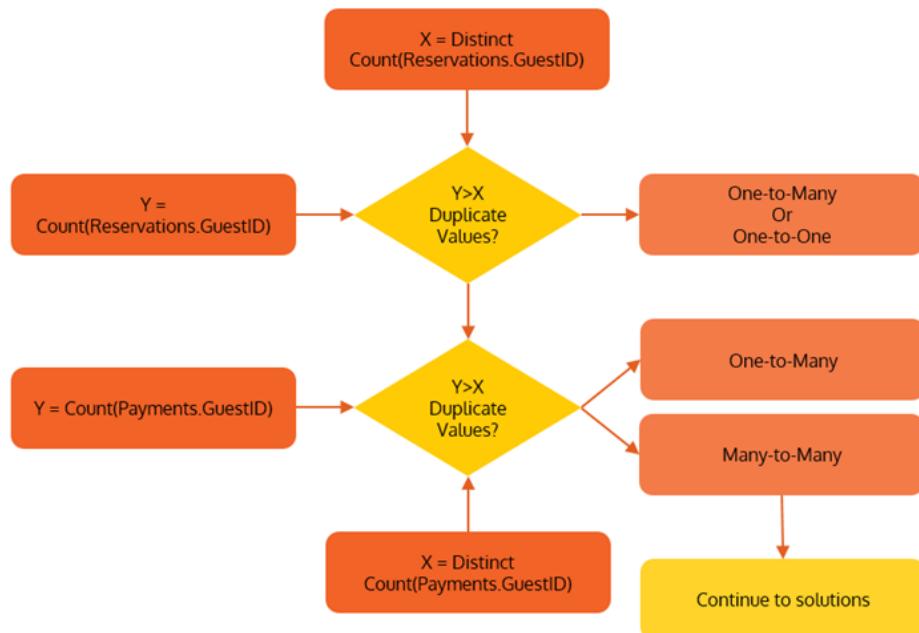
- ▶ Two Separate One-to-Many Relationships
- ▶ Creating an Aggregated Table

Two Separate One-to-Many Relationships

The direct solution for such a problem is to break this relationship into two separate one-to-many relationships.



You can use the following diagram to understand the logic behind this testing:



1- Create a custom SQL expression in the Elasticube. In the expression of this table select all the individual values for the identifier column from both sides. The expression should look like this:

```

SELECT * FROM
(SELECT DISTINCT r.GuestID, r.GuestName
FROM [Reservations] r
UNION
SELECT DISTINCT p.GuestID, p.GuestName
FROM [Payments] p) AS G
  
```

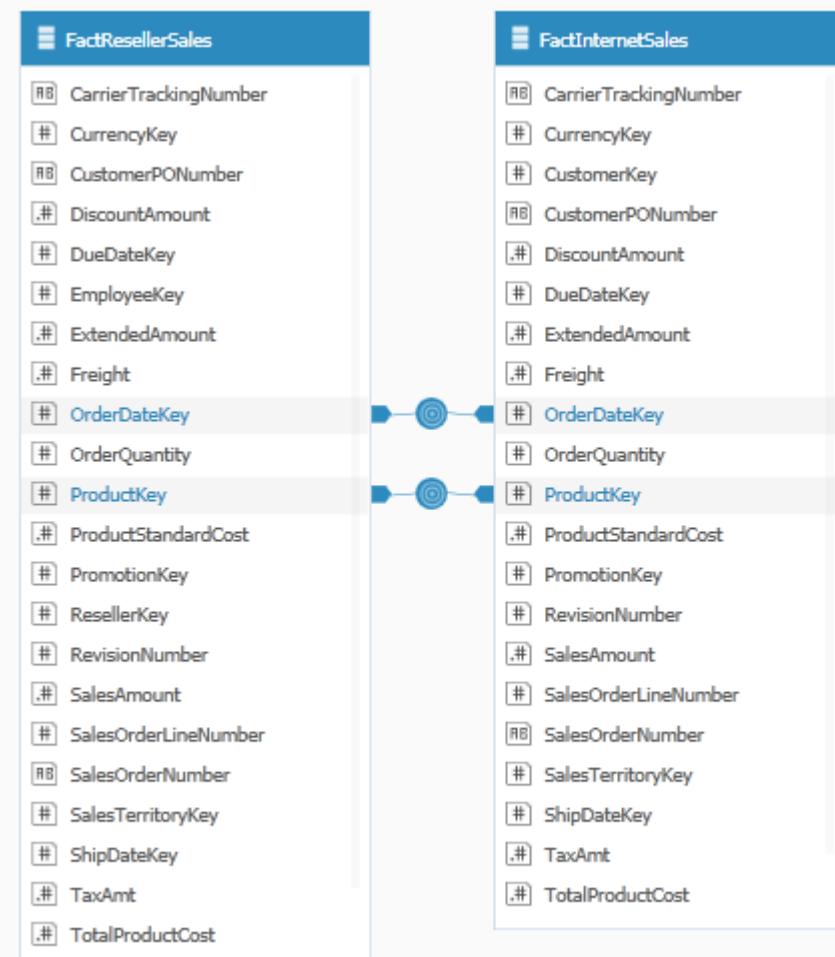
This query will take all Guest ID values from both tables, and using the UNION statement, will bring in only the unique values from both tables, making this a complete list of all distinct Guest ID values.

2 - Merge the Guest ID field from the new 'linking' table to the other two Guest ID fields from the other two tables, thus creating two One-To-Many relationships.

You can now use this Guest ID field as the rows or axes elements of a widget, pulling in the unique values from the new Guest Dimension, with measures from the two other tables. See image above.

Creating an Aggregated Table

In situations where you have more than one fact table (a Fact table is a primary table containing the measures or fields used for calculations in the dashboard) in the Elasticube, there are several situations when an aggregated table can resolve a many-to-many relationship.



Two fact tables

Assuming you want to segment your data according to a few different dimensions, creating relationships directly between these fields can and will create many-to-many relationships in one of two ways, according to the schema:

- ▶ Both tables don't hold unique values, and all values from one table are held in the second table. In this scenario either a linked dimension (as described in the first solution - Two Separate One-to-Many Relationships) or an aggregated table can be created which will hold all the unique values and the

desired calculations for one of the tables. To create an aggregate table, create a custom SQL expression and aggregate values from the table that includes all values; its own, and the subset present in the other table with the following expression:

```
SELECT i.OrderDateKey, i.ProductKey, sum(i.DiscountAmount),  
sum(i.SalesAmount), avg(i.UnitPriceDiscountPct)  
  
FROM [FactInternetSales] i  
  
GROUP BY i.OrderDateKey, i.ProductKey
```

```
SELECT i.OrderDateKey, i.ProductKey, sum(i.DiscountAmount), sum(i.SalesAmount), avg(i.UnitPriceDiscountPct)  
FROM [FactInternetSales] i  
GROUP BY i.OrderDateKey, i.ProductKey
```

This custom SQL expression will select the distinct OrderDateKeys and their corresponding ProductKeys from the FactInternetSales, grouped by these fields, together with single value aggregations for the different fields, in this case, Discount Amount, Sales Amount and the average unit Price discount. After merging the OrderDateKey and Product Key to the two other tables, you will be able to pull the values from this new table into the rows or axes panel of a widget in the Sisense Web Application with measures and additional aggregations from the two other tables.

Note: The non-aggregated table needs to be a subset in terms of the primary fields from the aggregated table.

- ▶ Both tables don't include unique values, and there are different values for several fields in both the tables. Resolving this scenario incorporates both solutions mentioned above. In this scenario, create an aggregated table and a dimension table

(both described above). The final resolution should look like this:



Two Fact tables with a date dimension table and an aggregative Products table

More than Two Tables with More than One Relationship

This section provides two possible workarounds when you have a schema that includes more than two tables with more than one relationship:

Options include:

- ▶ Using the Lookup function

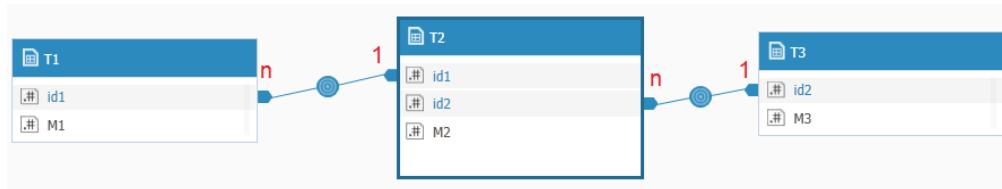
- ▶ Concatenating two tables into one

Using the Lookup Function

In most scenarios you will aggregate values according to a given ID from the unique side of the relationship to the duplicate side.

However in specific cases it'll be vice versa.

For example in the following scenario, in which we have three tables, and between them two one-to-many relationships, this can potentially create a many-to-many relationship, if you query the two leaf tables. This means that the query result table will have multiple rows which won't be distinguishable one from the other.



Two consecutive M-to-M relationships

Using the Lookup Function, you can import values from a remote table by matching values in a different column. This will create a new column in the table where you want to perform an aggregation of a given field(s), with the matching value of the identifying field from the other table. Taking the example of tables T1, T2 and T3, we'd like to run a query that will display aggregations from the duplicate IDs from T1, with a measure from T3. If you run the query as is, you will get multiple values for the query's result set, and we will not be able to run this aggregation. To resolve this, use the Lookup function to import the values from T3 into T2 and then re-run the query only on tables T1 and T2. Using the lookup function, available in the 'Miscellaneous Functions' in the custom SQL editor, you can import

the values of 'M3' from the 'T3' table into the 'T2' table. Create a new custom column, and use the Lookup function to import the values of attribute, In this case, the Lookup function should look like this:

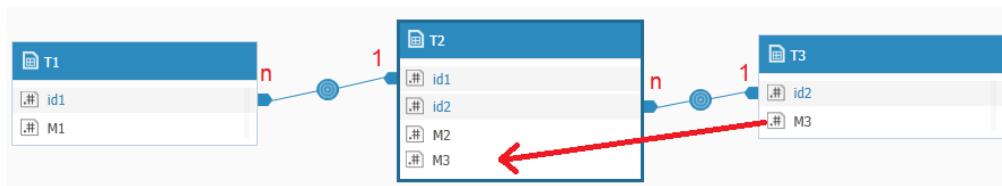
`Lookup([T3],[T3].[M3], [T2].id2,[T3].id2)`

Running this statement in table T2 will import the matching values of M3 from T3 according to the matching results in ID2 between the two tables.

`LOOKUP(remote_table,remote_result_column,current_match_column, remote_match_column)`

Matches the current value with another value from a remote table.

The result will be the value in `remote_result_column` for which the corresponding `remote_match_column` equals the `current_match_column`.

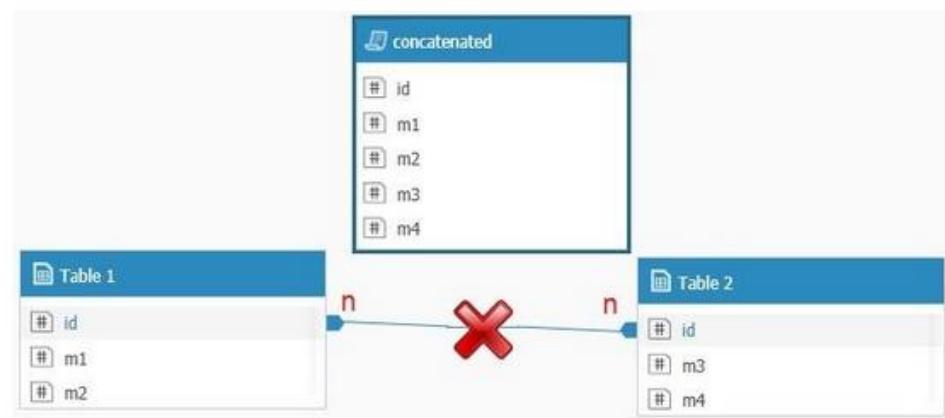


Two consecutive M-to-O relationships after Lookup fix

Concatenating Two Tables into One

Assuming you have two separate tables with duplicate ID values in each, and each including different columns for each ID, you can create a new table including all values for every ID, and pull the aggregations from this new table.

Note that the two original tables; Table_1, Table_2 have different columns.



Concatenating tables

Using the following SQL statement, you can import the data from both tables, with the IDs and the columns respectively:

```
SELECT s.id AS id, s.m1, s.m2,ToInt( NULL ) m3 ,ToInt( NULL ) m4
FROM [Table 1] s
UNION
SELECT t.id,ToInt( NULL ),ToInt( NULL ),t.m3,t.m4
FROM [Table 2] t
```

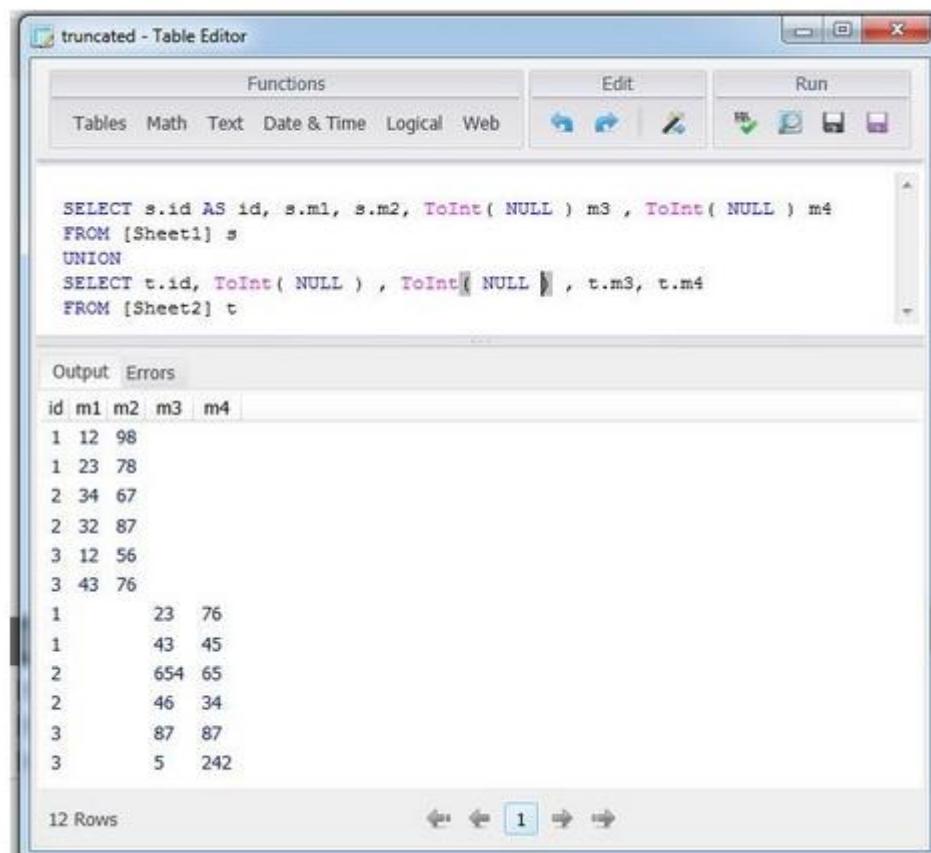
This will create a table with five columns:

1. Id
2. M1 (from table_1)
3. M2 (from table_1)
4. M3 (from table_2)
5. M4 (from table_2)

The values missing from each table respectively will be NULL's which will result in the following table.

Table 1 - Table		
	Output	Errors
id	m1	m2
1	12	98
1	23	78
2	34	67
2	32	87
3	12	56
3	43	76

Table 2 - Table		
	Output	Errors
id	m3	m4
1	23	76
1	43	45
2	654	65
2	46	34
3	87	87
3	5	242



The screenshot shows the Sisense Table Editor interface. At the top, there are two tables labeled "Table 1 - Table" and "Table 2 - Table". Below them is a query editor window titled "truncated - Table Editor" containing the following SQL code:

```

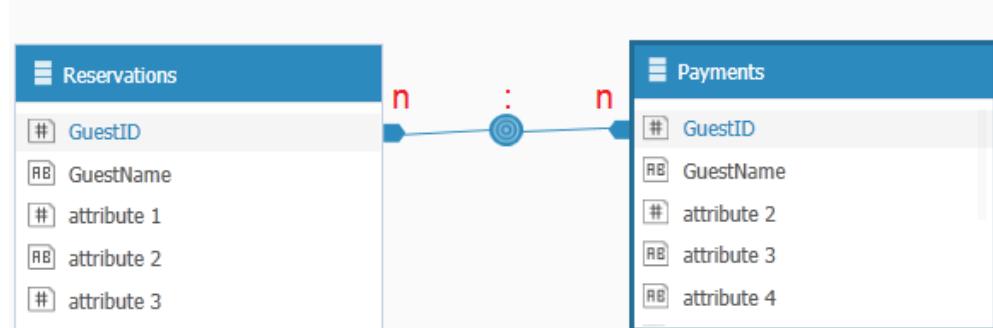
SELECT s.id AS id, s.m1, s.m2,ToInt( NULL ) m3 ,ToInt( NULL ) m4
FROM [Sheet1] s
UNION
SELECT t.id,ToInt( NULL ) ,ToInt( NULL ), t.m3, t.m4
FROM [Sheet2] t
    
```

The bottom half of the window displays the resulting concatenated table, which has 12 rows. The columns are labeled "id", "m1", "m2", "m3", and "m4". The data is as follows:

id	m1	m2	m3	m4
1	12	98		
1	23	78		
2	34	67		
2	32	87		
3	12	56		
3	43	76		
			23	76
			43	45
			654	65
			46	34
			87	87
			5	242

The status bar at the bottom indicates "12 Rows".

Concatenated table - result set

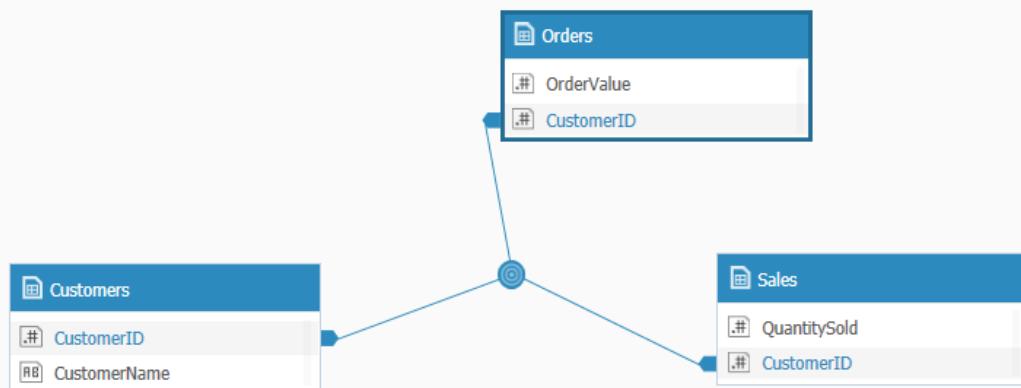


Chasm and Fan Traps

Chasm and fan traps should be avoided when building your ElastiCube schemas.

Chasm Traps

A chasm trap occurs when two “Many-to-One” joins converge on a single table, and the query includes measures from both leaf tables. As a result multiple rows are returned from the tables when processing the query.



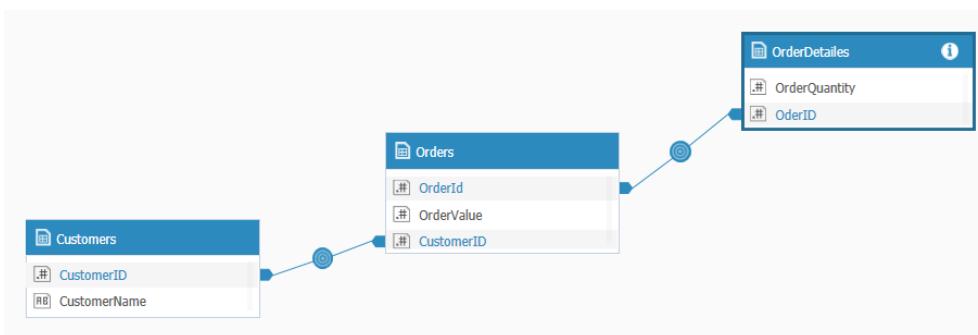
If you were to calculate both measures (Qty and Value) simultaneously, like in the following example, the values for Customers will be multiplied due to the inner join between the leaf tables, and the results may be incorrect:

```
SELECT [Customers].CustomerName,
       sum ([Orders].OrderValue) AS Value,
       sum ([Sales].QuantitySold) AS Qty
  FROM [Customers]
 JOIN [Orders] ON [Customers].CustomerID=[Orders].CustomerID
 JOIN [Sales] ON [Customers].CustomerID=[Sales].CustomerID
 GROUP BY [Customers].CustomerName
```

Output Errors		
CustomerName	Value	Qty
Jason	1800	180
Daniel	600	60

Fan Traps

A fan trap occurs when two “many-to-one” joins follow one another in master-detail form (OrderDetails), and the query includes a measure from both the leaf table (OrderDetails) and its immediate master (Orders).



If you try to aggregate both measures simultaneously (using the query below), you will probably get incorrect results:

```

SELECT [Customers].CustomerName,
       sum ([Orders].OrderValue) AS Value,
       sum ([OrderDetails].OrderQuantity) AS Qty
  FROM [Customers] JOIN [Orders] ON [Customers].CustomerID=[Orders].CustomerID
  join
 [OrderDetails] ON [Orders].OrderId=[OrderDetails].OrderID
 GROUP BY

```

CustomerName	Value	Qty
Jason	700	90
Daniel	400	60
Mike	400	60

The “Qty” measure, corresponding to the leaf measure table (OrderDetails) is calculated correctly, *but* the “Value” measure, corresponding to the measure held in its master (Orders), is not. This is because *we get the “Value” of every OrderID, which may inflate the expected results.*

The web application translation module *separates the calculations, and unions the results* by generating a query for each of the measures’ paths (path in terms of tables to go by). Then the web application translation module will group all the measures with the same tables’ paths into one query and union it with all the other “same path measures” with different paths.

The described “Chasm Trap” can be prevented like this:

```

SELECT [Customers].CustomerName, sum ([Orders].OrderValue) AS Value, 0 AS Qty
  FROM [Customers] JOIN [Orders] ON [Customers].CustomerID=[Orders].CustomerID
 GROUP BY [Customers].CustomerName
union
SELECT [Customers].CustomerName, 0 AS Value, sum ([Sales].QuantitySold) AS Qty
  FROM [Customers] JOIN [Sales] ON [Customers].CustomerID=[Sales].CustomerID
 GROUP BY [Customers].CustomerName

```

CustomerName	Value	Qty
Jason	600	0
Daniel	300	0
Mike	300	0
Jason	0	60
Daniel	0	30

And the “Fan Trap” will be prevented this way:

```
SELECT [Customers].CustomerName, 0 AS Value, sum ([OrderDetailes].OrderQuantity) AS Qty
FROM [Customers] JOIN [Orders] ON [Customers].CustomerID= [Orders].CustomerID
join
[OrderDetailes] ON [Orders].OrderId=[OrderDetailes].OderID
GROUP BY
[Customers].CustomerName
union
SELECT [Customers].CustomerName,sum ([Orders].OrderValue) AS Value, 0 AS Qty
FROM [Orders]
JOIN [Customers] ON [Orders].CustomerID=[Customers].CustomerID
GROUP BY CustomerName
```

Output Errors		
CustomerName	Value	Qty
Jason	0	90
Daniel	0	60
Mike	0	60
Jason	600	0
Daniel	300	0
Mike	300	0

Transforming and Enriching Data

Adding a New Custom Field

The ElastiCube Manager makes it easy for you to add new fields to existing tables. This can be useful if you need to combine data from different existing fields, and when you need to cleanse and prepare data. The new fields also provide designers with additional fields to use as is, or as a basis for even more advanced calculations in their widgets. You can use SQL to customize the values contained within the custom fields.

To add a new field:

1. Hover over the table to which you want to add a field.
2. Place the mouse cursor above the table name until the preference icons appear.
3. Click the Add new custom field icon.
4. The new field will be added to the table, and the field's settings window appears, with the following options:
 - ▶ **Name:** Click the field name in the settings window to rename or edit the field name.
 - ▶ **Type:** Select the field type from the drop-down list. Field types should be based on the data stored in the new field. For example text data must be set to *text* in the **Type** field, while date-time data must be set to **Date-Time**.
 - ▶ **Expression:** You can use SQL to customize the data contained in the new field. Click **Edit** to edit and view the field expression. See [Defining and Editing Field Formula](#) for further details on developing field expressions.
5. To delete a custom field, click the field and click **Remove**.

Field data can be converted to a different type using functions. For example, a number can be converted to a text field. See the function reference for more details.

Defining and Editing Field Formulas

In the ElastiCube Manager, you can use SQL expressions to define custom fields. The steps below explain how to edit a custom field expression.

1. Click the custom field in the table to display the custom field options, and click **Edit**.
2. In the Table Editor, you can enter SQL statements to access tables and fields that exist in the ElastiCube Manager schema. See also [SQL Reference](#) and [Function Reference](#).
3. Once you complete an SQL statement, click **Parse SQL Expression** to validate the statement.
4. To view results based on the SQL statement, click the **Preview result** icon. **Note:** To preview the results, the ElastiCube should have been built at least once.
5. To save changes or save and exit click on the top right icon.

Tips:

- ▶ Press Ctrl + Space to complete an SQL statement.
- ▶ To make it easier to access table fields, give the table name an alias and reference the alias to bring up the related fields. For example Select A. From table as A.
- ▶ You can add comments in expressions by using the `--` for a single line or `/*...*/` for blocks.

Adding a New Custom Table

With the ElastiCube Manager, you can easily create custom tables, which combine existing data. This is useful for preparing and cleansing data. Follow the steps below to learn how to add custom tables using SQL.

To create a custom table:

1. Click **Add Data** in the top menu of the ElastiCube Manager.
2. Under the **Custom** category, select **Custom SQL Expression**.
3. In the window that opens, enter the name of the custom table, and click .
4. In the Table Editor, you can enter SQL statements to access tables and fields that exist in the ElastiCube Manager schema. See also [SQL Reference](#) and [Function Reference](#).
5. Once you complete an SQL statement, click on the **Parse SQL Expression** icon to validate the statement. 
6. To view results based on the SQL statement, click the **Preview result table** icon .

Note: You can save an incomplete or invalid SQL expression, however you will not be able to build the custom expression. At least one build including the relevant base table must be completed in order to preview results.

7. To save changes or save and exit, click on one of the Save icons.
 - Press Ctrl + Space to complete an SQL statement.

- To make it easier to access table fields, give the table name an alias and reference the alias to bring up the related fields. For example *SELECT A. FROM A.*
- You can add comments in expressions by using the '--' for a single line or /./. for blocks.

Defining and Editing SQL Statements

The ElastiCube Manager makes it easy to edit custom tables. You can use SQL to customize the fields contained within a custom table.

1. Click the custom table you want to edit or add an SQL expression.
2. Place the mouse cursor above the table name until the preference icons appear. Click the **Edit table SQL expression** icon.
3. The Table Editor will open where you can enter SQL statements to access tables and fields that exist in the ElastiCube Manager schema. See related topics for more details on SQL and Function References.
4. Once you complete an SQL statement click on the top right icon **Parse SQL Expression** to validate the statement.
5. To view results based on the SQL statement, click on the top right icon **Preview result table**. (Note: At least one build including the relevant base table must have been completed in order to preview results.)
6. To save changes or save and exit, click on the top right icon.

 **Tips:**

- ▶ Press Ctrl + Space to complete an SQL statement.

- ▶ To make it easier to access table fields, give the table name an alias, and reference the alias to bring up the related fields. For example Select A. From table as A
- ▶ You can add comments in expressions by using the '--' for a single line or /./. for blocks.
- ▶ A good reference on compatible SQL commands is available on [Microsoft MSDN](#).

Previewing Results

You can preview results as follows:

To view results before connecting to a table:

1. Click **Add Data** in the top menu of the ElastiCube Manager.
2. Select the relevant connector.
3. Enter your login details and connect to the source data.
4. All tables and views associated with the database will appear in a new window. To preview data contained in a particular table, highlight the table or view in the list and click the preview pane below. To preview the table, select the **Preview** checkbox.

To view results in a Custom Table:

1. Click the custom table that you want to view.
2. Place the mouse cursor above the table name until the preference icons appear.
3. Click the **Edit table SQL expression** icon.
4. To view results based on the SQL statement, click on the top-right icon Preview result table.

Note: At least one build including the relevant base table must be completed in order to preview results.

To view results in a Custom Field:

1. Click the custom field until the fields setting menu opens and then click **Edit**.
2. On the top right icon Parse SQL Expression to validate the statement.
3. To view results based on the SQL statement click on the top right icon Preview result table.(**Note:** At least one build including relevant base table must be completed in order to preview results.)

SQL Reference

You can use SQL to create new tables and fields in the ElastiCube Manager.

SQL Structure

```
SELECT FieldName(s), Function(),
FROM TableName 1
INNER JOIN\ LEFT JOIN\RIGHT
JOIN\FULL JOIN TableName 2
ON TableName 1. JoinField
= TableName 2. JoinField
WHERE FieldName Condition
AND\OR FieldName Condition
GROUP BY FieldName(s)
ORDER BY FieldName(s)
HAVING FieldName(s) Condition
```

SQL Example

```
*   SELECT CustomerName,
ContactEmail,Count(Orders)
FROM Customer
INNER JOIN Order
ON Customer.CustomerID =
Order.CustomerID
WHERE Order.OrderId BETWEEN 10
AND 100
AND Customer.CustomerName IN
('John','Mary','David')
OR
Customer.CustomerLastName LIKE
'Harrison'
GROUP BY Customer.CustomerName
ORDER BY
Customer.CustomerLastName
HAVING Count(Orders) > 3
```

Basic SQL Syntax Guide

SELECT STATEMENT

SELECT	SELECT column_name(s) FROM table_name
SELECT *	SELECT * FROM table_name
SELECT DISTINCT	SELECT DISTINCT column_name(s) FROM table_name

FUNCTIONS AFTER SELECT CLAUSE

FUNCTIONS	AVG() - Returns the average value COUNT() - Returns the number of rows MAX() - Returns the largest value MIN() - Returns the smallest value SUM() - Returns the sum
-----------	---

TABLE JOIN FUNCTIONS

INNER JOIN	SELECT column_name(s) FROM table_name1 T1 INNER JOIN table_name2 T2 ON T1.column_name=T2.column_name
LEFT JOIN	SELECT column_name(s) FROM table_name1 T1 LEFT JOIN table_name2 T2 ON T1.column_name=T2.column_name
RIGHT JOIN	SELECT column_name(s) FROM table_name1 T1 RIGHT JOIN table_name2 T2 ON T1.column_name=T2.column_name
FULL JOIN	SELECT column_name(s) FROM table_name1 T1

	FULL JOIN table_name2 T2 ON T1.column_name=T2.column_name
AS (alias)	SELECT column_name AS column_alias FROM table_name or SELECT column_name FROM table_name AS table_alias
WHERE	SELECT column_name(s) FROM table_name WHERE column_name operator value

FUNCTIONS AFTER THE WHERE CLAUSE

AND / OR	SELECT column_name(s) FROM table_name WHERE condition AND OR condition
BETWEEN	SELECT column_name(s) FROM table_name WHERE column_name BETWEEN value1 AND value2
IN	SELECT column_name(s) FROM table_name WHERE column_name IN (value1,value2,...)
LIKE	SELECT column_name(s) FROM table_name WHERE column_name LIKE pattern

ORDERING AND GROUPING AFTER WHERE CONDITIONS

GROUP BY	SELECT column_name, aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_name
-----------------	---

ORDER BY	<pre>SELECT column_name(s) FROM table_name ORDER BY column_name [ASC DESC]</pre>
HAVING	<pre>SELECT column_name, aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_name HAVING aggregate_function(column_name) operator value</pre>

CLAUSES TO COMBINE TABLES

UNION	<pre>SELECT column_name(s) FROM table_name1 UNION SELECT column_name(s) FROM table_name2</pre>
UNION ALL	<pre>SELECT column_name(s) FROM table_name1 UNION ALL SELECT column_name(s) FROM table_name2</pre>

Function Reference

Mathematical Functions

You can use mathematical functions in custom tables and fields.

See Adding a New Custom Table and Defining and Editing Field

Formula for further details. Below are explanations of available mathematical functions.

ACOS(number)

Returns the angle, in radians, whose cosine is the given number. Also referred to as arccosine.

ASIN(number)

Returns the angle, in radians, whose sine is a given number. Also referred to as arcsine.

ATAN(number)

Returns the angle in radians whose tangent is a given number. Also referred to as arctangent.

CEILING(number)

Returns the smallest integer that is greater than, or equal to, the a given number.

COS(number)

Returns the trigonometric cosine of a given angle, in radians.

COSH(number)

Returns the hyperbolic cosine of a given number.

COT(number)

Returns the hyperbolic sine of a given number.

DIV(number1, number2)

Returns the number of times number2 fits completely in number1.

EXP(number)

Returns the exponential value of a given number.

FLOOR(number)

Returns the largest integer less than or equal to the given numeric expression.

ISINFINITY(number)

Returns -1 if the number evaluates to negative infinity, 1 if it evaluates to positive infinity, and 0 otherwise.

ISINTEGER(number)

Returns whether a given number is a whole number.

LOG(number)

Returns the natural logarithm of a given number.

LOG10(number)

Returns the base-10 logarithm of the given float expression

MAXVAL(number1,number2)

Returns the highest of two numbers.

MINVAL(number1,number2)

Returns the smallest of two numbers.

MOD(number1,number2)

Returns the remainder of dividing number1 by number2 (modulo).

PI()

Returns the constant value of the mathematical PI.

POWER(number1,number2)

Returns the value of the given number raised to a specified power.

RAND()

Returns a random number between 0 and 32767

RANK()

Returns the rank (i.e. row number) of the current record in a table.

RANKASC(order_by_field)

Returns the row number ordered by given field in ascending manner.

You can also sort by multiple columns: **RANKASC([partitionA],[partitionB].[partitionN],[orderByField])**

Example: RANKASC([EmployeeID],[CategoryID],[Quantity])

RANKDESC(order_by_field)

Returns the row number ordered by given field in descending manner.

You can also sort by multiple

columns: **RANKDESC([partitionA],[partitionB].[partitionN],[orderByField])**

Example: RANKDESC([EmployeeID],[CategoryID],[Quantity])

ROUND(number,precision)

Returns the given number, rounded to a specified precision.

SIN(number)

Returns the trigonometric sine of a given angle, in radians.

SINH(number)

Returns the hyperbolic sine of a given number.

SQRT(number)

Returns the square root of a given number.

TAN(number)

Returns the tangent of a given number.

TANH(number)

Returns the hyperbolic tan of a given number.

TOBIGINT(string)

Converts a string representing a valid BigInt value to BigInt type.

TODOUBLE(string)

Converts a string representing a valid double value to Double type.

TOINT(string)

Converts a string representing a valid Int value to Int type.

Date and Time Functions

You can use date and time functions in custom tables and fields.

See [Adding a New Custom Table](#) and [Defining and Editing Field](#)

[Formula](#) for further details. Below are explanations of available date and time functions.

ADDDYEARS(datetime, number)

ADDQUARTERS(datetime, number)

ADDMONTHS(datetime, number)

ADDDAYS(datetime, number)

ADDHOURS(datetime, number)

ADDMINUTES(datetime, number)

ADDSECONDS(datetime, number)

Adds a given number of years|quarters|months|days|hours|minutes|seconds to a specified date/time. An example of when to use this is when the fiscal year is not the same as the Gregorian calendar's beginning and end dates.

CREATEDATE(year,month,day)

Creates a timestamp from a given year, month and day. Time is set to midnight.

CURRENTDATE()

Returns the current date.

CURRENTTIME()

Returns the current time.

CURRENTTIMESTAMP()

Returns the current timestamp.

GETYEAR(datetime)

GETQUARTER(datetime)

GETMONTH(datetime)

GETDAY(datetime)

GETHOUR(datetime)

GETMINUTE(datetime)

GETSECOND(datetime)

Returns a number that represents the year|quarter|month|day|hour|minute|second in a given date/time.

DAYOFWEEK(datetime)

Returns the day, out of seven days in a week, represented by a given date/time.

DAYOFYEAR(datetime)

Returns the day, out of 365 days in a year, represented by a given date/time.

WEEKOFYEAR(datetime)

Returns the week, out of 52 weeks in a year, represented by a given date/time.

DAYNAME(number)

Returns the name of the day represented by the given number.

YEARDIFF(end,start)

QUARTERDIFF(end,start)

MONTHDIFF(end,start)

DAYDIFF(end,start)

HOURDIFF(end,start)

MINUTEDIFF(end,start)

SECONDDIFF(end,start)

Returns the difference in

years|quarters|months|days|hours|minutes|seconds

String and Text Functions

You can use string and text functions in custom tables and fields.

See Adding a New Custom Table and Defining and Editing Field

Formula for further details. Below are explanations of available string and text functions.

ASCII(char)

Returns the Int ascii code of a given character.

CHAR(number)

Returns a character for a given ascii code.

CONCAT(string1,string2)

Concatenates string1 and string2.

INDEXOF(string1,string2,n)

Returns the index of the n-occurrence of string2 in string1.

INSERT(string1,string2,overrides,index)

Inserts string2 into string1 at a specified index, possibly overriding a specified number characters in string1.

INSERTSTR(string1,string2,index)

Inserts string2 into string1 at specified zero-based index.

ISNULLOREMPTY(string)

Returns true if the string is null or empty.

LEFT(string,length)

Returns a substring of a specified length, starting at the first character.

LENGTH(string)

Returns the length of a given string.

LOWER(string)

Returns a lowercase version of a given string.

LTRIM(string)

Removes whitespaces from the start of the string.

REMOTEAT(string,index)

Removes a character at a specified index from a given string.

REMOVERANGE(string,index,count)

Removes a specified number of characters from a given string starting at a specified index.

REPEAT(string,number)

Returns a concatenation of a string repeated a given number of time.

REPLACEALL(string,oldstring,newstring)

Replaces all occurrences of old string with new string.

RIGHT(string,length)

Returns a substring of a specified length, end with the last character.

RTRIM(string)

Removes whitespaces from the end of the string.

SOUNDEX(string)

Returns a four-character (SOUNDEX) code to evaluate the similarity of two objects.

STRBETWEEN(string, startstring, endstring)

Returns the substring contained between startstring and endstring.

STRPARTS(string,delimiter,n)

Returns the string delimited by a specified delimiter. The search ends after the n-occurrence of the delimiter is found.

SUBSTRING(string,index,length)

Returns the substring of a specified length, starting at a specified index.

TOSTRING(object)

Converts a given value to a string.

TRIM(string)

Removes whitespaces from both ends of a string.

UPPER(string)

Returns an uppercase version of a given string.

Logical Functions

You can use logical functions in custom tables and fields. See Adding a New Custom Table and Defining and Editing Field Formula for further details. Below are explanations of available logical functions.

IFBIGINT(condition, true value, false value)

Returns the first value if the boolean evaluate is true, else returns the second value.

IFDATETIME(condition, true value, false value)

Returns the first value if the boolean evaluate is true, else returns the second value.

IFDOUBLE(condition, true value, false value)

Returns the first value if the boolean evaluate is true, else returns the second value.

IFINT(condition, true value, false value)

Returns the first value if the boolean evaluate is true, else returns the second value.

IFSTRING(condition, true value, false value)

Returns the first value if the boolean evaluate is true, else returns the second value.

INRANGE(value,start,end)

Returns true if a given value is between start and end.

ISNULL(value)

Returns true if a given value is null.

Web Functions

You can use web functions in custom tables and fields. See Adding a New Custom Table and Defining and Editing Field Formula for further details. Below are explanations of available web functions.

Note: In all circumstances, the URL must include 'http://'. For example if the URL is www.sisense.com, the syntax should be *GetFile* ('*http:// www.sisense.com*')

DomainIs: true/false DomainIs (URL,compared domain)

Get whether the given urls domain matches the given domain. For

example 'http://www.wikipedia.org' matches org, wikipedia, www.wikipedia.org.

GetFile: string **GetFile (url)**

Extract the filename portion of the path in the given url and returns it.
The filename is everything after the last slash in the path.

GetHost: string **GetHost (url)**

Extract host from the given url, for example 'www.wikipedia.org' in 'http://www.wikipedia.org'

GetPassword: string **GetPassword (url)**

Extract password from the given url.

Specified in the url following the user name, with leading colon (':') and trailing at sign (@), as in 'http://user:password@host/.'

GetPath: string **GetPath (url)**

Extract path from the given url.

Path is the url portion following the host name, for example '/wiki/Business_intelligence' in 'http://www.wikipedia.org/wiki/Business_Intelligence'

GetPort: string **GetPort (url)**

Extract port from the given url, parsed as integer, for example 123 in 'http://www.wikipedia.org:123/'

GetQuery: string **GetQuery(url)**

Extract the query from the given url.

Specified in the url after the '?' and before (optional) the '#' sign, for example 'par=val' in 'http://www.wikipedia.org/?par=val#ref'.

GetQueryParamValue (url,param name): string **GetQueryParamValue** (url,param name)

Extract param value from the given url's query, by the given param name.

Param/Value pairs when supplied, are delimited by an ampersand or a semicolon.

GetRef: string **GetRef (url)**

Extract ref from given url.

Specified in the url starting after last appearance of the '#' sign, for example 'ref' in 'http://www.wikipedia.org/?par=val#ref'

GetScheme: string **GetScheme (url)**

Extract scheme from given url.

Skips leading spaces and followed colon ':' sign, for example 'http://www.wikipedia.org/'

GetUserName: string **GetUserName (url)**

Extract user name from given url.

Specified in the url before the host name and delimited by an '@' as in 'http://user@host.'

HasHost: true\fast **HasHost (url)**

Get whether the given url specifies a host.

HasPassword: true\fast **HasPassword (url)**

Get whether the given url specifies a password.

HasPath: true\fast **HasPath (url)**

Get whether the given url specifies a path.

HasPort: true\fast **HasPort (url)**

Get whether the given url specifies a port.

HasQuery: true\fast **HasQuery (url)**

Get whether the given url specifies a query.

HasRef: true\fast **HasRef (url)**

Get whether the given url specifies a ref.

HasScheme: true\fast **HasScheme (url)**

Get whether the given url specifies a scheme.

HasUsername: true\fast **HasUsername (url)**

Get whether the given url specifies a user name.

HostIsIpAddress: true\fast **HostIsIpAddress (url)**

Get whether the given url host name is an ip address.

IsEmptyUrl: true\fast **IsEmptyUrl (url)**

Get whether the given url is empty.

IsFileUrl: true\fast **IsFileUrl (url)**

Get whether the given url is a file url.

IsSecuredUrl: true\fast **IsSecuredUrl (url)**

Get whether the given url scheme indicates a secured connection(https).

IsStandardScheme: true\fast **IsStandardScheme (url)**

Returns true if the scheme for the current url is a known 'standard' scheme.

Standard schemes have an authority and a path section (including file:).

IsValidUrl: true\fast **IsValidUrl (url)**

Get whether the given url has a valid form.

PathForRequest: true\fast **PathForRequest (url)**

Returns the path that should be sent to the server.

This is the path, parameter, and query portions of the url.

Schemels: true\fast Schemels (url)

Get whether the given url scheme portion matches the given scheme.

Miscellaneous Functions

You can use miscellaneous functions in custom tables and fields.

See Adding a New Custom Table and Defining and Editing Field

Formula for further details. The Miscellaneous Functions category currently includes the lookup function.

LOOKUP(remote_table,remote_result_column,current_match_column, remote_match_column)

The Lookup function imports a field from one table into another table, by matching two other corresponding fields from both tables. The result will be the value in remote_result_column for which the corresponding remote_match_column equals the current_match_column.

Building ElastiCubes for Advanced Business Scenarios

Introduction

This section provides examples of more advanced scenarios, and the recommended methods for implementing the required business logic.

The examples are categorized into the following data manipulation methods: Integrating, Formatting and Enhancing.

Integrating Data

Integrate and merge data from different sources into a single ElastiCube structure by identifying common keys between the different tables. Proper planning is important for merging the data; on

the one hand, you need to avoid creating unnecessary relationships, and on the other hand, make sure you don't have any many to many relationships. Examples include:

- ▶ **Creating a Common Date Selection:** Create a common date field from multiple date sets (from multiple data sources), and still keep the ability to use each original date field individually.
- ▶ **Financial Reporting:** Bring in an additional data source to help analyze data from transaction systems. For example, Financial GL data will include all transactions but may not have all the income statement or balance sheet reporting definitions.
- ▶ **Looking Up Values:** Look up a value from one table and bring it into another table. For example, knowing how much a marketing campaign costs versus the sales opportunity amount is an important KPI to measure.

Reformatting Data

Reformat field data to free space, and make fields more readable and usable. For example, convert a date field to numeric, or reduce the precision of real numbers. You can reformat fields within the ElastiCube using a custom SQL expression.

- ▶ **Numeric Representation of Date Fields:** Create a date table that is represented by a numeric representation instead of a date field to improve the query performance, as well as provide more flexibility, including the ability to filter a date range.

Enhancing Data

Enhance data by adding attributes/records that did not exist in the original data source.

- ▶ **Calculating derived facts:** Derived Facts are additional facts that we calculate while importing or delivering the data.
- ▶ **Calendar vs. Fiscal Year:** Align a fiscal calendar with a Gregorian calendar.
- ▶ **Time Zone Conversion:** Use a source table to convert dates and times from different time zones into a uniform data set.
- ▶ **Currency Conversion:** Convert one currency into another using custom fields and a currency exchange rate table.
- ▶ **Current vs. Previous Period for Specific Date Range:** Compare data such as sales between a current period and a past period.
- ▶ **Calculating the Number of Open Orders per Day:** Check open sales orders where the order has been placed but has not yet been delivered.
- ▶ **Slowly Changing Dimensions:** Transactional data does not usually change, however the data that describes the associated dimensions may change. See how to manage dimensions that may be updated with new values within the data warehouse at different points in time.

Integrating Data

The following examples explain how to integrate and merge data from different sources into a single ElastiCube structure. This requires properly planning how to merge the data; to avoid creating unnecessary relationships, while avoiding many to many relationships. Examples in this section:

- ▶ Creating a Common Date Selection
- ▶ Financial Reporting
- ▶ Looking Up Values

Creating a Common Date Selection

Business Case

When pulling together data from multiple sources, you will have a number of different dates. Marketing has a Campaign Date, Sales has an Opportunity Date and Finance has a GL Date.

Modeling Challenge

This type of data leaves us with three sets of dates. Modeling it properly will allow you to select from one common date field while still leaving you the option to choose from one of the three date fields individually.

Solution

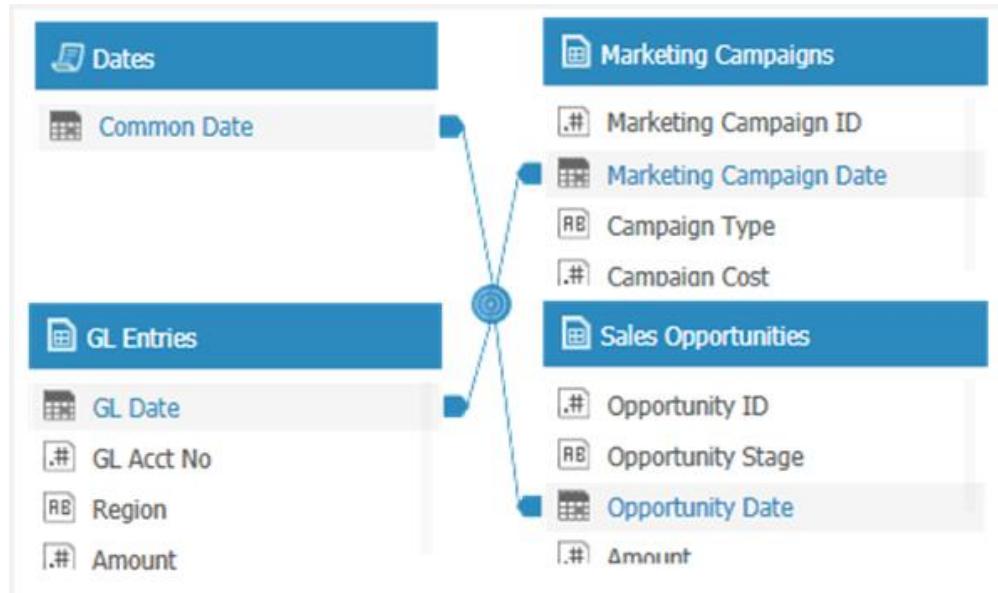
Create a custom table that retrieves a unique list of the dates used between the three tables.

GL Entries	Marketing Campaigns	Sales Opportunities
GL Date	Marketing Campaign ID	Opportunity ID
GL Acct No	Marketing Campaign Date	Opportunity Stage
Region	Campaign Type	Opportunity Date
Amount	Camoaion Cost	Amount

Use the following syntax:

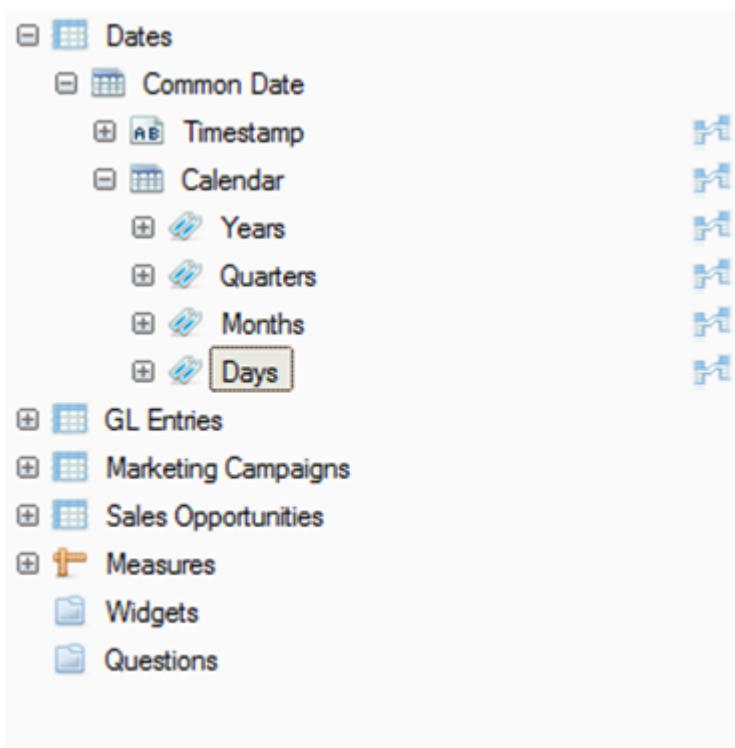
```
Select [GL Date] AS [Common Date]
FROM [GL Entries]
Union
Select [Marketing Campaign Date] AS [Common Date]
FROM [Marketing Campaigns]
Union
Select [Opportunity Date] AS [Common Date]
FROM [Sales Opportunities]
```

Link the four tables together:



Results

This gives us the common date field to use. This allows, for example, to select a month that will narrow down the selections across all three tables.



Financial

Reporting

Business Case

Transactional systems are meant for handling transactions and not for reporting and analysis. For example, Financial GL data will include all the transactions but may not include all income statement or balance sheet reporting definitions.

Modeling Challenge

This data usually resides in other data sources or tables.

Solution

Here is what our sample GL entries look like. It is not very useful to analysis and reporting.

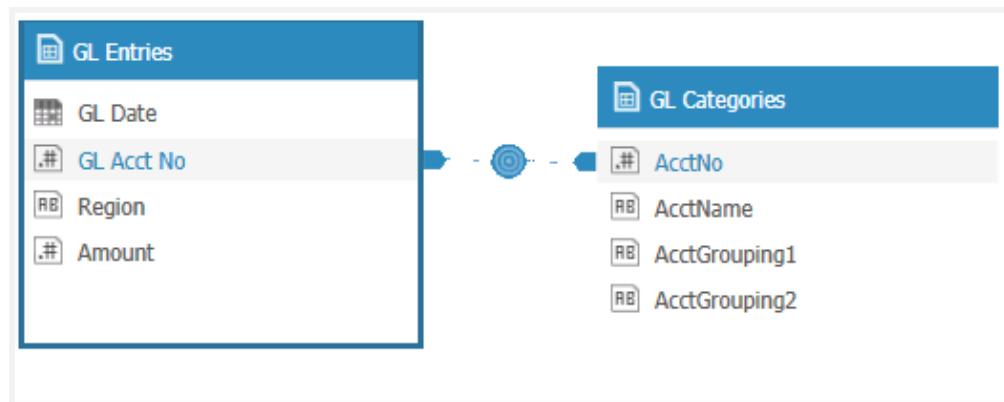
GL Entries - Table Preview

Output	Errors		
GL Date	GL Acct No	Region	Amount
3/10/2013 12:00:00 AM	1004	US	49490
4/14/2013 12:00:00 AM	1002	US	86719
7/14/2013 12:00:00 AM	1004	US	6674
1/13/2013 12:00:00 AM	1003	US	81897
4/14/2013 12:00:00 AM	1004	US	50708
7/18/2013 12:00:00 AM	1001	US	26601
6/21/2013 12:00:00 AM	1005	US	39608
2/12/2013 12:00:00 AM	1003	US	39412
2/4/2013 12:00:00 AM	1001	US	60773
1/5/2013 12:00:00 AM	1004	US	10856
1/2/2013 12:00:00 AM	1002	US	4735
6/20/2013 12:00:00 AM	1002	US	61178
1/31/2013 12:00:00 AM	1003	US	51932
4/27/2013 12:00:00 AM	1004	US	87825

499 Rows

Another data source is needed to help define how the data will be analyzed and reported.

Start by bringing in another Data Source that contains details about the accounts and how they are categorized:



The two tables are linked on the account number. As there is one record in the GL Categories for multiple rows in the GL Entries, this is known as a One-to-Many Relationship.

The end results show that we can easily view the data in an organized fashion for analysis and reporting.

Revenue, COGs and Expenses i			
AcctGrouping2	AcctGrouping1	AcctNo	Sum Amount
COGS	COGS	1009	35,434
		1010	231,231
		2001	44,634
	Salaries	3001	3,543,432
		3002	242,423
		3003	523,543
		3004	91,343
Expenses	Operationg Expenses	3009	8,546,214
		3010	8,214,443
	Travel & Entertainment	3005	123,134
		3006	54,354
		3007	234,234
		3008	4,374,573
Revenue	License	1001	799,231
		1002	2,264,080
	Maintenance	1005	996,276
		1006	1,286,252
		1007	321,347
		1008	6,534,321
	Services	1003	3,319,673
		1004	2,335,265
Sum Amount			44,115,437

Looking Up Values

Business Case

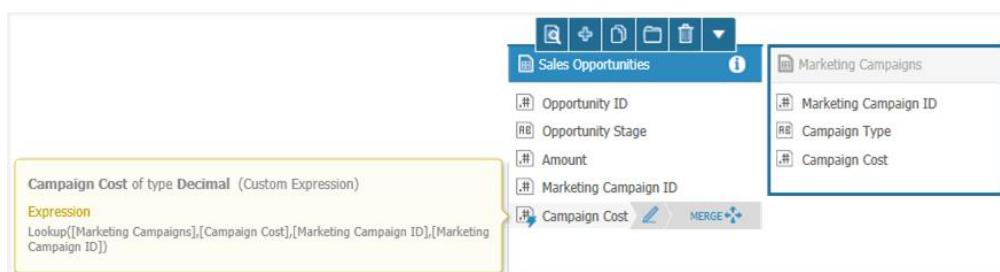
Sometimes it is necessary to look up a value from one table and bring it into another table. For example, knowing how much a Marketing Campaign costs versus the Sales Opportunity amount.

Modeling Challenge

These two amounts typically reside in different systems. We need to look up the value from the Marketing system and bring it into the main Sales table.

Solution

In the Sales Opportunities table, create a custom field that will allow to look up a value from the Marketing Campaigns table based on the Marketing Campaign ID. You can do this using the lookup function.



The screenshot shows the Sisense Data Studio interface. On the left, a yellow callout box highlights a custom expression named "Campaign Cost of type Decimal (Custom Expression)". The expression is defined as "Lookup([Marketing Campaigns],[Campaign Cost],[Marketing Campaign ID],[Marketing Campaign ID])". On the right, the "Sales Opportunities" table is shown with fields: Opportunity ID, Opportunity Stage, Amount, Marketing Campaign ID, and the newly created "Campaign Cost" field. A "Marketing Campaigns" table is also visible, containing fields: Marketing Campaign ID, Campaign Type, and Campaign Cost. A "MERGE" button is present between the two tables.

Make the Marketing Campaigns table Invisible as you only need the one field in the Sales Opportunities table. As there is only One Marketing Campaign ID per Opportunity ID, this is known as a 1-to-1 relationship.

Reformatting Data

By reformatting a field, you can create a more readable, and more usable format for analysis, as well as less space consuming in some cases. For example, convert a date field to a numeric field. You can reformat fields within the ElastiCube using a custom SQL expression.

An example:

Numeric Representation of Date fields

Business Case

Create a "Date" table using a numeric representation of a Date field instead of the Date field itself to improve query performances. In addition, a numeric representation of a Date can provide the ability of filtering the data by time range.

Solution

To convert a Date ("4/21/2012 12:36:56 AM") to an integer (20120421), use the following syntax:

```
10000*getyear(Date)+100*getmonth(Date)+getday(Date)
```

To get the hours and minutes too, use a bigint numeric representation.

So, "4/21/2012 12:36:56 AM" will become: 201204210036, using the following syntax:

```
tobigint(100000000*getyear(DateTime)*+getmonth(DateTime)*1000  
000+getday(DateTime)*1000+100*gethour(DateTime)+getminute(Da  
teTime))
```

Enhancing Data

The following examples explain how to add attributes and/or records that did not exist in the data source. Examples include:

- ▶ Calculating Derived Facts
- ▶ Calendar vs. Fiscal Year
- ▶ Time Zone Conversion
- ▶ Currency Conversion
- ▶ Current vs. Previous Period for Specific Date Range
- ▶ Calculating the Number of Open Orders per Day
- ▶ Slowly Changing Dimensions

Calculating Derived Facts

Business Case

Derived Facts are additional facts that we calculate while importing or delivering the data. For example:

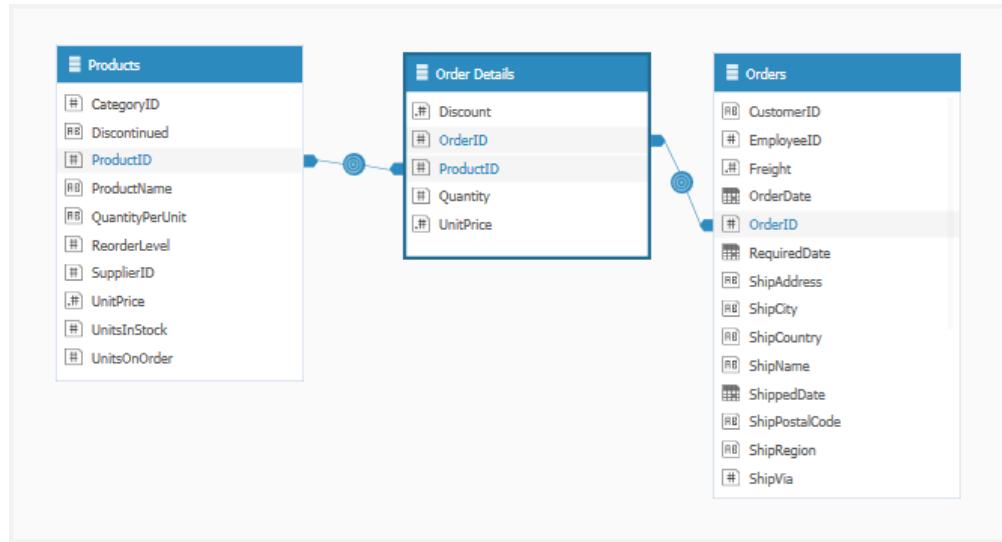
$$\begin{aligned} \text{Amount} &= \text{Qty} * \text{UnitPrice} \\ \text{Profit} &= \text{Amount} - \text{Cost} \\ \text{Inventory Ratio} &= \frac{\text{Sum(Qty Sold)}}{\text{Sum(Qty Ordered)}} \\ \text{AVG Price} &= \frac{\text{Sum(Qty*UnitPrice)}}{\text{Sum(Qty)}} \end{aligned}$$

Modeling Challenge

You must decide whether to calculate the derived facts "on demand", meaning in the web application, or in advance in the ElastiCube. Take into consideration that calculating 'On Demand' Derived Facts in the web application can enable more dynamic filtering, while calculating them in the ElastiCube stage will save query time when retrieving the data, and enforce calculation consistency, especially with non-trivial facts. This is due to the fact that the dashboard designer/end users will receive consistent results for measures, instead of having to create the complex measures individually, by their own understanding.

Solution

In the following schema you can create a derived fact to calculate the *inventory ratio per product*.



Create a custom table using an SQL Expression that joins the “Order Details” table with the “Products” table and returns the division result of “Quantity” and “UnitOnOrder”, with the following Syntax:

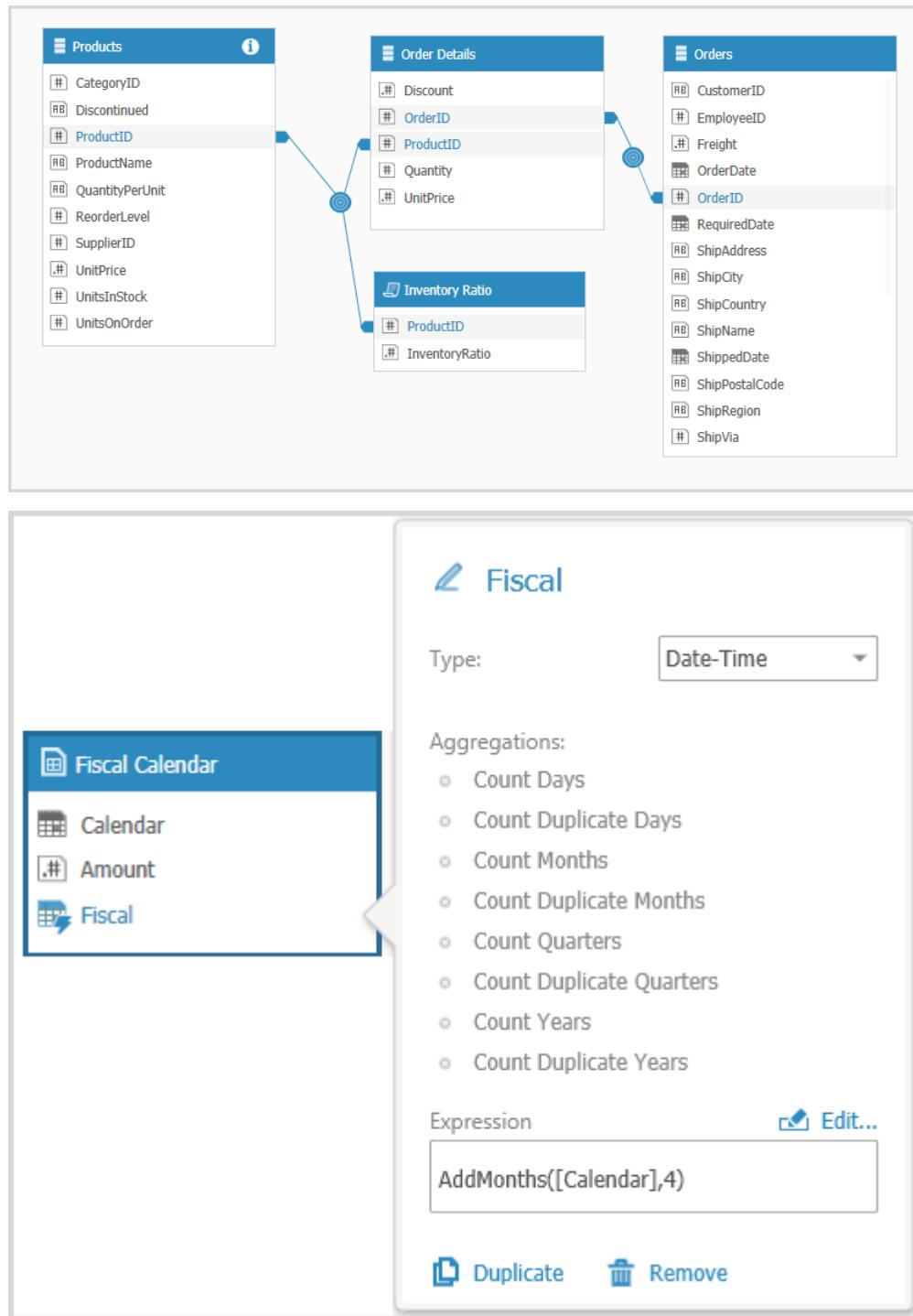
```

SELECT
    [Products].ProductID,
    tofloat(sum(UnitsOnOrder))/tofloat(sum(Quantity)) AS
InventoryRatio
FROM [Products] JOIN [Order Details]
ON [Products].ProductID=[Order Details].ProductID
GROUP BY [Products].ProductID
HAVING tofloat(sum(UnitsOnOrder))/tofloat(sum(Quantity))>0
  
```

The result table will give the desired results:

ProductID	InventoryRatio
11	1.614731
31	2.555476
49	2.423077
74	0.8754209
2	1.66509
32	2.020202
21	1.535433
37	2.4
70	0.4773562
56	0.3958828
43	0.4827586
68	0.4255319
3	2.560976
64	3.243243
66	3.34728
45	1.929134
48	3.043478

Connect the custom table to the rest of the tables:



Note: You can also add the “InventoryRatio” measure to the “Products” table using the `Lookup()` function by “`ProductID`”.

Calendar vs. Fiscal Year

Business Case

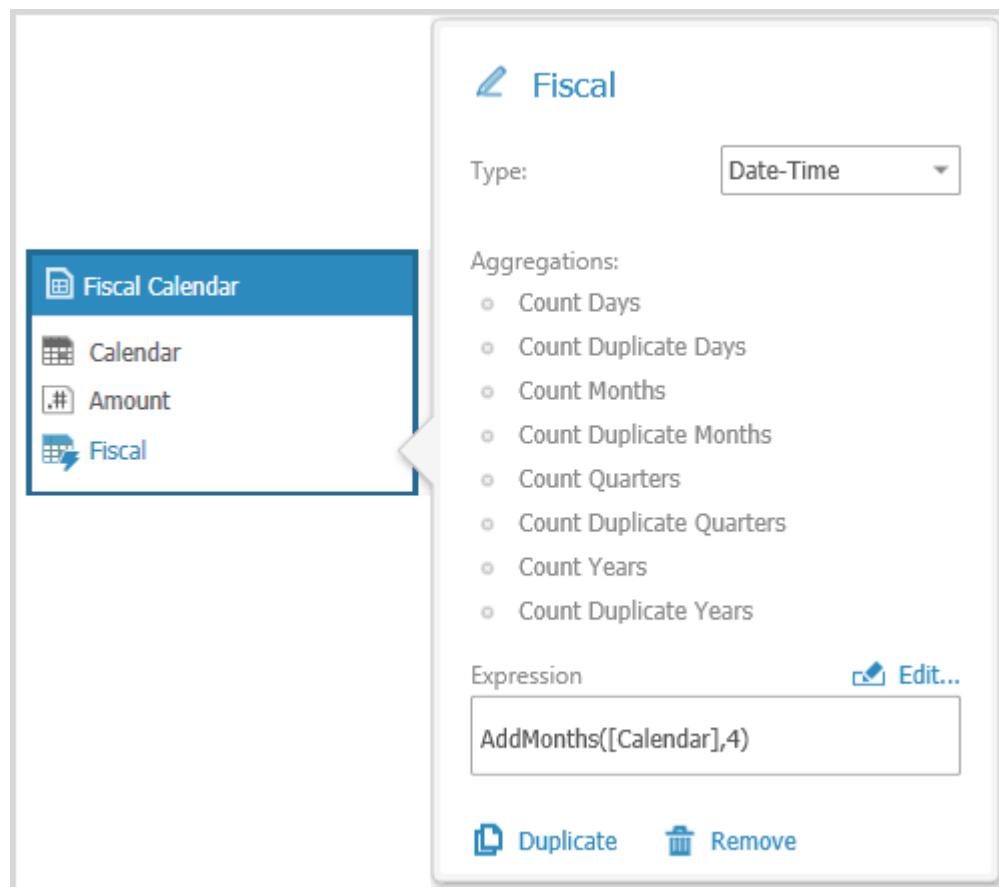
A large number of companies use a fiscal calendar that does not comply with the Gregorian 12-month calendar.

Modeling Challenge

This requires modeling the data properly so that the data can be reported or analyzed via the normal calendar or via the revised fiscal calendar.

Solution

In this example, let's assume that the Fiscal Calendar starts on September 1st. So if we are in the calendar year of 2013, then the fiscal year of 2014 starts September 1st. To accomplish this, we create a custom field that takes the date field and adds four months to it.



The screenshot shows the Sisense Data Studio interface for creating a new field. On the left, there is a sidebar with a tree view containing 'Fiscal Calendar', 'Calendar', 'Amount', and 'Fiscal'. The 'Fiscal' node is selected and highlighted with a blue border. On the right, the main configuration panel has the following details:

- Name:** Fiscal
- Type:** Date-Time
- Aggregations:** A list of aggregation options including Count Days, Count Duplicate Days, Count Months, Count Duplicate Months, Count Quarters, Count Duplicate Quarters, Count Years, and Count Duplicate Years.
- Expression:** AddMonths([Calendar],4)
- Buttons:** Duplicate and Remove

When you create a pivot table in the web application, you will see that the new year (2014) starts in September using the Fiscal field.

Days in Date	Calendar Year	Fiscal Year	Total Amount
08/29/2013	2013	2013	35,756
08/30/2013	2013	2013	3,687
08/31/2013	2013	2013	35,617
09/01/2013	2013	2014	1,108
09/02/2013	2013	2014	91,387
09/03/2013	2013	2014	11,330
09/04/2013	2013	2014	69,273

Time Zone Conversion

Business Case

In many cases, we need to generate reports based on data from different time zones.

Modeling Challenge

When working with different time zones, the challenge is to store all of the business transactions in an absolute time reference that does not change with the seasons, locations (for instance - GMT), or daylight saving. Therefore, the absolute transition time is a combination of *location* and *date*.

Solution

The aim is to add an “absolute time” field to every business transaction, based on its location and time.

Step 1 - Create a reference source table

Create a source table (database table / Excel / CSV) that contains the countries and cities that exist in the database, a numeric representation of timestamp range to determine if the transaction belongs to daylight savings time or not (see the this web site), and the UTC to allow the conversion to GMT.

For example:

Country	City	DST_From	DST_To	UTC
USA	Seattle	20120311.2	20121103.1	-7
USA	Seattle	20121103.1	20130310.2	-8
USA	Seattle	20130310.2	20131027.1	-7
USA	Seattle	20131027.1	20140309.2	-8
UK	London	20120325.1	20121028.2	0
UK	London	20121028.2	20130330.1	1
UK	London	20130330.1	20131027.2	0
UK	London	20131027.2	20140330.1	1

Step 2 - Add a numeric representation of the OrderDate

To associate the Order Date with its UTC, create a custom field of type "Decimal" with a numeric representation of the Date timestamp, using this SQL statement:

```
getyear(OrderDate)*10000+getmonth(OrderDate)*100+getday(Order  
Date)+ToDouble(gethour(OrderDate))/100
```

The result table should look like this:

CustomerID	OrderID	EmployeeID	OrderDate	DateNum	ShipAddress	ShipCity	ShipCountry	Freight
VINET	10248	5	7/4/2012 12:00:00 AM	20120704	59 rue de l'Abbaye	Reims	France	32.38
TOMSP	10249	6	7/5/2012 12:00:00 AM	20120705	Luisenstr. 48	Münster	Germany	11.61
HANAR	10250	4	7/8/2012 12:00:00 AM	20120708	Rua do Paço, 67	Rio de Janeiro	Brazil	65.83
VICTE	10251	3	7/8/2012 12:00:00 AM	20120708	2, rue du Commerce	Lyon	France	41.34
SUPRD	10252	4	7/9/2012 12:00:00 AM	20120709	Boulevard Tirou, 255	Charleroi	Belgium	51.3
HANAR	10253	3	7/10/2012 12:00:00 AM	20120710	Rua do Paço, 67	Rio de Janeiro	Brazil	58.17
CHOPS	10254	5	7/11/2012 12:00:00 AM	20120711	Hauptstr. 31	Bern	Switzerland	22.98
RICSU	10255	9	7/12/2012 12:00:00 AM	20120712	Starenweg 5	Genève	Switzerland	148.33
WELLI	10256	3	7/15/2012 12:00:00 AM	20120715	Rua do Mercado, 12	Resende	Brazil	13.97
HILAA	10257	4	7/16/2012 12:00:00 AM	20120716	Carrera 22 con Ave. Carlos Soublette #8-35	San Cristóbal	Venezuela	81.91
ERNSH	10258	1	7/17/2012 12:00:00 AM	20120717	Kirchgasse 6	Graz	Austria	140.51
CENTC	10259	4	7/18/2012 12:00:00 AM	20120718	Sierras de Granada 9993	México D.F.	Mexico	3.25
OTTIK	10260	4	7/19/2012 12:00:00 AM	20120719	Mehrheimerstr. 369	Köln	Germany	55.09
QUEDE	10261	4	7/19/2012 12:00:00 AM	20120719	Rua da Panificadora, 12	Rio de Janeiro	Brazil	3.05
RATTC	10262	8	7/22/2012 12:00:00 AM	20120722	2817 Milton Dr.	Albuquerque	USA	48.29
ERNSH	10263	9	7/23/2012 12:00:00 AM	20120723	Kirchgasse 6	Graz	Austria	146.06
FOLKO	10264	6	7/24/2012 12:00:00 AM	20120724	Åkergratan 24	Bräcke	Sweden	3.67
BLONP	10265	2	7/25/2012 12:00:00 AM	20120725	24, place Kléber	Strasbourg	France	55.28
WARTH	10266	3	7/26/2012 12:00:00 AM	20120726	Torikatu 38	Oulu	Finland	25.73
FRANK	10267	4	7/29/2012 12:00:00 AM	20120729	Berliner Platz 43	München	Germany	208.58
GROSR	10268	8	7/30/2012 12:00:00 AM	20120730	5º Ave. Los Palos Grandes	Caracas	Venezuela	66.29
WHITC	10269	5	7/31/2012 12:00:00 AM	20120731	1029 - 12th Ave. S.	Seattle	USA	4.56
WARTH	10270	1	8/1/2012 12:00:00 AM	20120801	Torikatu 38	Oulu	Finland	136.54
SPLIR	10271	6	8/1/2012 12:00:00 AM	20120801	P.O. Box 555	Lander	USA	4.54
RATTC	10272	6	8/2/2012 12:00:00 AM	20120802	2817 Milton Dr.	Albuquerque	USA	98.03
QUICK	10273	3	8/5/2012 12:00:00 AM	20120805	Taucherstraße 10	Cunewalde	Germany	76.07
VINET	10274	6	8/6/2012 12:00:00 AM	20120806	59 rue de l'Abbaye	Reims	France	6.01
MAGAA	10275	1	8/7/2012 12:00:00 AM	20120807	Via Ludovico il Moro 22	Bergamo	Italy	26.93
TORTU	10276	8	8/8/2012 12:00:00 AM	20120808	Avda. Azteca 123	México D.F.	Mexico	13.84
MORGK	10277	2	8/9/2012 12:00:00 AM	20120809	Heerstr. 22	Leipzig	Germany	125.77

Step 3 - Join between the two tables

The third step includes creating a custom SQL expression that joins between the two tables and creating the “Absolute Time” custom field within it (“GMTDate”). This is to create a synchronization between all the transactions. The custom field will be created using the “add hours” function with the matching UTC value. See the following script:

```

SELECT
    [Orders].CustomerID,
    [Orders].EmployeeID,
    [Orders].Freight,
    [Orders].OrderDate,
    [Orders].OrderID,
    [Orders].ShipAddress,
    [Orders].ShipCity,
    [Orders].ShipCountry,
    AddHours(([Orders].OrderDate), [GMT
Conversion.csv].UTC) AS GMTDate
FROM [Orders]
JOIN
[GMT Conversion.csv]
ON
[Orders].ShipCity=[GMT Conversion.csv].City AND
[Orders].ShipCountry=[GMT Conversion.csv].Country AND
[Orders].DateNum>=[GMT Conversion.csv].DST_From AND

```

[Orders].DateNum<[GMT Conversion.csv].DST_To

The result table will look like this:

CustomerID	EmployeeID	Freight	OrderDate	OrderID	ShipAddress	ShipCity	ShipCountry	GMTDate
WHITC	5	4.56	7/31/2012 12:00:00 AM	10269	1029 - 12th Ave. S.	Seattle	USA	7/30/2012 4:00:00 PM
WHITC	5	4.56	7/31/2012 12:00:00 AM	10269	1029 - 12th Ave. S.	Seattle	USA	7/30/2012 5:00:00 PM
WHITC	5	4.56	7/31/2012 12:00:00 AM	10269	1029 - 12th Ave. S.	Seattle	USA	7/30/2012 4:00:00 PM
WHITC	5	4.56	7/31/2012 12:00:00 AM	10269	1029 - 12th Ave. S.	Seattle	USA	7/30/2012 5:00:00 PM
BSBEV	7	22.77	8/26/2012 12:00:00 AM	10289	Fauntleroy Circus	London	UK	8/26/2012 1:00:00 AM
BSBEV	7	22.77	8/26/2012 12:00:00 AM	10289	Fauntleroy Circus	London	UK	8/26/2012 12:00:00 AM
BSBEV	7	22.77	8/26/2012 12:00:00 AM	10289	Fauntleroy Circus	London	UK	8/26/2012 1:00:00 AM
BSBEV	7	22.77	8/26/2012 12:00:00 AM	10289	Fauntleroy Circus	London	UK	8/26/2012 12:00:00 AM
WHITC	4	23.29	11/1/2012 12:00:00 AM	10344	1029 - 12th Ave. S.	Seattle	USA	10/31/2012 4:00:00 PM
WHITC	4	23.29	11/1/2012 12:00:00 AM	10344	1029 - 12th Ave. S.	Seattle	USA	10/31/2012 5:00:00 PM
WHITC	4	23.29	11/1/2012 12:00:00 AM	10344	1029 - 12th Ave. S.	Seattle	USA	10/31/2012 4:00:00 PM
WHITC	4	23.29	11/1/2012 12:00:00 AM	10344	1029 - 12th Ave. S.	Seattle	USA	10/31/2012 5:00:00 PM
SEVES	5	288.43	11/21/2012 12:00:00 AM	10359	90 Wadhurst Rd.	London	UK	11/21/2012 1:00:00 AM
SEVES	5	288.43	11/21/2012 12:00:00 AM	10359	90 Wadhurst Rd.	London	UK	11/21/2012 12:00:00 AM
SEVES	5	288.43	11/21/2012 12:00:00 AM	10359	90 Wadhurst Rd.	London	UK	11/21/2012 1:00:00 AM
SEVES	5	288.43	11/21/2012 12:00:00 AM	10359	90 Wadhurst Rd.	London	UK	11/21/2012 12:00:00 AM
EASTC	1	71.97	11/26/2012 12:00:00 AM	10364	35 King George	London	UK	11/26/2012 1:00:00 AM
EASTC	1	71.97	11/26/2012 12:00:00 AM	10364	35 King George	London	UK	11/26/2012 12:00:00 AM
EASTC	1	71.97	11/26/2012 12:00:00 AM	10364	35 King George	London	UK	11/26/2012 1:00:00 AM
SEVES	1	22.21	12/9/2012 12:00:00 AM	10377	90 Wadhurst Rd.	London	UK	12/9/2012 1:00:00 AM
SEVES	1	22.21	12/9/2012 12:00:00 AM	10377	90 Wadhurst Rd.	London	UK	12/9/2012 12:00:00 AM
SEVES	1	22.21	12/9/2012 12:00:00 AM	10377	90 Wadhurst Rd.	London	UK	12/9/2012 1:00:00 AM
SEVES	1	22.21	12/9/2012 12:00:00 AM	10377	90 Wadhurst Rd.	London	UK	12/9/2012 12:00:00 AM
SEVES	2	34.86	12/19/2012 12:00:00 AM	10388	90 Wadhurst Rd.	London	UK	12/19/2012 1:00:00 AM
SEVES	2	34.86	12/19/2012 12:00:00 AM	10388	90 Wadhurst Rd.	London	UK	12/19/2012 12:00:00 AM
SEVES	2	34.86	12/19/2012 12:00:00 AM	10388	90 Wadhurst Rd.	London	UK	12/19/2012 1:00:00 AM

Step 4 - Make schema adjustments

For the next step, do the following:

- ▶ Replace the current Orders table with the new one,
- ▶ Refer to the new “Absolute Time” custom field (“GMTDate”) as the leading date field
- ▶ Make the reference tables (“Orders” and “GMT Conversion.csv”) invisible.

Currency Conversion

Business Case

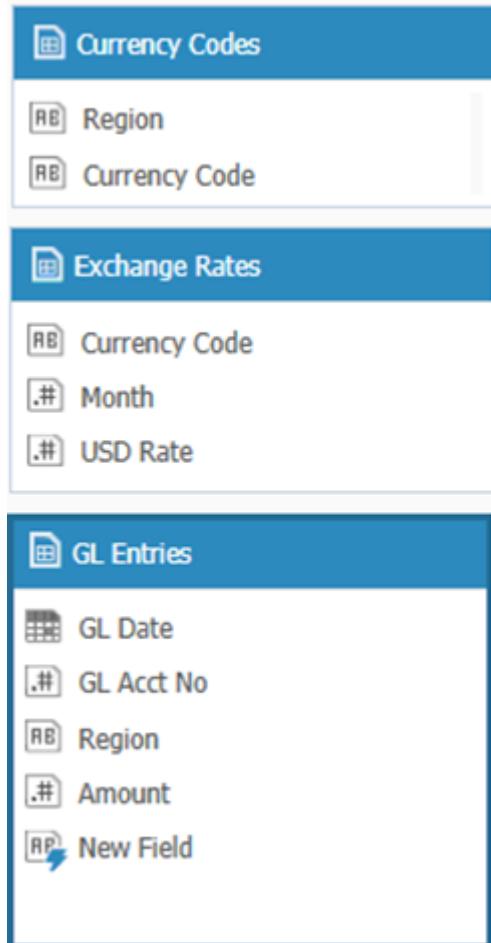
Most data for entities is recorded in their local reporting currency (ie \$ for United States, £ for UK). Here we want to convert all the amounts to USD.

Modeling Challenge

This requires determining the Currency Rate of the region and then multiplying the value in local currency by the associated Exchange Rate by Month.

Solution

Create two custom fields in the GL Entries. The first will look up the Currency code of the region. This field will be used along with a month field to link to the Exchange Rates table.



The screenshot shows three tables in a data modeling interface:

- Currency Codes**: Contains fields "Region" and "Currency Code".
- Exchange Rates**: Contains fields "Currency Code", "Month", and "USD Rate".
- GL Entries**: Contains fields "GL Date", "GL Acct No", "Region", "Amount", and "New Field". A blue arrow points from the "New Field" button in the GL Entries table to the "Currency Code" field in the Exchange Rates table.

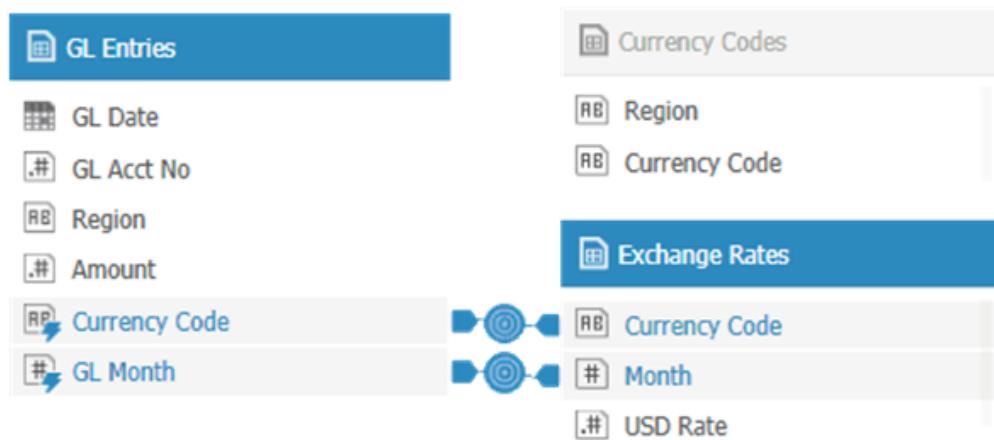
The first field in the GL Entries is created using the lookup function to retrieve values from the Currency Codes table.

```
Lookup([Currency Codes], [Currency Code], Region, Region)
```

Then create a second Custom Field for the Month of the GL Date.

```
GetMonth([GL Date])
```

Next, link the fields together (note that both Month fields were set to Integer and the Currency Codes table to Invisible).



The Local Amount multiplied by the Exchange Rate gives the Converted USD Amount.

LOCAL and USD

Region	GL Month	Local Amount	Exchange Rate	Converted USD Amount
IL	1	1,303,490	.25	325,873
	2	833,853	.27	225,140
	3	1,915,644	.28	536,380
	4	1,303,288	.27	345,371
	5	1,232,185	.29	357,334
	6	1,743,100	.28	479,353
	7	1,387,776	.28	388,577
	8	17,116	.29	4,964
US	1	939,352	1.00	939,352
	2	752,517	1.00	752,517
	3	604,556	1.00	604,556
	4	648,043	1.00	648,043
	5	769,723	1.00	769,723
	6	884,302	1.00	884,302
	7	550,515	1.00	550,515

Current vs. Previous Period for Specific Date Range

Business Case

In many cases we would like to compare our business' performance last week, to the week before, or maybe we would like to see a percentage of sales growth for the current month/quarter compared to the previous month/quarter.

Modeling Challenge

Since we want the compared time range to be as flexible as possible, the solution has to include both layers - ElastiCube and web application.

Solution

1. Create a [custom table](#) in the ElastiCube to summarize the totals/counts per day for the source table:

```
SELECT
    a.Date,
    sum(a.Revenue) AS value
FROM [Accord 2011 Client List] AS a GROUP BY a.Date
```

2. Create a custom table in the ElastiCube with current vs. previous values, by adjusting the script below:

```
SELECT
    curr.Date AS date,
    curr.value AS current,
    prev.value AS prev
FROM [sum] curr
LEFT JOIN [sum] AS prev
ON curr.Date = addyears(prev.Date,1)
UNION
SELECT
    addyears(prev.Date,1) AS date,
    curr.value,
    prev.value
FROM [sum] prev
LEFT JOIN [sum] AS curr
ON prev.Date= addyears(curr.Date,-1)
```

3. In the web application, add a 'date range picker' using the days from the custom table. Then add two new numeric indicators. In the first numeric picker add the 'sum of the current value', in the second numeric picker, add the 'sum of the previous value'.
4. In the date range picker, select the days of interest and you will see the current and previous values.

Calculating the Number of Open Orders per Day

Business Case

An open sales order is where the order has been placed but has not yet been delivered. If for example there is an order for 100 items and

against this order only 50 items have been delivered (it is partially delivered). A high level of open orders per day may indicate that something is wrong with orders handling.

Modeling Challenge

We cannot just count the number of orders per day because it will exclude orders that were open on a certain day and are already closed. Therefore, we will need to create a snapshot of the number of open orders per day.

Solution

1. Import an Excel file with all dates listed in the Orders table into the ElastiCube.
2. To improve query performance, convert all the date fields into numeric representations (for more information, see Numeric Representation of Date fields).
3. Create the following custom table:

```
SELECT
    s.Dates,
    tm.Created_At,
    tm.Closed_At,
    tm.TicketId
FROM [All Dates] s LEFT JOIN [Orders] tm
    ON s.DateInt >= tm.CreatedAtInt
    AND (tm.SolvedAt IS NULL OR s.DateInt <= tm.SolvedAtInt)
```

Slowly Changing Dimensions

Business Case

Transactional data typically does not change, however the data that describes the associated dimensions may change. This example demonstrates how to manage dimensions that may be updated with new values within the data warehouse at different points in time.

For example, a customer that was living in NYC and moved to LA earlier this year.

Date	Customer	City
1/1/2011	John Smith	NYC
1/1/2013	John Smith	LA

Modeling Challenge

Following the example above, when the transactions were made last year, the customer was living in NYC. Later this year, the customer moved to LA.

If you decide to refer only to the last city and summarize revenue by city, the NYC transaction will be credited to LA only because the customer currently lives there.

If you connect the "Customer" field within the above table to the "Customer" field in the fact table, you will create a "Many to many" relationship because "Customer" is not a unique identifier of this table.

Solution

The solution is to change the level of granularity of the "Customer_City" table and add the "Date" field to the key -

1. Concatenation of the Slowly Changing Dimension table's unique identifier (for example - Customer_ID + Date)
2. Creation of the same concatenated field in the transactions table.
3. Merge between the 2 keys.
4. This way, you can associate the [Customer_ID + Date] key of every transaction with the relevant customer city.

Building the ElastiCube

After defining a schema, you will need to 'build' your ElastiCube. This will import the data from your data sources that you added to your schema. You must complete at least once before the ElastiCube data can be used in a dashboard.

To build an ElastiCube:

1. Click ElastiCube in the top menu, and then **Build the ElastiCube**.

or

Within your schema, click Build in the top menu. Select **Build the ElastiCube**.

2. Define the following options:

- ▶ **Target Server:** Select the Sisense server location that hosts the ElastiCube. Click Add Server to add a new location.
- ▶ **Build Options:** Select which data to import. For more information, see Build Settings.
 - **Build Schema Changes:** Imports data into new tables and fields where a change has been made to the schema since the previous build.
 - **Build Entire ElastiCube:** Imports data for all tables with a default of overriding existing data in the ElastiCube. If tables and fields in your schema are set to import accumulative data, then an additional option **Delete existing Data** exists. If selected, all data will be replaced instead of accumulative data where relevant.

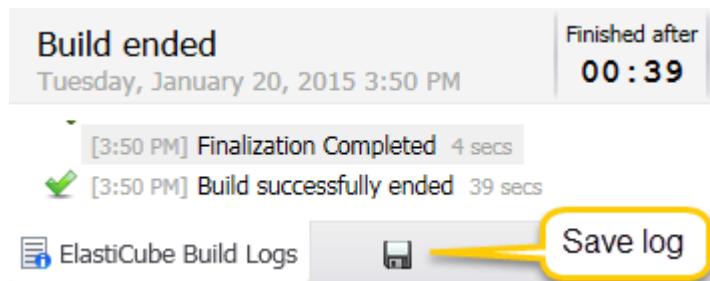
- ▶ Click **More Settings** to reveal the following settings:
 - **Build Logging:** Specifies the level of detail to store in the build log.
 - **Sample data:** Specifies the limitation on the number of table rows to import.
3. Click **Build** to begin the import process.
 4. A log panel at the bottom of the screen will detail each stage of the build. When the build is complete, the following message appears in the logs: *Build successfully ended.*

Build Logs

The build logs record all steps, activities and alerts occurring during a particular build.

These logs can be helpful in determining when the last build occurred, what changes were made, and the status of the ElastiCube.

1. Click **ElastiCube** on the top menu.
2. Click **Recent Logs**.
3. The logs will load and appear at the bottom of the screen.
4. To save the log, click the save icon below the logs panel.



Making Changes to an Existing ElastiCube

This section describes how to make changes in your ElastiCube.

To update an ElastiCube:

1. Open the ElastiCube Schema file using one of the following methods:
 - In the ElastiCube Manager, click **File**, and locate the ElastiCube Schema file (*.ecube), and double click to open.
 - In the **Sisense Server Console**, select the ElastiCube you want to open, and click **Edit**.

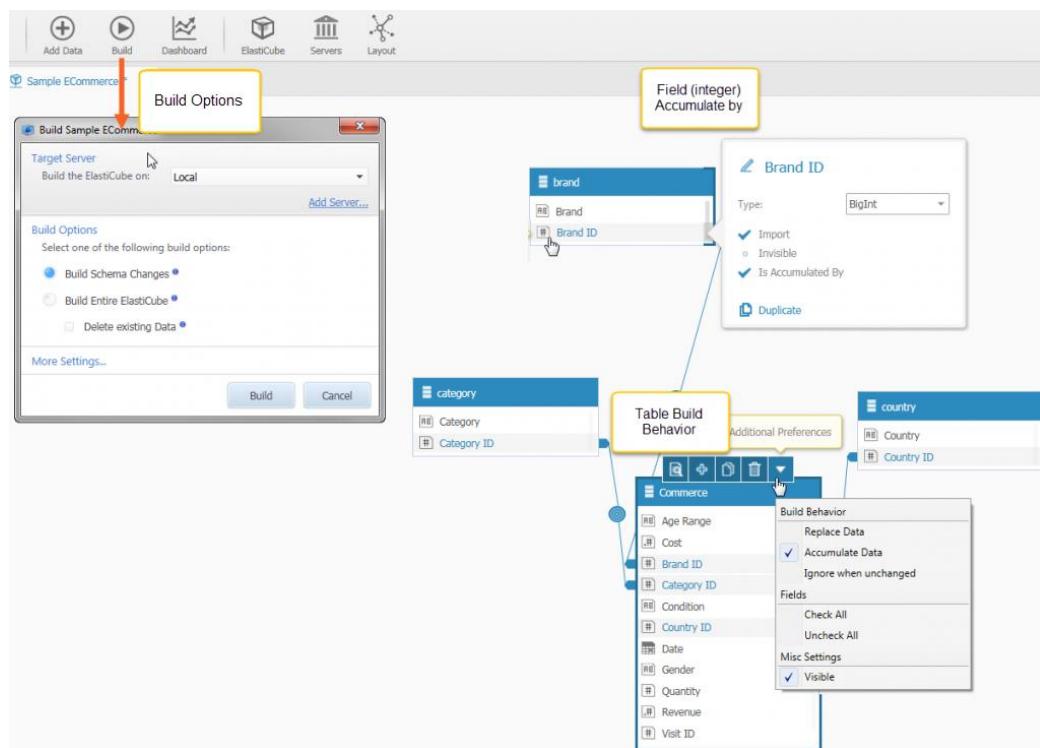
The file will open displaying all associated tables, fields and relationships in the ElastiCube Manager.

2. You can now edit the ElastiCube. Hover over the table name to see editing options. Common actions include:
 - **Add Data:** See Connecting to Source Data.
 - Rename, edit, delete or copy tables. See Managing Tables.
 - **Add Custom Fields or Tables:** See Adding a New Custom Table and Adding a New Field.
3. To apply any changes, click **ElastiCube > Build the ElastiCube...** in the top panel of the ElastiCube Manager.

Build Settings and Data Accumulation Behavior

When you build your Elasticube, data is imported and accumulated based on the following settings:

- ▶ **Build Options:** Defines whether to import all data overriding any previous data, or import only data where a change has been made to the schema since the previous build.
- ▶ **Field Build Behavior:** Defines a specific integer or date field as an index in the table. The index column will be used to determine whether source records are new, and whether to accumulate data in the table based on a higher index value at the time of the build. New builds will only include values greater than the maximum existing value.
- ▶ **Table Build Behavior:** Defines the behavior on the table level. You can replace all data at the time of the build, or add only additional/accumulated data.



Build Options

There are two primary Build Options, **Build Schema Changes** and **Build Entire ElastiCube**.

To access Build Options, click **Build** in the main menu of the ElastiCube Manager. See also Building the ElastiCube.

Build Schema Changes: Updates the build only if changes were made to the schema since the previous build, and imports data only if new tables were added. Conditions for updating the build include:

- New, removed or changes to custom tables

- New, removed or changes to custom fields

- New or removed tables

- New or removed table relationships.

Build Entire ElastiCube: Imports data according to the build behavior defined. By default, Sisense replaces and overrides all existing data in the ElastiCube. While building entire ElastiCubes ensures you have an exact copy of the underlying data source, it can sometimes take a long time and be expensive on the database depending on the size of your data source. You can customize your builds to update only the tables and fields where data was accumulated since the last build. This is useful when you frequently need to refresh a large data source. For example, if you have a data source that is updated daily, rather than rebuild the entire ElastiCube daily, you can just import the new data added each day. This option can significantly reduce the amount of time it takes to complete build depending on the size of your ElastiCube.

While building the ElastiCube with schema changes or entire ElastiCube builds, you can continue to run queries. In the event that the build fails, Sisense restores the original version of the ElastiCube and attempts the build again. While Sisense restores the original ElastiCube, some downtime may occur while Sisense copies and pastes the ElastiCube on your drive. The downtime is the amount of time it takes to copy and paste the ElastiCube locally. If you require high availability for your data, Sisense provides this through ElastiCube Sets.

Note: When Sisense restores an ElastiCube a copy of the original ElastiCube is created. You should verify that your server can store multiple copies ElastiCubes until the restoration process is complete.

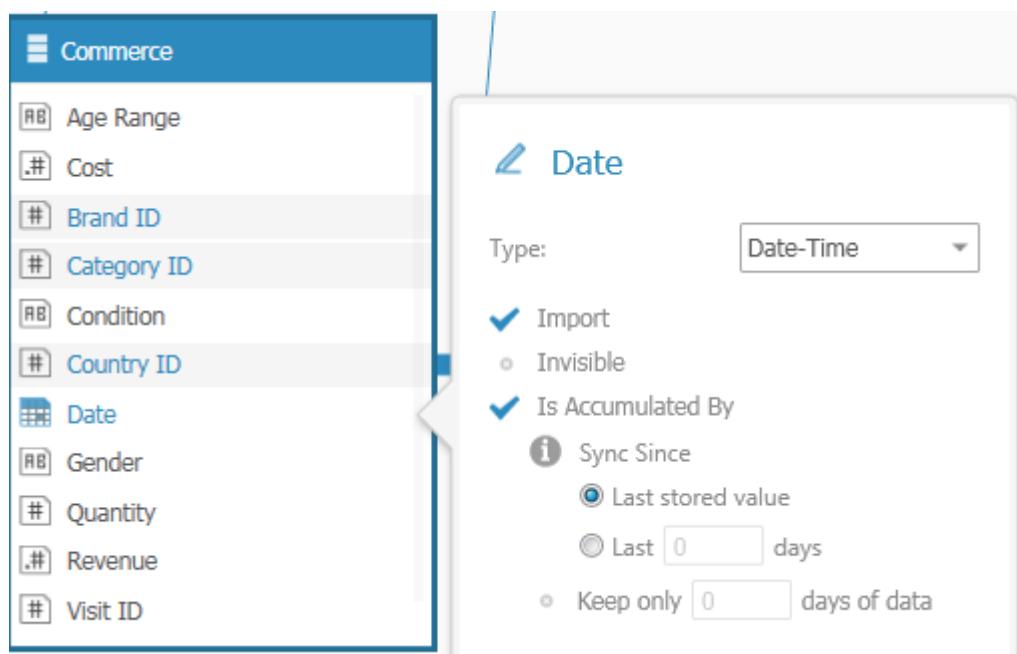
For more information on how to customize your builds, see Field Build Behavior and Table Build Behavior below.

Field Build Behavior

You can select a specific integer or date field that will be used to determine whether to accumulate data at the time of the build. The field will act as an index, and if the index value is greater in a subsequent build then data will be accumulated for the table.

To select the integer or date field within a table as the parameter to accumulate by, click on the field name, and then select **Is Accumulated by**.

Note: The source data must be ordered by the index.



Accumulation behavior for integers and for dates are as follows:

Integer

When you select an integer, only source rows with a value greater than the maximum index value in the ElastiCube table will be inserted. Existing data in the ElastiCube table will not be modified or deleted.

Note: Some data sources (for example, csv files) do not support accumulating data on the field level, and in such cases data will be duplicated. See **Accumulative Build Support** below.

The following image demonstrates this logic (when supported):

Accumulate By Index - Integer

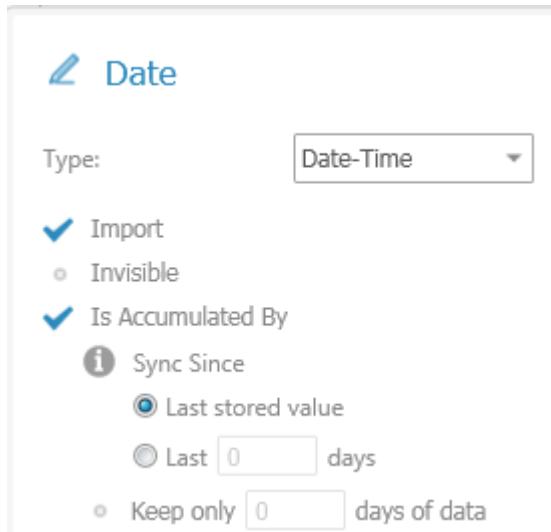
Load 1		Load 2	
Source	Ecube Table	Source	Ecube Table
1	1	1	1
3	3	3	3

Load 1		Load 2	
Source	Ecube Table	Source	Ecube Table
1	1	1	1
3	3	3	3
2		2	4
4		4	

After Load 1, the maximum index value is 3. In Load 2, the source index value of 2 is not inserted (since it is less than 3), but the source index value of 4 is inserted (since it is greater than 3).

Date

When accumulating by date, you can select one of the following options:



The screenshot shows the 'Date' accumulation settings. The 'Type' is set to 'Date-Time'. Under 'Is Accumulated By', 'Import' and 'Sync Since' are checked. 'Sync Since' has 'Last stored value' selected. There is a text input field containing '0' days. An unselected option 'Keep only 0 days of data' is also present.

Last stored value: This is identical to the integer index functionality.

Only source rows where the index is greater than the ElastiCube table's maximum date index value will be inserted. ElastiCube data is never removed or modified after it has been built.

Note: Some data sources (for example, CSV files) do not support accumulating data on the field level, and in such cases data will be duplicated. See **Accumulative Build Support** below.

Last Days: This option specifies the number of days preceding the build date to synchronize.

Note: Syncing “Since Last X days” requires the source data to be ordered by the date.

Synchronization consists of deleting all of the rows in the ElastiCube table where the index is in the specified range, and inserting all source rows where the index value is in the specified range. This option does allow for ElastiCube table data to be deleted or modified. The following example will demonstrate this logic using a three-day range.

Accumulate By Index – Last 3 Days

Load 1 (12/5/2014)		Load 2 (12/6/2014)	
Source	Ecube Table	Source	Ecube Table
12/3/2014	\$5	12/3/2014	\$5
12/4/2014	\$10	12/4/2014	\$10

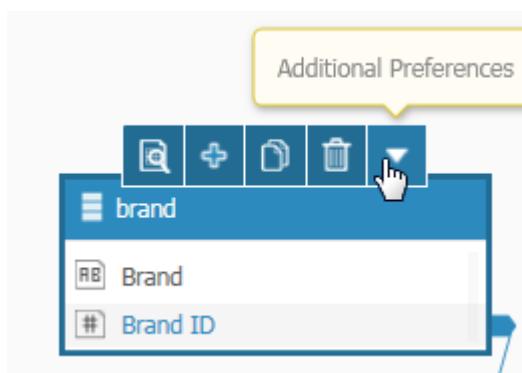
In Load 1, the Ecube table started empty, so nothing was deleted. Both of the source rows are between 12/3/2014 and 12/5/2014 (the 3 day range), so they are inserted. In Load 2, all Elasticcube table records

with the date between 12/4/2014 and 12/6/2014 are removed. In the source table, the data for 12/3/2014 and 12/4/2014 have been updated. However, since 12/3/2014 is outside of our 3 day range, the Elasticube data for that date has not changed. Since 12/4/2014 is in the range, the original Elasticube table record for that day was removed and was replaced by the source table record for that day. 12/5/2014 data is new, so it was also inserted into the Elasticube table.

Table Build Behavior

A table's Build Behavior lets you define whether to import and replace data for the entire table or bring in only accumulated data at the time of the ElastiCube build.

To select a table's Build behavior, click the table's header and then **Additional Preferences**.



The following options exist:

Replace Data (default): Replaces all data at the time of the build. This is recommended for dimension tables, for example: store attributes, or dimensions, which describe the objects in a fact table.

Accumulate Data: Adds additional data to existing table data, without making comparisons and without omitting data. This is recommended for detailed fact tables, for example: store quantitative information for analysis.

Note: If one of the fields in the table is set to accumulate data (see below), then data will be accumulated from the selected integer or date field, and not from the table.

Ignore When Unchanged: Does not import any data unless changes have occurred in the table schema. This is recommended with summary/snapshot fact tables and with data marts (smaller subsets of data, tailored for specific needs).

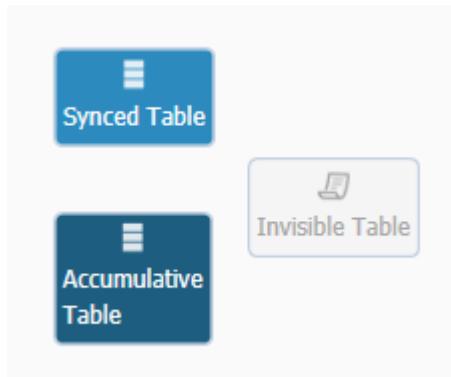
Build Options Working Together with Table and Field Behavior

The following table describes how data will be imported using different build options and table behavior combinations.

	Replace Data	Accumulate Data	Ignore when unchanged
Build Entire Elasticcube	Replaces all data at the time of build.	Only adds additional data to existing table data. If a field in the table is marked as Accumulate by, then data is accumulated only for the selected field. Note: If enabled, an additional checkbox exists to delete all existing data during the build.	Ignores tables that have remained unchanged, unless changes have occurred in the table schema. Note: If enabled, an additional checkbox exists to delete all existing data during the build.

	Replace Data	Accumulate Data	Ignore when unchanged
Build	All data remains unchanged	If a field in the table is marked as accumulate by, then data is accumulated only for the selected field.	Ignores tables that have remained unchanged, unless changes have occurred in the table schema.
Schema Changes	during the build.		

Note: Tables with accumulative build settings are highlighted with a dark blue color.



Accumulative Build Support

The following list displays different data sources and their support for accumulating data on the table and field levels.

Data Source Provider	Accumulate by Table	Accumulate by Field
Oracle	Yes	Yes
MySQL	Yes	Yes
MS SQL	Yes	Yes
PostgreSQL	Yes	Yes

Data Source Provider	Accumulate by Table	Accumulate by Field
ODBC	Yes	No
OLEDB	Yes	No
Hive	Yes	Yes
MS Excel	Yes	No
CSV file	Yes	No
MS Access	Yes	No
Salesforce	Yes	No
Google AdWords	Yes	No
Google Analytics	No	No
Google Spreadsheets	Yes	No
Facebook	Yes	Yes
HubSpot	Yes	Yes
DynamoDB	Yes	Yes
ZenDesk	Yes	No
Amazon Redshift	Yes	Yes
Heroku Postgres	Yes	Yes
DB2	Yes	Yes
Teradata	Yes	Yes

Data Source Provider	Accumulate by Table	Accumulate by Field
MongoDB (ODBC)	Yes	Yes

Managing ElastiCubes

Sisense Server Console

You can use the Sisense Server Console to perform various management tasks on your ElastiCubes, including:

- ▶ Scheduling data loads
- ▶ Importing and exporting data
- ▶ Attaching and detaching directories
- ▶ Starting and stopping an ElastiCube
- ▶ Changing the location of your ElastiCube data storage
- ▶ Cancelling running queries

To open the Sisense Server Console:

- ▶ Right-click the systems tray and double-click the console icon. 
- or
- Click on the Windows start menu, and select **All Programs > Sisense > Sisense Server Console.**

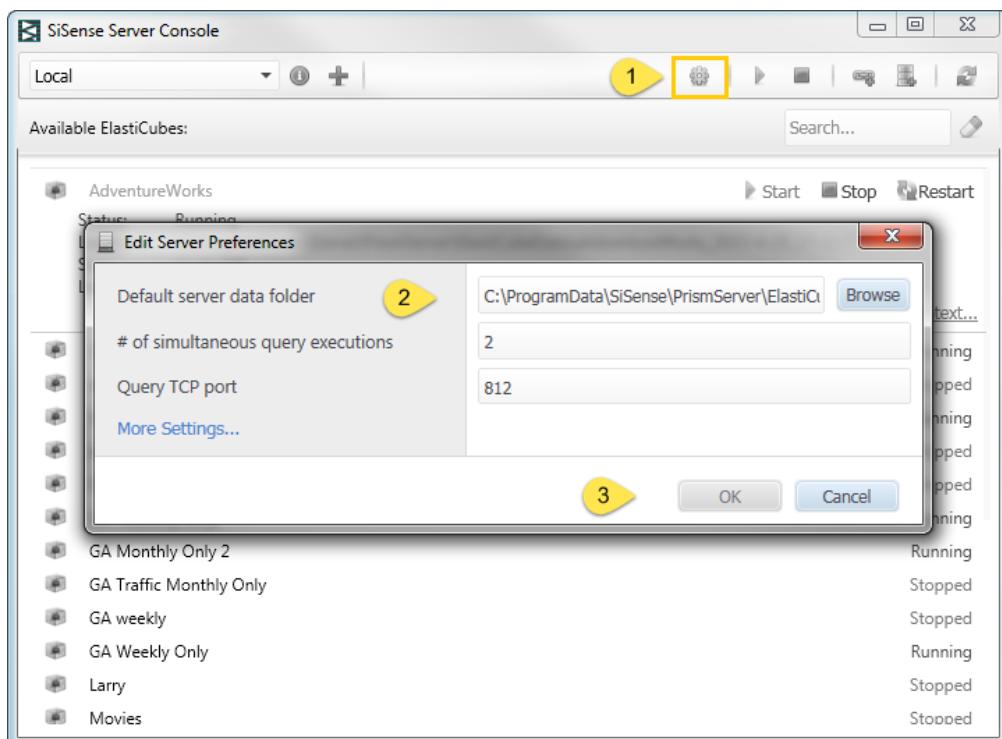
ElastiCube data is stored in a default folder, which you can change in the Sisense Server Console.

Changing the Location of the ElastiCube Data Storage

To change the folder:

1. From the Windows start menu, open the Sisense Server Console.
2. In the top panel, click on the cog icon to view/change server preferences.
3. Click **Browse** next to **Default server data folder**.

4. Select the location of the folder where you want to store ElastiCube data, and click **OK**.



Cancelling Running Queries

You can cancel all running queries in your ElastiCube to free up resources. This is useful in cases where your ElastiCube is consuming too many resources, and as a result, causing the Sisense web app to be unresponsive.

To cancel all queries:

1. Click on the Windows start menu, and select **All Programs > SiSense > SiSense Server Console**.
2. Click the relevant ElastiCube to view the menu panel.
3. Click **Cancel All Queries**.

Scheduling Data Loads

After you build an ElastiCube, you can schedule future builds to automatically synchronize with the underlying data source to ensure all data is up to date.

By default, scheduled builds are Entire builds unless any table within the ElastiCube is set to Accumulative or a table is set to be ignored.

Your ElastiCube's build configuration determines how the scheduled build is to be performed.

Note: By default, scheduling is disabled.

Synchronization using the ElastiCube Manager

1. Open the ElastiCube Manager.
2. Open the relevant ElastiCube.
3. On the top menu select **ElastiCube > Schedule Build Settings**.
4. Select the relevant scheduling option:
 - ▶ **Disable**: Do not automatically build the ElastiCube (default)
 - ▶ **Time of Day**: Build the ElastiCube daily, at a specified time
 - ▶ **Days of Week**: Build the ElastiCube on specific weekdays at a specified time
 - ▶ **Day of Month**: Build the ElastiCube monthly, at a specified date
 - ▶ **Intervals**: Build the ElastiCube at specified intervals
6. Click **Update** to save the scheduling settings.

Synchronization using the Sisense Server Console

1. Open the Sisense Server Console.
2. Select the relevant ElastiCube.
3. Click the **Schedule Synchronization** menu item.

4. Select the relevant scheduling option:
 - ▶ **Disable:** Do not automatically build the ElastiCube (default)
 - ▶ **Time of Day:** Build the ElastiCube daily, at a specified time
 - ▶ **Days of Week:** Build the ElastiCube on specific weekdays at a specified time
 - ▶ **Day of Month:** Build the ElastiCube monthly, at a specified date
 - ▶ **Intervals:** Build the ElastiCube every specified time interval
5. Click **Update** to save scheduling settings.

Importing and Exporting ElastiCube Data

ElastiCube data can be transferred to different machines. This is useful when you are deploying on a new machine or need to transfer the data to a different server.

The transferred data is stored in a compressed ecdat file. The file contains all of the information contained in an ElastiCubeData folder (including the source data), and is used by the Sisense Server Console to build a new ElastiCube.

To export data:

1. Open the Sisense Server Console.
2. Click on the relevant ElastiCube.
3. Click the 'Stop' button to stop the specific ElastiCube (not the entire server).
4. Click **Export**. Enter a name for the file you are saving, and click **Save**.

To import data:

1. Open the Sisense Server Console.
2. Click on the import ecdat file icon. 

3. Select the location of the eedata file and click **Open**.
4. A window will appear showing the import progress. Once complete, the new ElastiCubes will be added to the list of existing ElastiCubes. Make sure the ElastiCube is running before accessing it.

Note: You can also transfer data by copying the folder where Sisense stores the data locally, instead of exporting and importing the compressed file..

Creating Dashboards

This section describes how to create a new dashboard.

There are two ways you can create dashboards, first, through the Sisense Web Application, and second, through the Sisense REST API.

The difference between these two methods is the Sisense Web Application provides a UI and how you populate the dashboard with widgets.

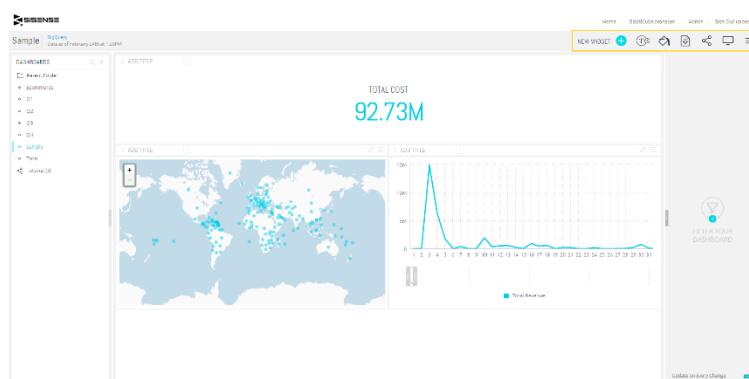
When creating a dashboard through the Sisense Web Application, you build an ElastiCube or select a data set, create a dashboard, and then populate it with widgets.

To create a dashboard through the REST API, you define the dashboard and then populate it with widgets that already exist through their widget IDs.

Creating a New Dashboard

Note: This section is for users with dashboard editing rights (usually with a user role of Designer, and as determined by the user license).

Tip! If you have dashboard editing rights, you should see the editing buttons in the top-right corner of the dashboard.

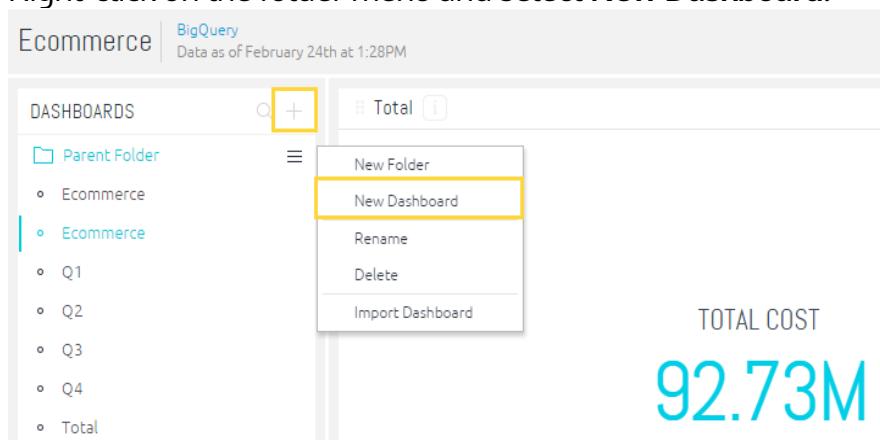


Note: Each Sisense Web Dashboard is based on the data selected from one or more ElastiCubes or data sets. The instructions below assume that the required ElastiCube(s) and data set(s) have already been created. You can also use the sample ElastiCubes that are installed with Sisense if you have not yet created your own.

To create a dashboard:

1. In the HOME page, click the '+' symbol above the Dashboards list.
OR

Right-click on the folder menu and select **New Dashboard**.



The following window appears:

New Dashboard / Please select an Data Set to connect to

Data Set: Commerce

Title: Commerce (1)

Create

Cancel

2. Click the name of the **Data Set** displayed to select the ElastiCube or data set you want to work with. ElastiCubes have the  next to their name while data sets have the  icon next to their name.
3. Click the name of the **TITLE**, and enter a name for the dashboard. This name will appear at the top of the dashboard, and in the **My DASHBOARDS** list.
4. Click **CREATE**.

You are then automatically guided through the process of creating your first Widget in the Widget Wizard, as described in Adding Widgets to a Dashboard.

Adding Widgets to a Dashboard

To create widgets quickly and intuitively, you can use the Widget Wizard. The Widget Wizard automatically guides you through a very simple process of creating a widget while offering the best recommendations for displaying the data that you select.

Almost always, simply accepting the wizard's recommendations will help you create a great Widget. However, if you want more control over your Widget's appearance and behavior, click the **Advanced Configuration** option in the bottom left of the wizard to access a rich variety of additional design options, as described in [Using the Widget Designer](#).

Using the Widget Wizard

To add a widget to the dashboard:

If this is your first widget, click **Select Data**. The Sisense Data Browser opens, and from there you can select one of the fields (columns) from this dashboard's ElastiCube. If you already have widgets in your

dashboard, then click the  button.

Note: To select data from a different ElastiCube, click **from** on the right side of the **NEW WIDGET** window, and select an ElastiCube.

You can add a title to your widget now or after creating your widget.

NEW WIDGET

+ Select Data

ADD TITLE

In the **Data Browser**, select a field to add to your widget. The field that you select is automatically displayed in a suggested widget. When only a single numeric field is selected, it is displayed as an INDICATOR Widget. A single descriptive field is displayed as a PIVOT Widget.

NEW WIDGET

from ECommerce

Total Revenue

+ Add Data



ADD TITLE

TOTAL REVENUE

14M

Advanced configuration

CREATE

CANCEL

Single Numeric Field

NEW WIDGET

from [ECommerce](#)

Brand  Add Data



ADD TITLE

Brand
ABC
APPLES
LONY
MINITENDO
NG
POKIA
Reseller
SHANON
SUNSUNG
WHITEBERRY

[Advanced configuration](#)

CREATE

CANCEL

Single Descriptive Field

But your widget is not really interesting yet, because it has only one field!

Repeat the step above to add more fields to the widget.

As you add fields, the relevant options are displayed as buttons to the left of your selection.



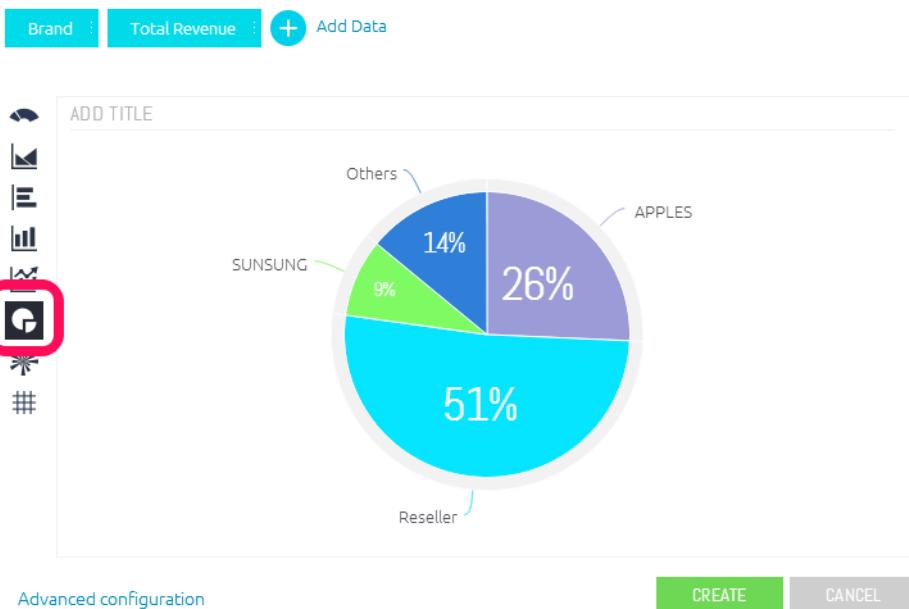
The fields that you select are listed across the top left of the wizard.

You can click on each visualization button to display the selected fields in that visualization/chart.

For example, the same Widget as above could be displayed as a PIE CHART:

NEW WIDGET

from ECommerce



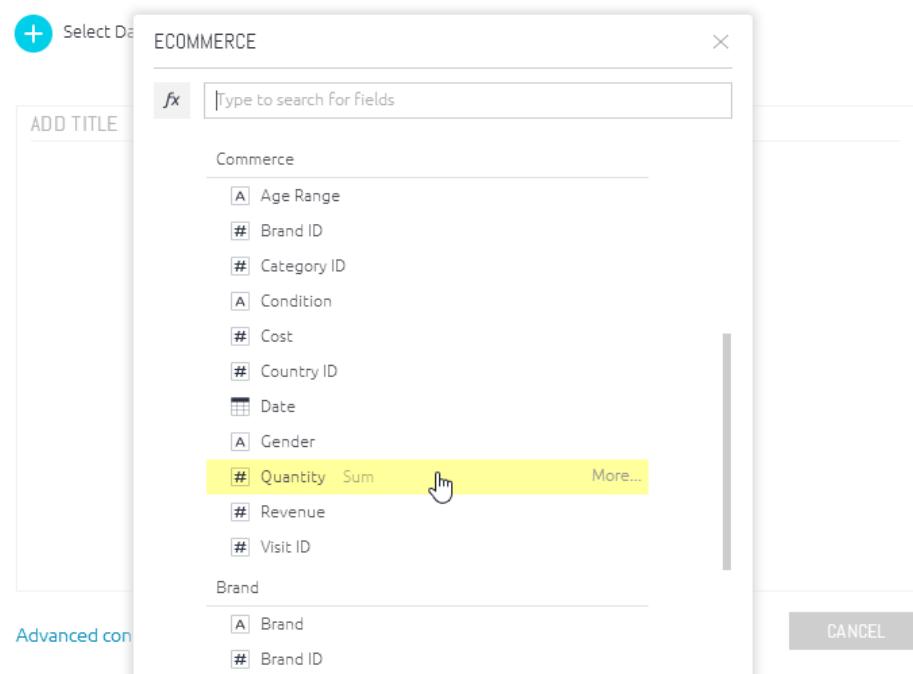
Tip: Sisense provides its best recommendation for your widget design. However, if you want more control, you can click the **Advanced Configuration** option on the bottom left of the window to provide a rich variety of additional design options, as described in Using the Widget Designer.

Data Browser

In the Data Browser, you can select and add fields (columns) from an ElastiCube to your widget.

NEW WIDGET

from ECommerce



The screenshot shows the Sisense Data Browser interface. On the left, there's a sidebar with a '+ Select Data' button and an 'ADD TITLE' input field. The main area is titled 'ECOMMERCE' and contains a search bar with placeholder text 'Type to search for fields'. Below the search bar, there are two sections: 'Commerce' and 'Brand'. The 'Commerce' section lists various fields with their data types indicated by icons: Age Range (A), Brand ID (#), Category ID (#), Condition (A), Cost (#), Country ID (#), Date (#), Gender (A), Quantity (#), Revenue (#), and Visit ID (#). The 'Quantity' field is highlighted with a yellow background and has a hand cursor icon over it. The 'Brand' section lists 'Brand' (A) and 'Brand ID' (#). A 'CANCEL' button is located on the right side of the interface.

An icon appears to the left of each field to indicate its data type:

-  Date
-  Alphanumeric string
-  Numeric

As you type into the Data Browser, the list is dynamically filtered to only show the fields that contain the text you typed.

HOSPITAL X

Admissions

Room_ID

Rooms

Bed_count

Division_ID

ID

Room_number

Simply clicking a field to add it to the widget generally provides great results, however there are more options in the Data Browser.

- ▶ Hover over a field in the Data Browser and click **More ...** to display additional aggregation (quick functions) and filtering options.
- ▶ Click on the  button to define formulas (free-form expressions) that define the field values and filters of a widget. A rich variety of functions are provided for you to use in the formula that you define.

ECOMMERCE X

Commerce

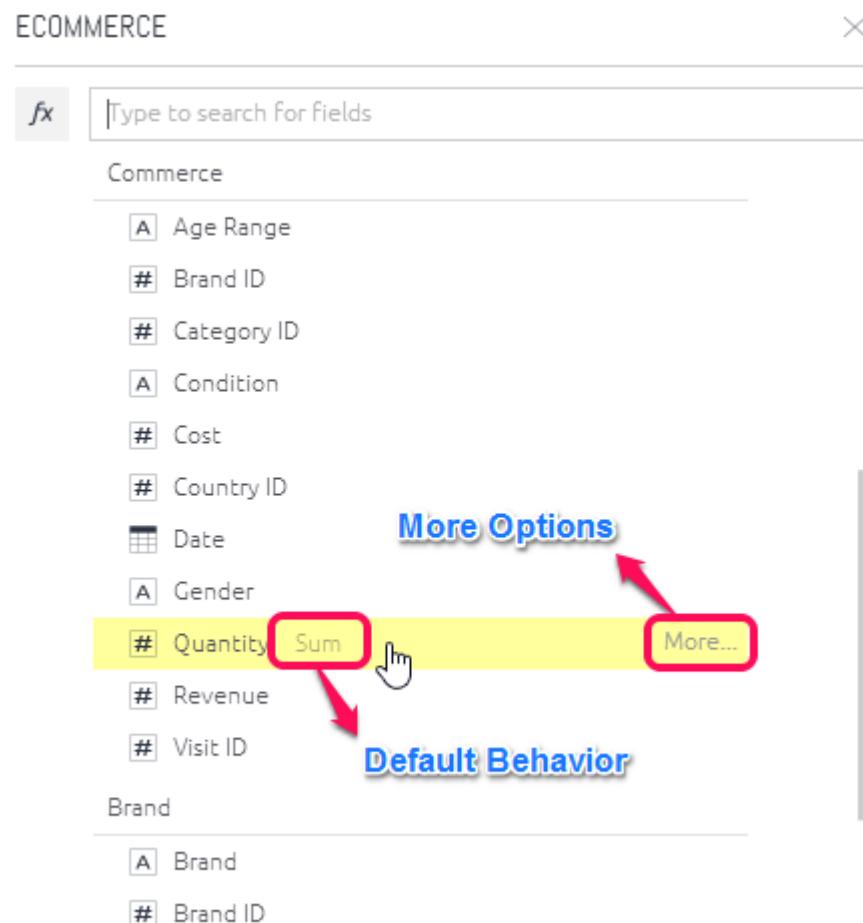
- Age Range
- Brand ID
- Category ID
- Condition
- Cost
- Country ID
- Date
- Gender
- Quantity Sum More...
- Revenue
- Visit ID

More Options More...

Default Behavior

Brand

- Brand
- Brand ID



To learn more about these options, see [Using Formulas](#).

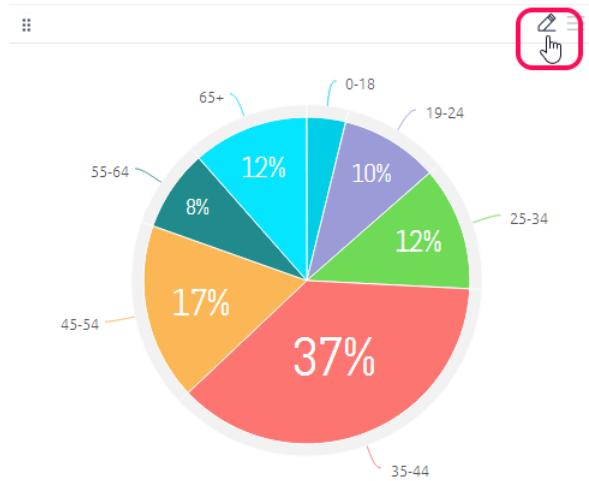
Using the Widget Designer

The Widget Designer lets you fine-tune a widget's appearance and behavior.

Opening the Widget Designer

To display the Widget Designer:

- On the Dashboard, click the **Pencil (Edit)** button  that appears in the top-right corner of a Widget.

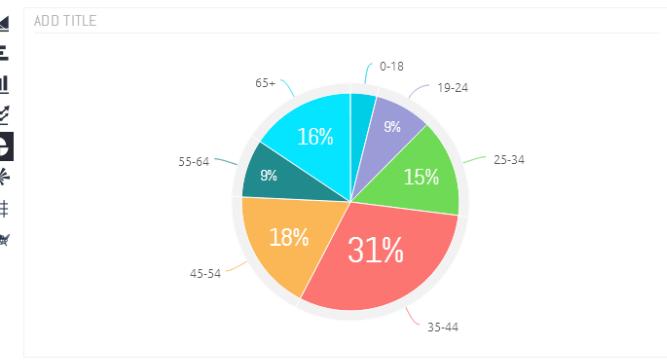


Alternatively, while creating a new widget, in the Widget Wizard, click the **Advanced Configuration** option that appears at the bottom-left of each window. The Widget Designer is then displayed showing the widget in the same state as in the Widget Wizard preview.

NEW WIDGET from Commerce

Total Revenue Age Range + Add Data

ADD TITLE



Age Range	Percentage
35-44	31%
45-54	18%
25-34	15%
19-24	9%
65+	16%
55-64	9%
0-18	1%

Advanced configuration CREATE CANCEL

The Widget Designer – A Quick Tour

The Widget Designer enables you to select the data to be included in a widget in addition to providing a variety of options for customizing the visualization used to show the data.



1. **Adding Data to the Widget (Data Panel):** Used for selecting the values that you want to appear in the widget and those needed for grouping the data. The options differ depending on the visualization (chart type).
2. **Adding a title to the dashboard:** Click *Set a Title* and type in a new title for the widget.
You can also add a widget title directly from the dashboard without entering the widget designer.
3. **Previewing the Widget:** The center of the window displays the current design and content of the widget, which automatically changes each time you select a different option in this window.
4. **Selecting the Widget Visualization:** Enables you to change the visualization of the widget.
5. **Accessing More Options:** Displays a menu of additional options for the widget.
6. **Filtering the Widget (Filter Panel):** Lets you manage the filters that affect this widget. Learn more in the filtering page.

7. **Designing the Widget (Design Panel):** Provides a variety of options for fine-tuning the appearance of the Widget, including labels, legends, line types and more. Some of these options need to be turned on (**ON** ) for you to configure them. The options differ depending on the visualization (chart type).
8. **Update on every change:** Selected – Data changes are automatically and dynamically updated in the display. Not selected – you have to click **UPDATE** to display your latest changes in the widget. Clearing this checkbox may be useful when you are working with very large datasets where query times might be slower.

Fine-tuning a Widget

- ▶ To get an overview of the different widgets available, [click here](#).
- ▶ To learn more about adding data and fine-tuning the design of a specific widget, click on a widget from the list below.
 - ▶ INDICATOR
 - ▶ COLUMN CHART
 - ▶ LINE CHART
 - ▶ AREA CHART
 - ▶ BAR CHART
 - ▶ PIE CHART
 - ▶ POLAR CHART
 - ▶ SCATTER CHART
 - ▶ PIVOT
 - ▶ SUNBURST
 - ▶ TREEMAP
 - ▶ CALENDAR HEATMAP
 - ▶ TABLE

Choosing the Right Widget

The following guide will help you choose which widgets or charts to use in your dashboard. In addition to having lots of charts to choose from, you can also customize the design of each chart, as well as filter the data presented in the chart.

Indicator

Use the Indicator widget to display one or two numeric values as a number, gauge or ticker. You can also add additional titles and a color-coded indicator icon representing the value, such as a green up arrow or a red down arrow.



Gauge Indicator



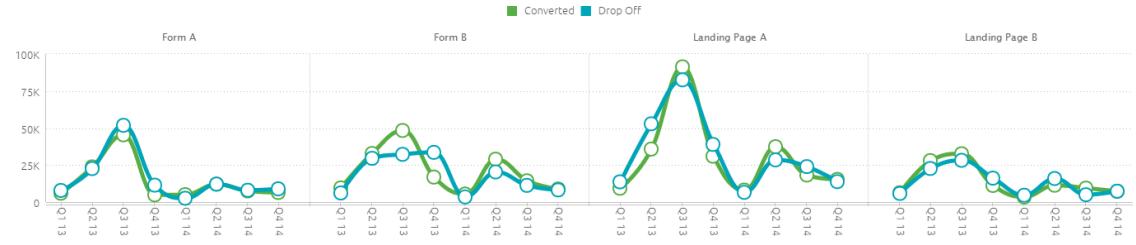
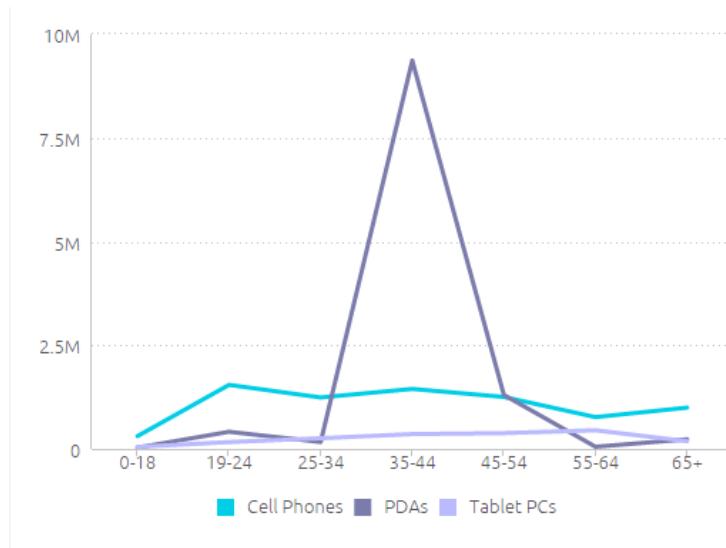
Numeric Indicator

Line Chart

The line chart is one of the most commonly used charts. You can use the line chart for various business cases, including:

Comparing data over time and viewing trends. Example: Analyze sales revenue for the past year.

Comparing changes over the same period of time for more than one group or category. Example: Analyze expenditures of different business units for the past year.

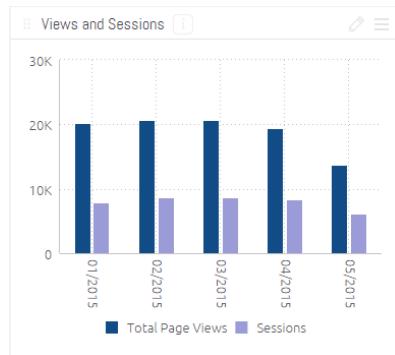


A line chart with a 'break by' category

Column Chart

The column chart can be used for comparing items and comparing data over time. The chart can include multiple values on both the X and Y axis, as well as a break down by categories displayed on the Y axis.

You can also combine the column chart with a line chart.



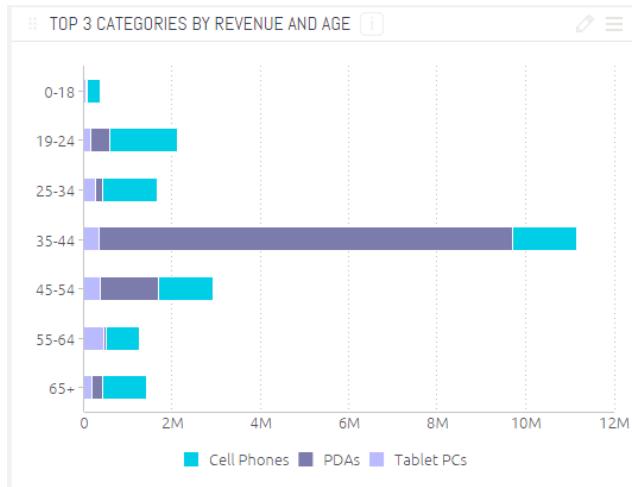
Column Chart



Column and Line Chart

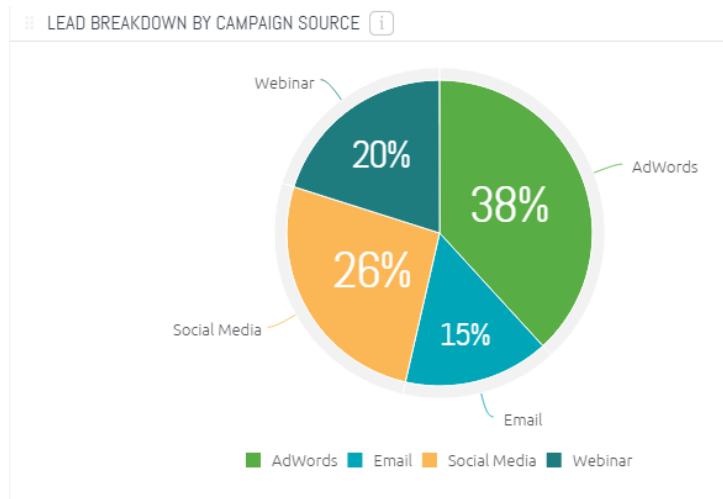
Bar Chart

Use the bar chart to compare many items. The bar chart typically presents categories or items (descriptive data) displayed along the Y axis, with their values displayed on the X axis. You can also break up the values by another category or groups.



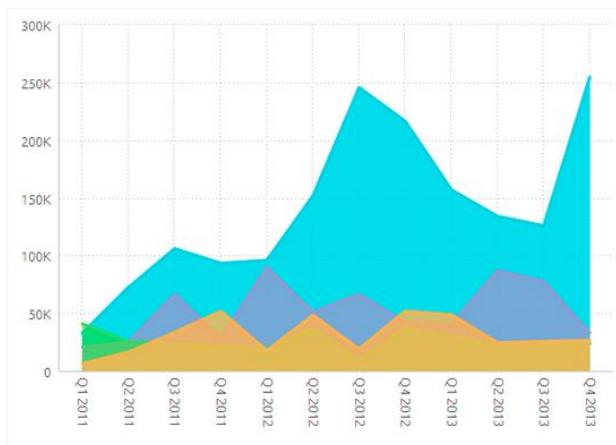
Pie Chart

Use the Pie chart to display proportional data, and/or percentages.



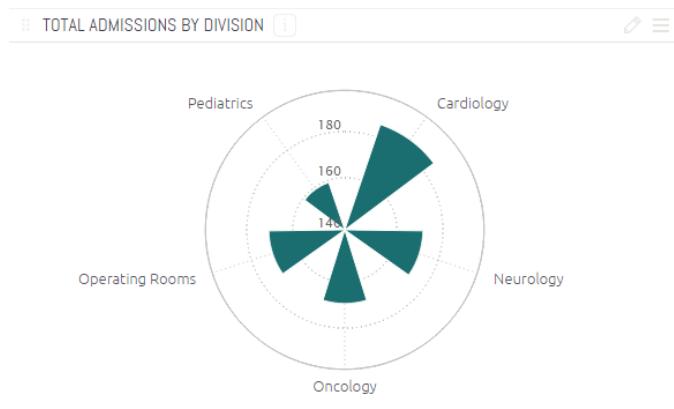
Area Chart

An Area chart is very similar to a Line chart except that the areas under each line are filled in (colored), and it is possible to display them as stacked. We recommend this chart to display absolute or relative (stacked) values over a time period.



Polar Chart

Use the polar (radar) chart to compare multiple categories/variables with a spacial perspective in a radial chart.



Pivot Table

Pivot tables are one of the most useful widgets for visualizing data.

You can quickly summarize and analyze large amounts of data.

Make use of additional features such as color formatting and data bars to enhance the visual aspects.

Age Range	Condition	Total Quantity
0-18	New	1,209
	Refurbished	292
	Unspecified	360
	Used	1,913
19-24	New	2,544
	Refurbished	592
	Unspecified	569
	Used	3,790
25-34	New	5,950
	Refurbished	1,401
	Unspecified	1,407
	Used	8,831

Example using color formatting

DIAGNOSIS	# PATIENTS	Avg Cost	Avg Stay (Days)
Bypass	116	\$781,957	8.07
Cardiac Arrest	120	\$796,833	8.92
Chemotherapy	119	\$804,966	9.17
Chronic Headache	122	\$780,387	10.03
Diabetes	154	\$788,317	7.88
Ear Inflection	122	\$764,097	6.81
EKG	131	\$790,577	8.63
Epilepsy	110	\$829,782	8.70
Hypoglycemia	125	\$803,405	7.93
Radiotherapy	168	\$768,751	8.45
Grand Total	199	\$788,621	8.42

Example using data bars

Table

The Table widget displays a broader view of your data, presenting raw and non-aggregated data in columns, with as much fields and metrics as needed.

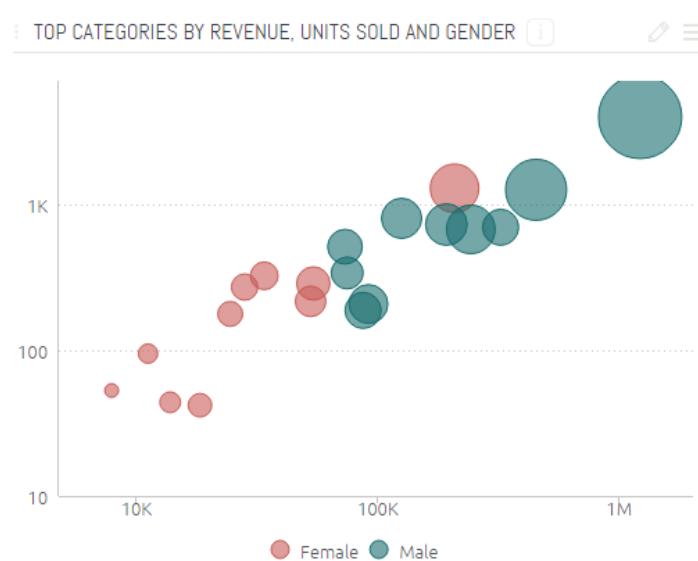
FirstName	Years in BirthDate
A. Scott	1962
Alan	1978
Alejandro	1983
Alex	1984
Alice	1972
Amy	1951
Andreas	1983
Andrew	1982
Andrew	1982
Andy	1977
Angela	1985
Anibal	1968
Annette	1972
Annik	1971
Arvind	1968
Ashvini	1971

« < 1 2 3 4 5 6 7 > »

Scatter Chart

Use the SCATTER chart to display the distribution and relationship of two variables. The circles on the chart represent the categories being compared (circle color), and the size or numeric data (indicated by the circle size).

Example: Compare revenue and units sold by gender.



Scatter Chart

Use a scatter map to visualize geographical data as data points on a map.

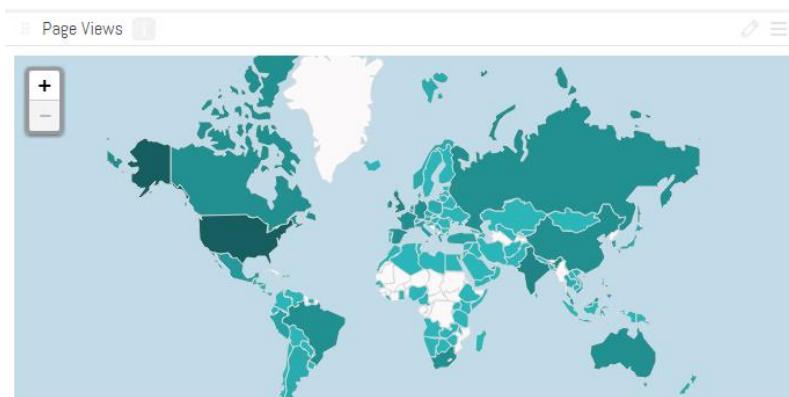
You can visualize up to two sets of numeric data using circle color and size to represent your data.



Area Map

Area Maps allow you to visualize geographical data as polygons on a map.

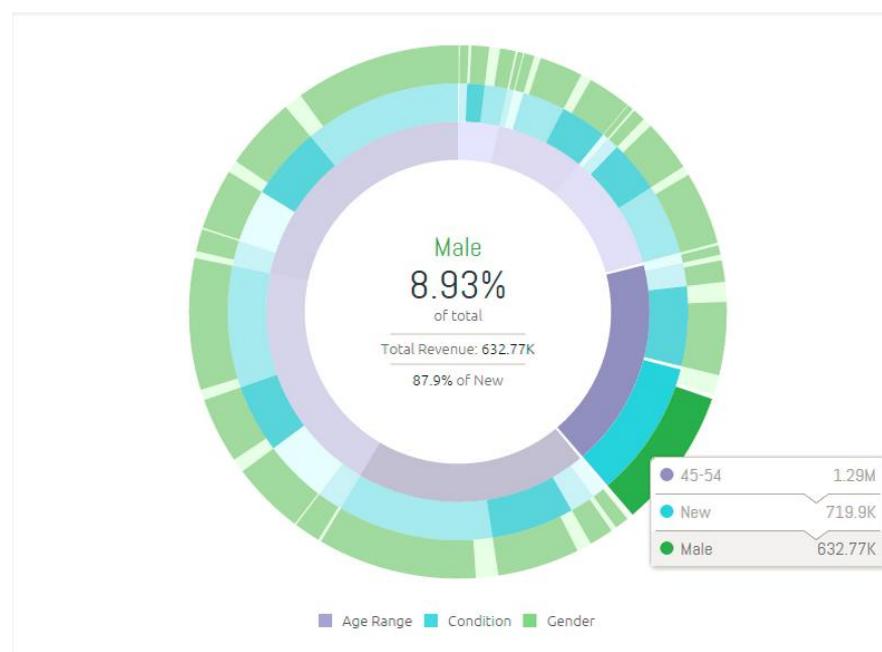
You can use your data to affect the color of the areas.



Sunburst Widget

The Sunburst widget is similar to a pie chart but is multi-dimensional. Whereas a pie chart combines one field and one numeric value, the

Sunburst widget can display multiple rings, one for each field. Each ring in the Sunburst shows a breakdown of its parent ring slice.



Treemap

The Treemap is a multi-dimensional widget that displays hierarchical data in the form of nested rectangles. You can use this type of chart in different scenarios, for example, instead of a column chart when you want to compare many categories and sub-categories.



Calendar Heatmap

The Calendar Heatmap widget visualizes values over days in a calendar-like view, making it easy to identify daily patterns or anomalies.

You can choose to display the data in a number of ways including 1, 3, 6 or 12 months at a time. In addition, you can display a classic or week view.



Adding Text Widgets

Add text widgets to create titles and texts that stand out, and when you want to add more descriptive explanations to the dashboard and surrounding visualizations.

In the widget's settings, you can find plenty of options for creating different text styles. Some examples of what you can do include changing the font color, selecting a background color, adding hyperlinks, and defining the text alignment.

To add a text widget:

1. Click on the Text Widget icon in the top menu.
A new Text Widget appears in your dashboard.
2. Type in your text.
3. Use the tools in the widget's toolbar to format your text. See some examples below.
4. When done, click outside of the widget area to save the text and formatting.

Examples

To select a font style:

1. Click anywhere in the relevant paragraph.
2. From the styles drop-down list, select one of the five predefined styles.

To add a hyperlink

1. Select the word or text fragment to which you want to add a link.
2. Click on the hyperlink icon, and enter the hyperlink.

3. You can also select which text to display instead of the selected (linked) text, and whether to open the link in the same window, or in a new window.

To change the background color:

1. Click on the paint bucket icon to open the color palette.
2. Select a color from the palette, or enter a custom color (Hexadecimal format).

Additional Widget Design Options

This section describes how to edit the widget further using the following features:

- ▶ Selecting Colors in Widgets
- ▶ Renaming Fields in Widgets
- ▶ Adding a Description to a Widget
- ▶ Formatting Numbers in Widgets
- ▶ Changing a Widgets's Visualization
- ▶ Combining Two Types of Visualizations in a Widget
- ▶ Adding Drill hierarchies to a Widget

Selecting Colors in Widgets

Different field values are represented by different colors. The Widget Designer automatically assigns these colors and displays them in the widget legend that is displayed in the Widget Designer and in the dashboard.

The Widget Designer provides the following options for manipulating these colors:

- ▶ **Single Color:** For selecting a specific color for all values
- ▶ **Range:** For selecting a range of colors
- ▶ **Conditional:** For defining colors based on conditional data.

Selecting a Specific Color – Color Tab

The following procedure describes how to assign a specific color to a data value that does not change. This color only applies to the widget in which it is defined.

To select a specific color:

1. Click on a color in the Widget Designer Data Panel to display a color picker window.
2. Select the **Single Color** tab.
3. Select a color and click **OK**.

Note: To select a custom color that does not exist in the palette, enter a hexadecimal value into the *#Custom* field.

Selecting a Range of Colors – Range Tab

The following procedure describes how to assign a range of colors to a field according to its value.

To select a color range:

1. Click on a color in the Widget Designer Data panel to display a color picker window.
2. Select the **Range** tab.
3. By default, the range configuration is set to 'Auto', which automatically assigns a color range.

Select **Manual** to manually configure the color range, as follows:

- ▶ **Change Left/Right colors:** Click the color rectangles at the right and left to change the start and end color of the range.
- ▶ **Customize range:** By default, the color range is set to match the minimum and maximum values of the data. Instead, you can manually set a minimum value. All values that are below this value will be assigned to the color on the left. Accordingly, selecting a Max value means that all values larger than it will be assigned the color on the right. By default, the 'Middle' color is grey. All values larger than this will start gradually getting the right color. All values smaller than this will start gradually getting the left color. Changing the 'Middle' value controls which value the color grey is assigned to.
- ▶ **Coloring Method:** Selecting **Min/Max** will display the full color range. Selecting **Min** will display all the colors between the minimum and middle values, and higher than the middle value will be gray. Selecting **Max** will display all values between the middle value and the maximum value, while all other values will be displayed in gray.
- ▶ **Transition:** Select **Steps** to define a distinct number of colors in the range. Select **Gradient** for a linear color range.

Defining Conditional Coloring – Conditional Tab

Define colors based on the value of the field as defined by a logical expression.

In addition, for INDICATOR Widgets, you can also define whether an icon appears in the widget and select the icon's color. For example, you can define a red arrow pointing downwards for a negative value.

To set conditional coloring:

1. Click on a color in the Widget Designer Data panel to display a color picker window.
2. Select the **Conditional** tab. Each row in this window represents a condition to which you can apply a value. If the condition is true, then that value is shown in the color that you select in this row. For example, one row can define that a negative value is displayed in red and another row can define that a positive value is green.
3. In each row, define an expression to be applied to the value of this field by selecting an operator and specifying a value, as shown below.
4. Select a color for the field.
5. If required, add and define additional condition rows by clicking **+ Add condition**.
6. In the dashboard, these conditions are evaluated in the order in which they appear in this window –from top to bottom. The first condition that is true, determines the color. Reorder the condition rows as necessary by dragging them up or down.
7. Click **OK**.

Renaming Fields in Widgets

The names of fields in widgets are taken from the raw data. To make these names more comprehensible, you can rename the fields in your widget.

To rename a field in a widget:

1. In the Widget Designer, right-click on a field, for example, in the X-AXIS, Y-AXIS, VALUES, CATEGORIES areas and so on.
2. In the menu, select **Rename**.

Note: Renaming a field only affects how it is displayed in the current widget. This field's name is not affected in the Data Browser or in other widgets in the dashboard.

Adding a Description to a Widget

You can add a textual description to widgets, which is handy when you need to add extra information or a note to dashboard viewers.

The description is visible when hovering over the "i" icon, when in view mode.

Note: If you want to add text that stands out with more formatting options, see Adding Text Widgets.

To add a description:

- ▶ Hover over the widget's "i" button when the dashboard is in edit mode, and click **Add Description**.

Note: You can also add the description from inside the Widget Designer screen, by using the "i" button next to the widget's title.

To edit a description:

1. Hover over the widget's "i" button when the dashboard is in edit mode, and click **Edit Description**.
2. Make your changes and click on the tick icon.

To delete a description:

1. Hover over the widget's "i" button when the dashboard is in edit mode, and click **Edit Description**.
2. Click **Reset**.

Formatting Numbers in Widgets

You can change the way numbers are displayed and formatted in widgets. For example, you can display a number as a percentage, and you can define the number of decimal places.

To format numbers in a widget:

1. In the Widget Designer, for example, in the X-AXIS, Y-AXIS, VALUES, CATEGORIES areas and so on, hover over, and click **123**.
Select how the number, currency or percentage appears. The preview area at the bottom of this window shows a preview of how each option will appear in the widget.

Locale Settings

The formats for dates, times and numbers in your dashboards are based on your computer's operating system or browser's locale settings (depending on which browser you use).

The following locales are supported:

- ▶ United States
- ▶ United Kingdom
- ▶ Israel
- ▶ Canada
- ▶ South Africa

- ▶ Australia
- ▶ Netherlands
- ▶ Germany
- ▶ Ireland
- ▶ Mexico
- ▶ France
- ▶ China
- ▶ Brazil

Notes

- ▶ You can override the default locale settings for all users, by updating a parameter via the Rest API.
- ▶ To use a locale file that is not included in the above list of countries, you can manually add a locale file in the following location: *C:\Program Files\Sisense\PrismWeb\client\resources\base\localization*.
[Click here](#) to access Locale files and view their codes.

Currency Settings

The default currency symbol used in the dashboard is the US dollar (\$), however you can change the symbol by either selecting another symbol from the drop-down list, or by typing in a different symbol. The list of currency symbols will include common currency symbols, the currency defined in your browser's locale settings, and the symbol that you added (if you did so).

After you select a different symbol, additional viewers will see the same symbol that you applied.

Changing a Widget's Visualization

You can change the visualization of a widget (for example, from a LINE CHART to a PIE CHART). When you change the type, all relevant values and definitions are transferred to the new visualization.

To change a widget's visualization:

1. Open the Widget Designer, as described in Opening the Widget Designer.
2. Click the Visualization selector, shown below, to display a drop-down menu of visualization options.
3. Select a different visualization, and click **APPLY**.

Combining Two Types of Visualizations in a Widget

In chart widgets (such as LINE CHARTS, AREA CHARTS, COLUMN CHARTS and BAR CHARTS), you can represent one or more selected field(s) using a different visualization than the visualization of the original widget.

For example, the following example shows a LINE CHART in which the **Total Revenue** is still represented as a LINE CHART (the default), but the **Total Quantity** is represented as a COLUMN CHART.

To select a different visualization for a field in the widget:

1. In the Widget Designer, in the **VALUES** area, right-click the field that you want to change.
2. Select **Series Type** from the menu. A menu of alternative visualization types is displayed.

3. Select the visualization to use for this field only.

Tip: If necessary, drag the field up or down in the Widget Designer to have a visualization brought to back or front.

Note: The chart keeps the same visualization type as was originally created. The Design panel still displays only the design options of the widget's original visualization type.

Adding Drill Hierarchies to Widgets

Dashboard viewers have the option to drill down in a widget, and get an in-depth view of a selected value.

While viewers can select a drill-down path from the complete list of fields, it's easier to select a commonly needed drill hierarchy from a short list. This is especially true when there is a lot of data, and the viewer needs to remember specific fields, and select them each time.

As a designer, you can add predefined drill hierarchies to widgets.

Note: Your Sisense administrator must first create the drill hierarchies before you can select and add them in the widget designer.

To enable drill hierarchies in a widget:

1. In the Widget Designer, click on the menu of the value (dimension), and select **Hierarchies...**
2. Select the checkboxes next to the drill hierarchies that you want to make available to dashboard viewers.

Note: Hierarchies with a lock icon will always appear in the widget. To unlock this option, please contact your Sisense administrator.

Disabling the Drill to Anywhere Option

As a designer, you can restrict viewers to predefined hierarchies only, by disabling the option to drill down freely to any field.

The drill-to-anywhere option is enabled by default for all users and widgets. You can disable the drill-to-anywhere option in the Sisense web app per widget (see below), or by user role in the REST API.

To disable the drill-to-anywhere option:

- ▶ In the widget menu, deselect the **Enable Drill to Anywhere** option.

As a result, the **Choose Another...**option is removed from the widget's **Drill** options.

The left image below displays the widget with the option to drill to anywhere. The image on the right displays the widget with only predefined drill hierarchies.

Creating a Continuous Chart with Missing Date Values

If your data is missing date values, you can add the missing dates to your chart to create a continuous data flow. In addition, you can select whether to display missing dates as gaps in your chart, or include null values as zeros, to avoid gaps, and maintain a continuous data flow.

For example, if your chart begins June 8th and ends June 20th, but there is no value on June 13th, you can choose to include the missing date on the the x-axis. You can also choose whether to continue the graph (create zero values) or cut the graph where there are missing values.

The default chart is missing June 13th.

After selecting the Continuous option, the chart includes the missing date(s).

In addition, you can select to continue the line, by treating null (missing) values as zero values.

To create a continuous chart:

1. In the X-AXIS panel, click on the menu, and select **Continuous**.
2. To treat null values as zero values, click on the menu of the VALUES panel, and select **Treat Null as Zero**.

Managing Widgets

Editing a Widget

When you first create a new dashboard (as described in [Creating a New Dashboard](#)) or a new widget (as described in [Adding Widgets to a Dashboard](#)), the Widget Wizard automatically guides you through Sisense Web's best recommendations for visualizing the data that you select.

Then, after a widget is created, you can fine-tune/edit it in the Widget Designer. The Widget Designer provides a variety of options for changing the widget's appearance and behavior.

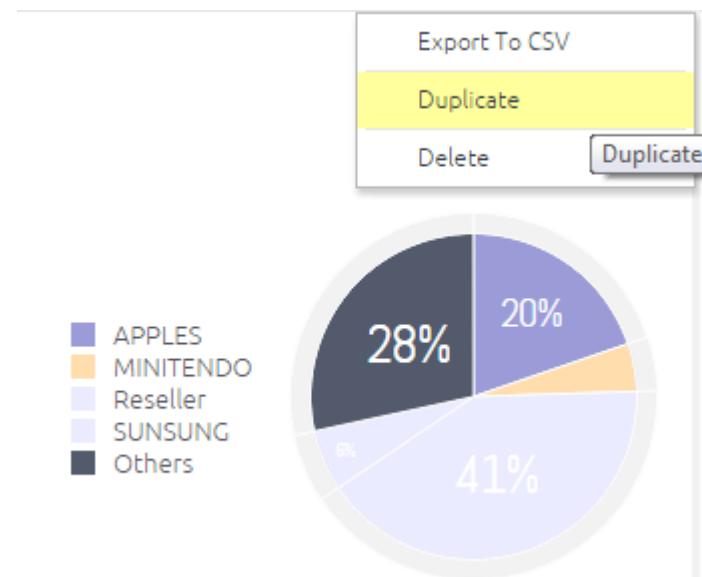
See [Fine-tuning a Widget](#) for more information.

Copying a Widget

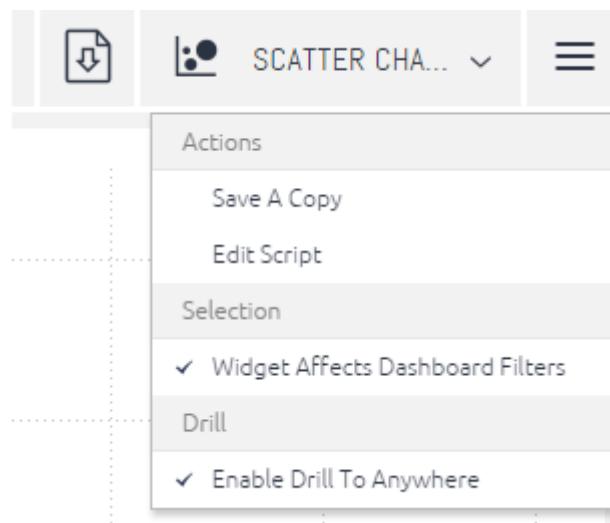
You can create a copy of a widget within the same dashboard or another dashboard in your list. This is useful for saving the original widget in the dashboard before making changes. The new widget is separate from the original so no changes to the new widget affect the original widget.

To copy a Widget within the same dashboard:

- ▶ In Dashboard view, click the widget menu  button in the top-right corner of the widget to be copied and select **Duplicate**.



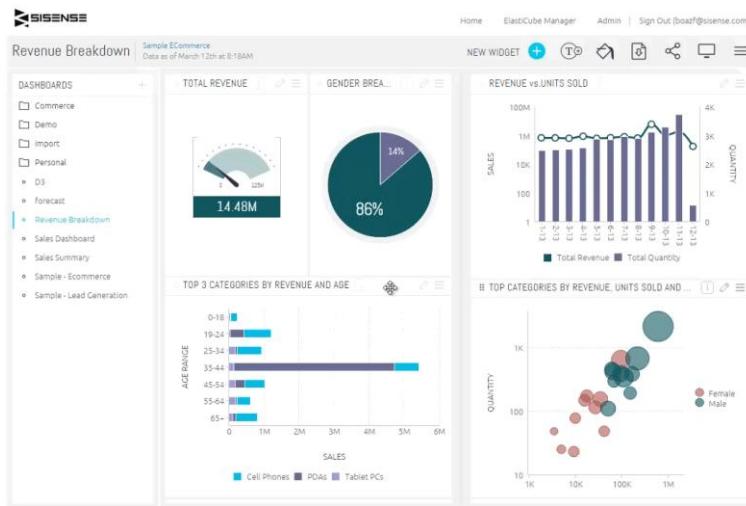
If you are editing the widget, click the widget menu  button in the top-right corner of the widget to be copied and select **Save A Copy**.



To copy a Widget to another dashboard:

Select the Header of the widget to be copied and drag it to the dashboard you want to copy it to.

Note: When you copy a widget to another dashboard whose data source is another ElastiCube, the Dashboard filters of the new dashboard will not affect the widget.



Deleting a Widget

To delete a Widget:

- ▶ Click the widget menu  in the top-right corner of the Widget and select Delete. Confirm the deletion.

Customizing the Dashboard Layout

This section describes how to rearrange your dashboard layout. By default, widgets are arranged in a dashboard one underneath the other. When you add a new widget, it is added at the bottom of the dashboard.

You can drag widgets one on top of another to split them horizontally.

The dashboard can also be organized in columns. By default a dashboard is created with a single column, but you can add up to four columns. Each column can contain multiple widgets, but a widget cannot span more than one column.

You can rearrange the order of the widgets, reorganize the widgets in columns, add/remove widgets from columns, and resize columns/widgets.

To move a widget:

- ▶ Drag the widget while clicking on the top part of the widget.

Layout Mode and View Mode

A dashboard can be viewed in either Layout mode or in View mode.

Layout Mode

Layout mode is the default mode in which you can add new widgets and rearrange them on the dashboard, as described throughout this section.



is displayed in the top-right corner of the screen when the dashboard is in Layout mode. Click this button to display View mode.

View Mode

View mode enables you to see what the dashboard looks like when a user is only viewing the dashboard. In View mode, the dashboard cannot be rearranged or edited.

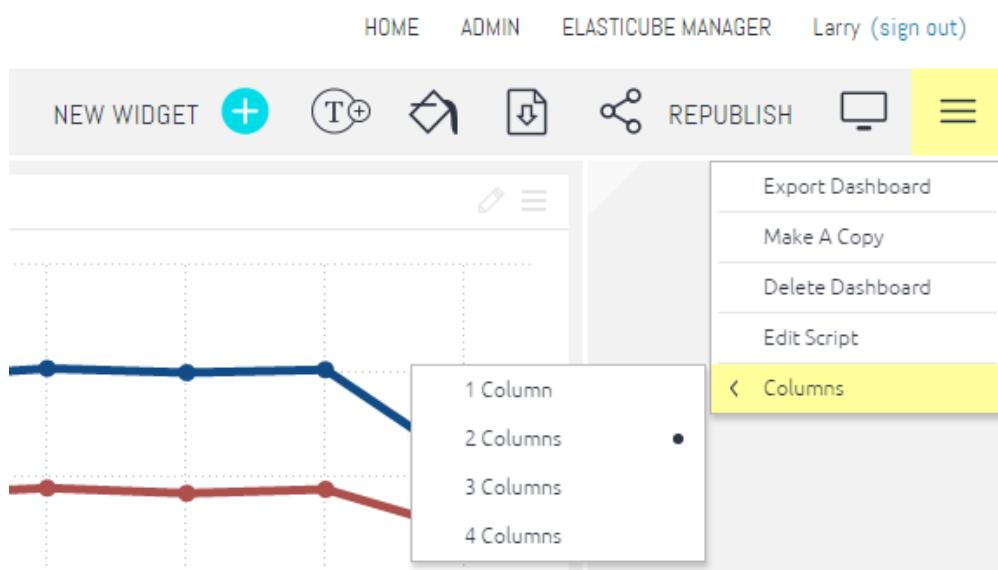


Is displayed when the dashboard is in View mode. Click this button to display Layout mode.

Adding Dashboard Columns

To change the number of columns:

- ▶ Click on the dashboard's menu, and select **Columns** and the number of columns.



Moving a Widget on the dashboard

To move a widget to a different column or position:

1. Make sure that you are in Layout mode (default mode). The

Layout mode indicator  is displayed in the top-right

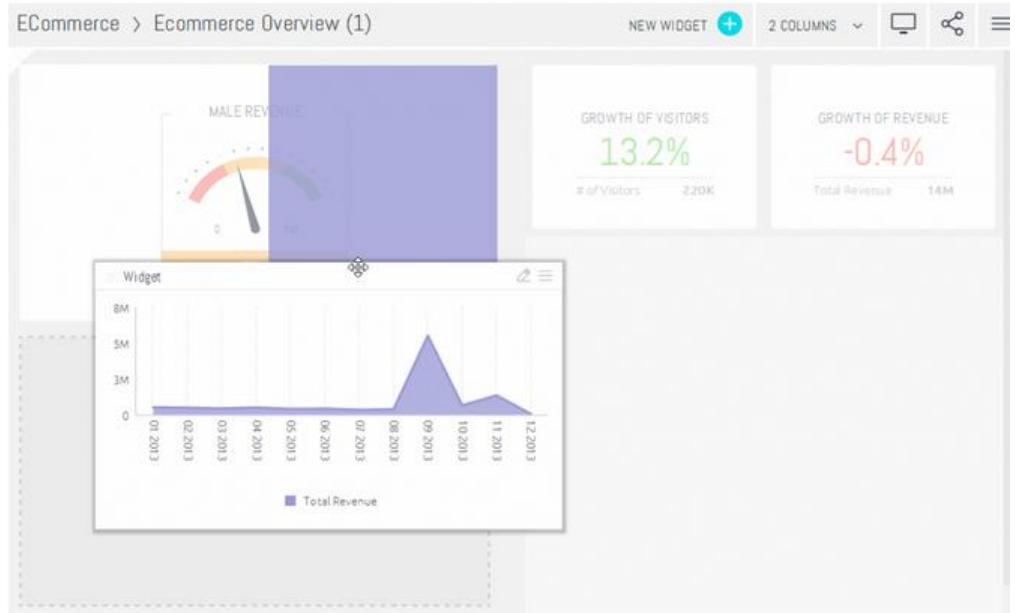
corner of the window. If not, click  to go from View Mode to Layout mode.

2. Drag and drop the widget into the desired position by dragging it by the top part of the widget.

Dragging a widget from one column to another:



The position of a widget can be split to contain multiple widgets within a single column by dragging one widget on top of another:



Placing Widgets Side-by-Side

You can place widgets side-by-side using one of the following methods:

- ▶ By putting each widget in a different column.

or

By dragging one widget on top of the other. Both these widgets will then be in the same column.

Resizing Widgets

Resizing the column height or width automatically resizes the widgets in the best possible way.

To resize a widget:

- ▶ Resize the dashboard column in which the widget is located by dragging the edge of the column right or left. All the widgets in

this same column are resized accordingly in an optimal manner.

- ▶ Resize the widget by dragging its edges right, left, up or down.

Refer to Ticker widgets for a description of how an INDICATOR Widget turns into a ticker type widget when you reduce its height.

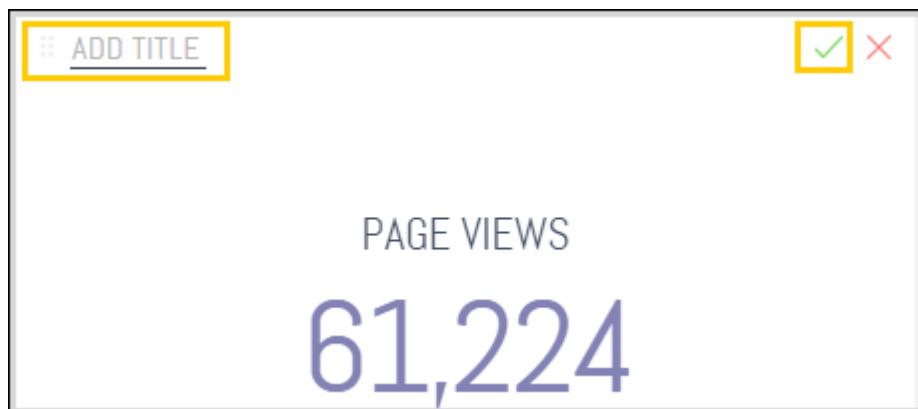
Adding Widget Titles

You can add titles to your widgets directly in the dashboard view.

If you already added a title when you created the widget in the widget designer, you can edit the title directly in the dashboard.

To add a widget title from the dashboard:

1. To add a title, click ADD TITLE at the top of the widget, and type in the title. If you are editing a title, click on the existing title, and type in the new title.



2. Click on the  icon to save the title.

Working with Ticker Widgets

If you resize the height of an INDICATOR Widget, then it automatically turns into a Ticker-type widget. An INDICATOR Widget that was a

numeric INDICATOR appears slightly different to an INDICATOR widget that was a gauge.

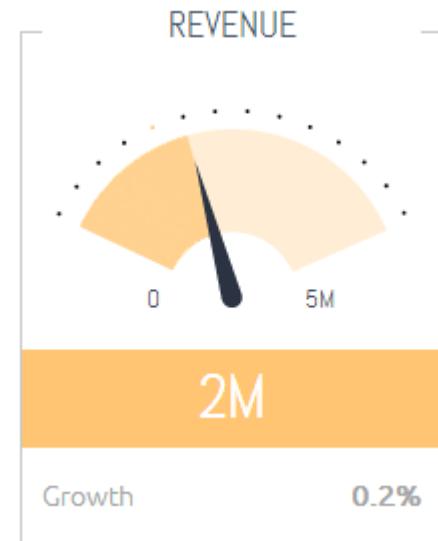
Regular Indicators

GROWTH OF VISITORS

13.2%

of Visitors

220K



Full size INDICATOR widget

Full size GAUGE widget

Ticker Indicators

| GROWTH OF VISITORS: **13.2%** | # of Visitors: **220K** |

| **2M** | REVENUE: **2M** | Growth: **0.2%** |

INDICATOR widget after resizing down

GAUGE widget after resizing down

Changing the Dashboard's Color Palette

You can easily change the color scheme of the dashboard to suit your professional or brand preferences.

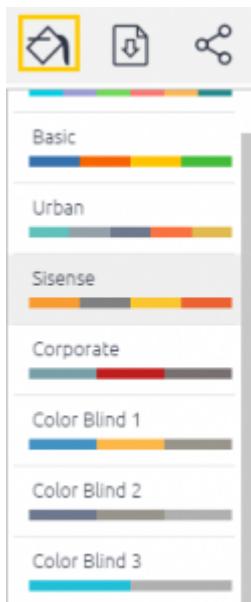
Note: The following procedure explains how to change the existing palette by way of selecting a different predefined palette. If you want

to create a custom palette, it is possible using the REST API. To learn more about creating a custom palette using the REST API, see this [support article](#).

To change the color palette:

- ▶ Click the paint bucket icon in the top menu, and select the preferred color palette. The default palette is *Vivid*.

Note: Three of the palettes are suitable for color blindness: *ColorBlind 1, 2, and 3*.

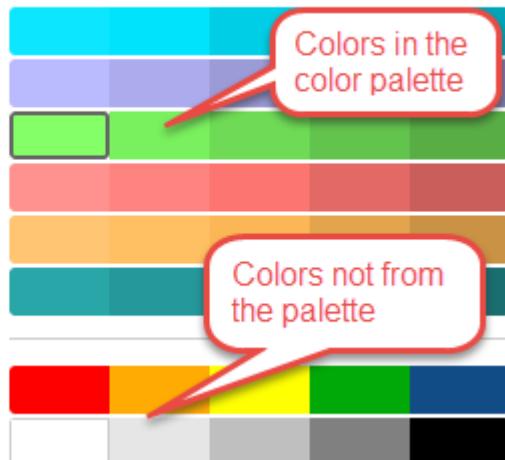


Note: When you change the color of an individual widget to a new color that is not one of the palette colors, the new color will be applied to the widget regardless of the applied palette. If you select a new color from one of the palette colors, then when changing the dashboard's palette, the color will change according to the new palette.

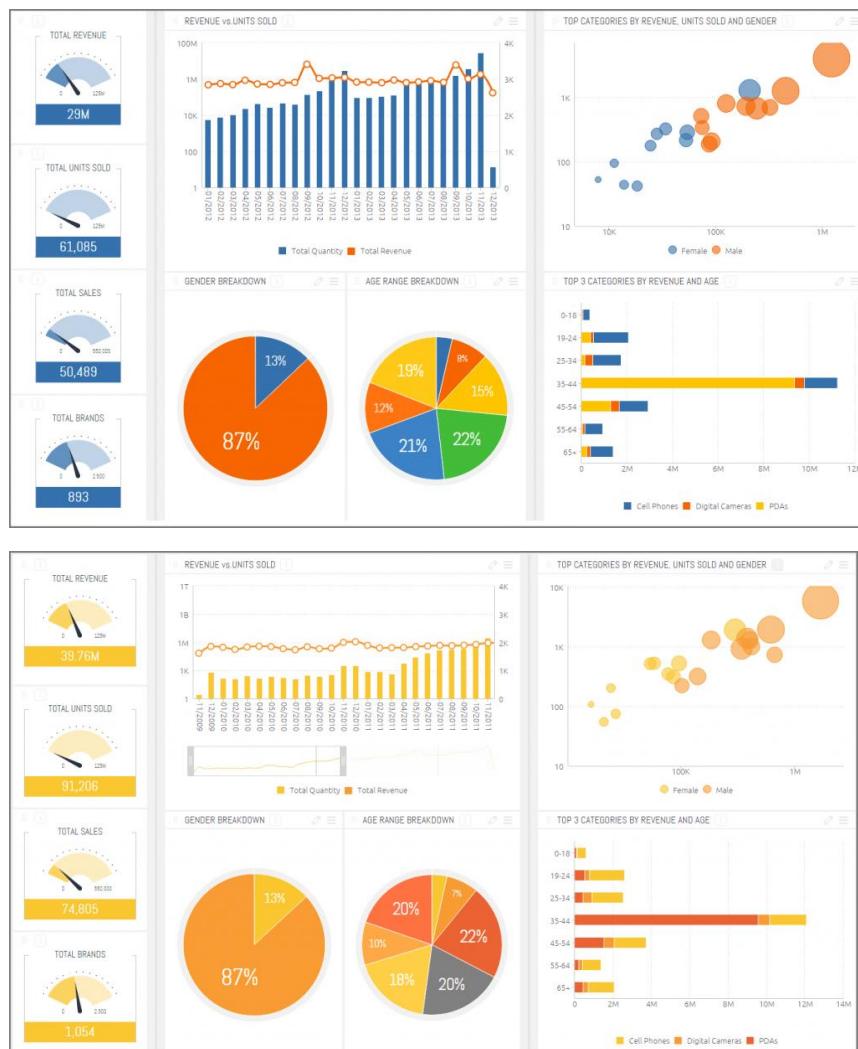
Recent Colors



Color Palette



Examples



Filtering Dashboards and Widgets

If you have dashboard editing rights, then you can define both dashboard filters and widget filters, as follows:

- ▶ **Dashboard Filters:** This type of filter affects all the widgets in the dashboard (except the widgets that you configured as independent). Designers can create dashboard filters and Viewers can view and interact with them.
- ▶ **Widget Filters:** A Designer can define which subset of data values is displayed in a widget. Viewers do not see these filters and cannot change them.

The following sections include procedures for:

- ▶ Creating dashboard filters
- ▶ Editing and deleting a dashboard filter
- ▶ Creating widget filters
- ▶ Creating dependent filters
- ▶ Working together with dashboard and widget filters
- ▶ Limiting Filters to Specific Values with Background Filters
- ▶ Locking filters
- ▶ Switching filters on and off
- ▶ Saving your default filters view

Creating Dashboard Filters

This section explains how to create a dashboard filter, using the different filtering options.

In this section you will learn how to:

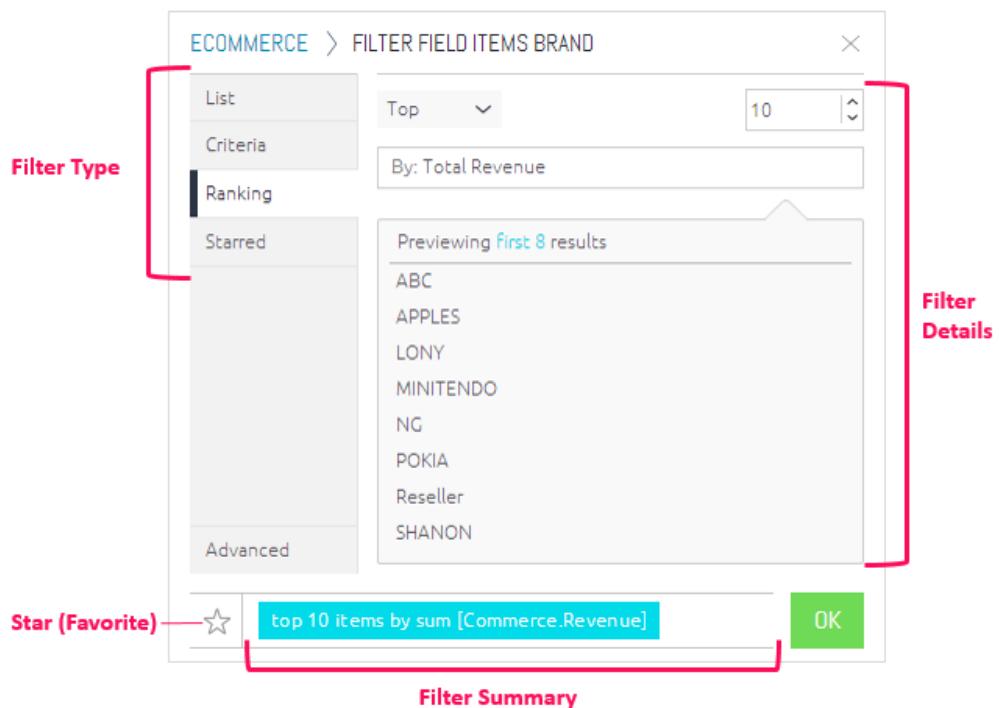
- ▶ Create a dashboard filter

- ▶ Create a list filter
- ▶ Create a text filter
- ▶ Create a numeric value filter
- ▶ Create a top/bottom ranking filter
- ▶ Create a time filter
- ▶ Create advanced filtering criteria
- ▶ Select how to update your dashboard during the design process

Creating a Dashboard Filter

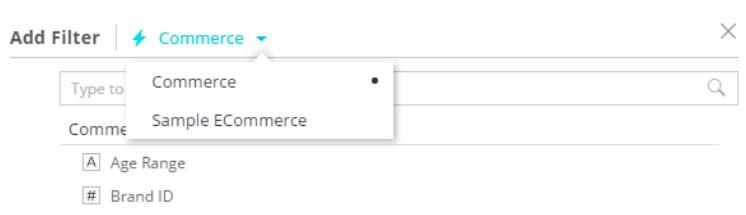
To create a dashboard filter:

1. Click **FILTER YOUR DASHBOARD** in the dashboard's right panel (if it's your first filter), or + if you are adding another filter.
- The Add Filter dialog box is displayed.



2. (Optional) If you have multiple data sources, select the data source that contains the fields you want to filter. Filters applied

to fields from one data source do not affect fields from another data source even on the same dashboard.



3. Select the field by which to filter.
4. Select the type of filter from the pane on the left, such as **List, Criteria, Calendar or Ranking**. The options provided in this window may vary according to the selected field type (Date, Numeric or Descriptive).
5. Select the required filtering details in the panel on the right to create one of the following types of filters:

Creating an Include/Exclude Filter

Creating a Text/Numeric/Date Filter

Creating a Top/Bottom Ranking Filter

Creating a Calendar Filter

These are explained in depth below.

6. Click the **Star Favorite**  button to save this filter. You can give your starred filter a unique name by clicking in the text field and editing the given name. This will be useful when you need to quickly add a new filter (with same field type) or update a filter to reflect the unique parameters that you defined in your starred filter.

Note: Your favorite filters are visible to shared users.

7. Click **OK** to filter the data in the Dashboard and add this filter to the FILTER panel on the right of the Dashboard.

After a filter has been added to the Dashboard, it stays displayed on the Dashboard's FILTER panel. Any changes that you make to the filter are immediately reflected in the Dashboard.

Tip: You can rearrange the filters in the Filters panel by dragging a filter above or below another filter.

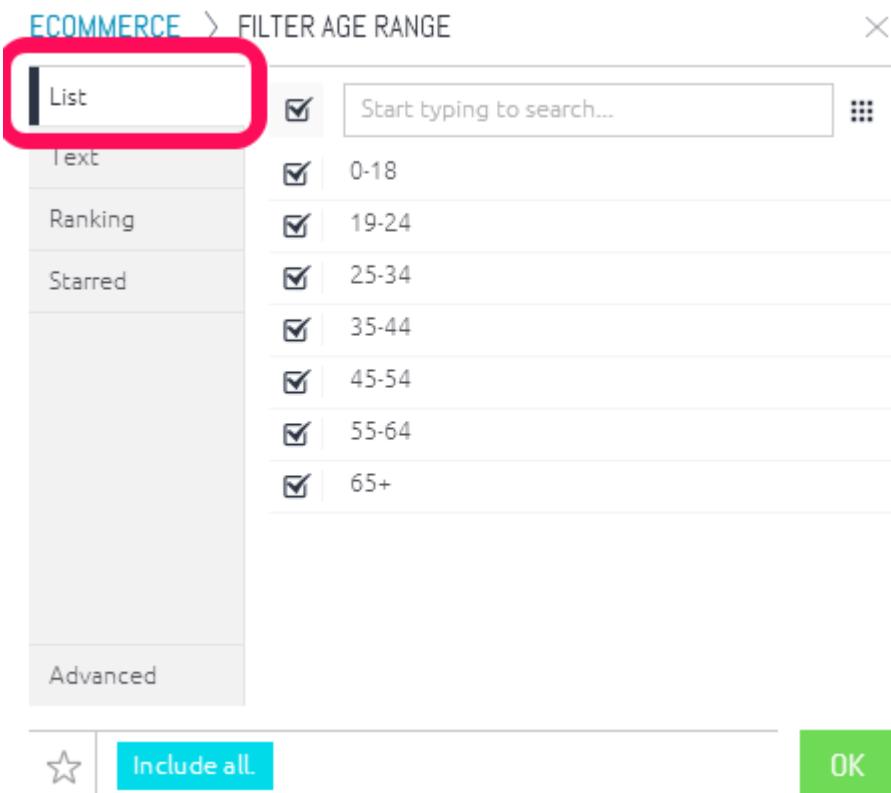
Creating an Include/Exclude Filter (List Filter)

This type of filter enables you to define whether selected field values are included or excluded.

To define a List filter:

1. Follow steps 1 - 3 of Creating Dashboard Filters.

2. Select the **List** option in the left panel.



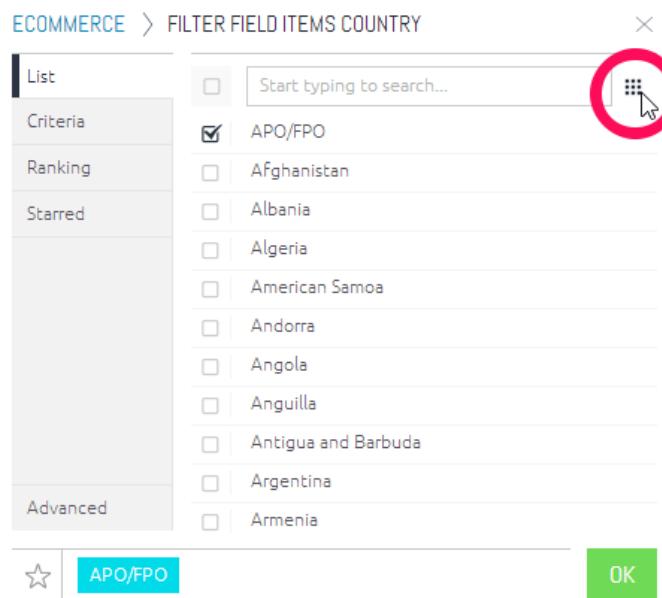
3. Select/clear the field values to be included/excluded in the Dashboard.
4. Click **OK** to filter the data in the Dashboard and add a List filter to the FILTER panel of the Dashboard.

Selecting Single Selection or Multi Selection

By default list filters are set to multi selection. To toggle between single and multi-selection click the button on the upper right of the list filter:

Multi Selection:

ECOMMERCE > FILTER FIELD ITEMS COUNTRY



List

- Start typing to search...
- APO/FPO
- Afghanistan
- Albania
- Algeria
- American Samoa
- Andorra
- Angola
- Anguilla
- Antigua and Barbuda
- Argentina
- Armenia

Criteria

Ranking

Starred

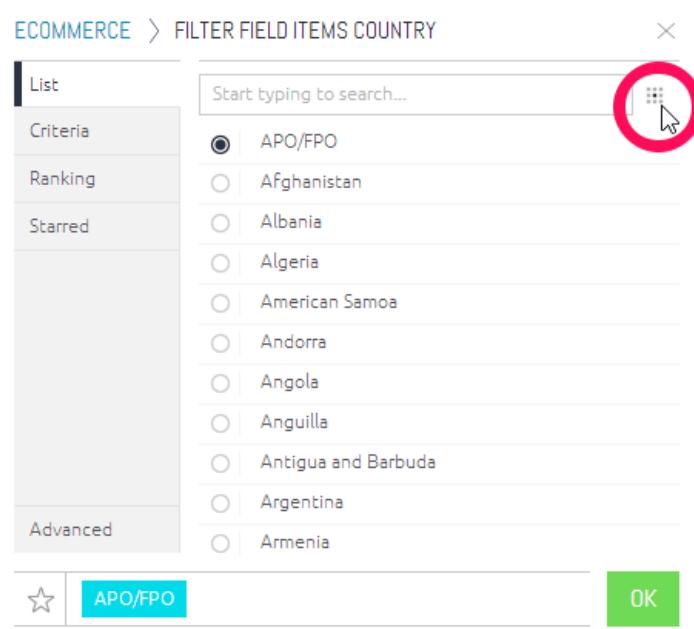
Advanced

APO/FPO

OK

Single Selection:

ECOMMERCE > FILTER FIELD ITEMS COUNTRY



List

- APO/FPO
- Afghanistan
- Albania
- Algeria
- American Samoa
- Andorra
- Angola
- Anguilla
- Antigua and Barbuda
- Argentina
- Armenia

Criteria

Ranking

Starred

Advanced

APO/FPO

OK

Creating a Text Filter

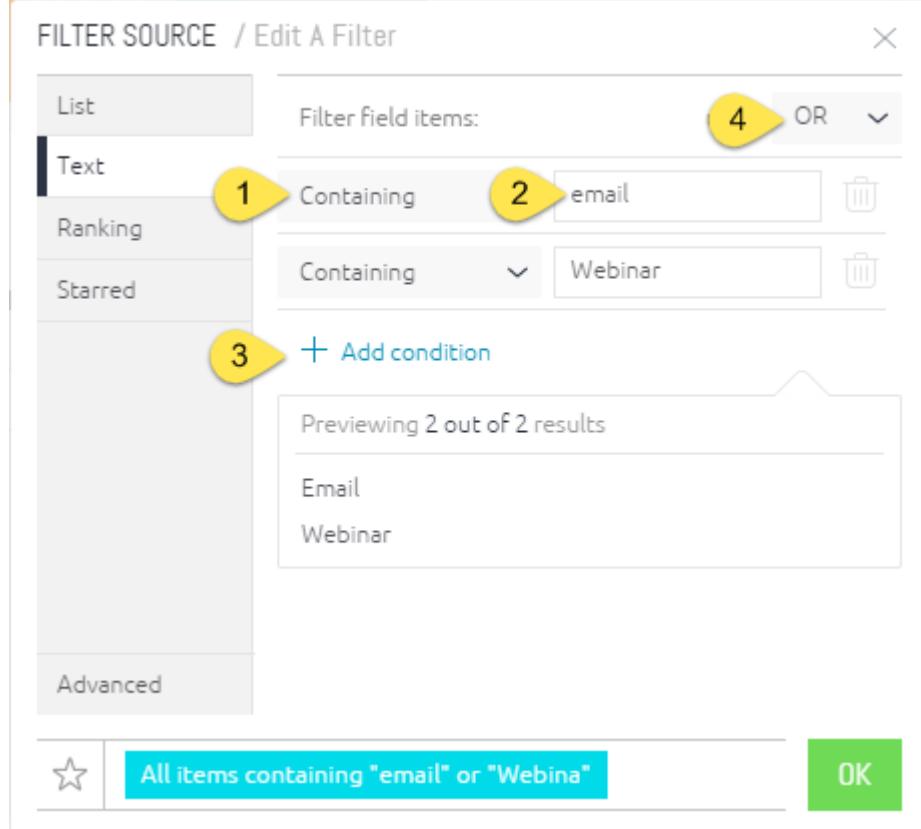
This type of filter enables you to filter according to text matching.

To define a Text filter:

1. Select the **Text** tab.
2. Follow steps 1–3 of Creating Dashboard Filters. Select a Textual field.
3. Select the operator, for example *Starts with* or *Containing*.
4. Type in the value by which to filter.

5. If you require additional conditions, click **+ Add condition**.

Select **AND** or **OR** to define how filter operators are combined.



Creating a Numeric Value Filter

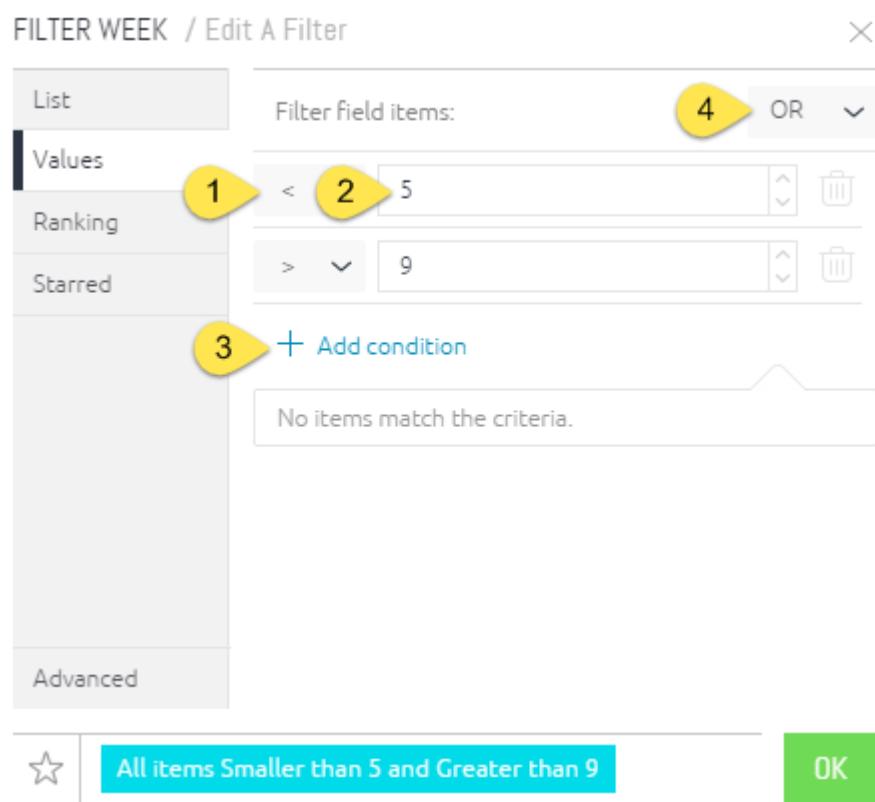
This type of filter enables you to limit your dashboards to specific value ranges. For example:

- ▶ Keep only sales above 100\$.
- ▶ Keep only product IDs between 1000020 and 1000030.

To define a Numeric filter:

1. Select the **Values** tab.
2. Follow steps 1–3 of [Creating Dashboard Filters](#).
3. Select the operator, for example *Equals* or *Greater than*.
4. Type in the value by which to filter.
5. If you require additional conditions, click **+ Add condition**.

Select **AND** or **OR** to define how filter operators are combined.



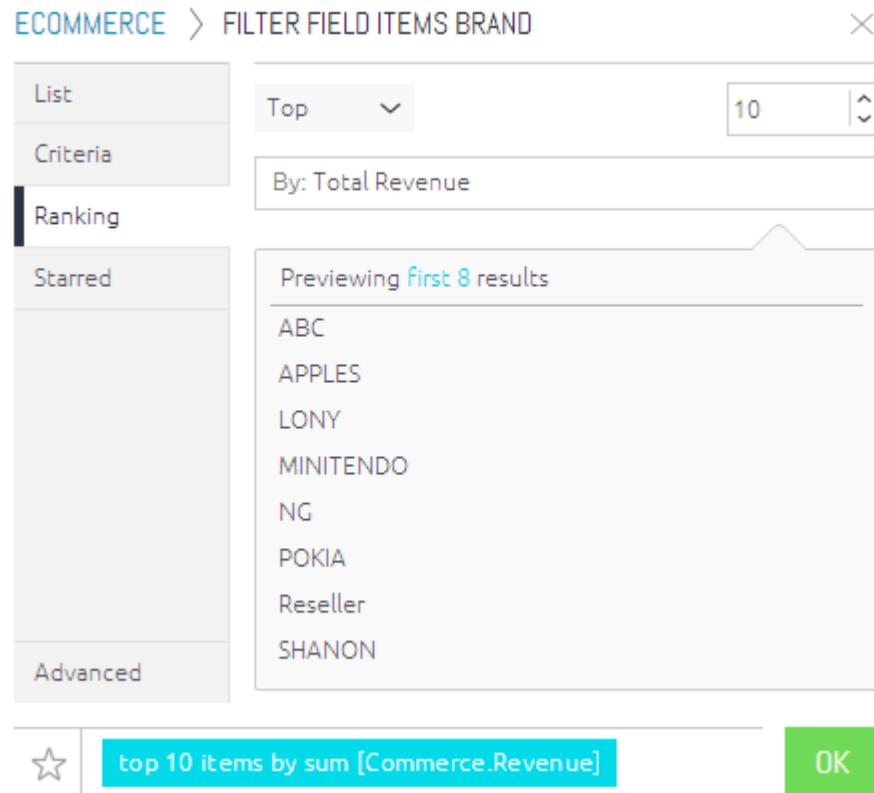
Creating a Top/Bottom Ranking Filter

This type of filter enables you to select whether to include only the top/bottom ranking fields.

To define a Ranking filter:

1. Follow steps 1 - 3 of Defining Dashboards Filters.

2. Select the **Ranking** filter tab.



3. Select **Top** or **Bottom** and the number of items to include.
4. In the **By** field, select a field by which to rank the fields. A list of the fields that match these definitions is displayed so that you can verify that the results are as you expect.
5. Click **OK** to filter the data in the Dashboard and add a Ranking filter to the FILTER panel of the Dashboard.

Creating a Time Filter

There are two types of time filtering options:

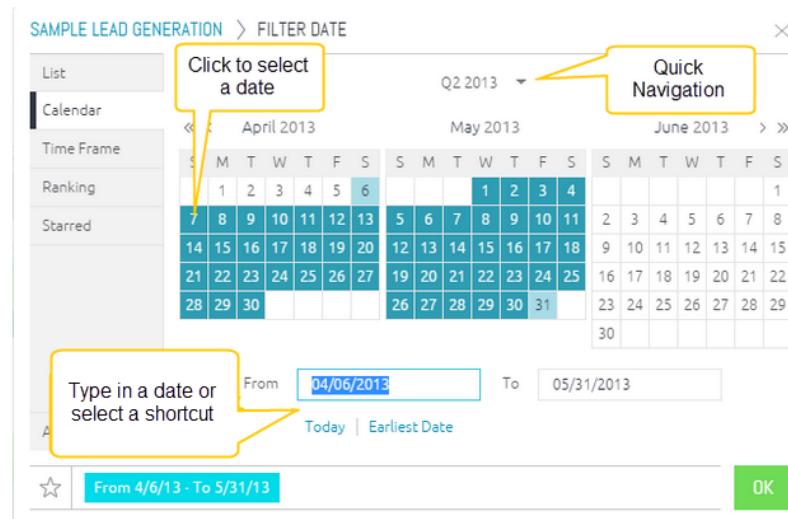
Calendar Filter

This filter enables you to select custom date ranges from a calendar.

To define a Calendar filter:

1. Follow steps 1 - 3 of Creating Dashboard Filters. (Select a date field).
2. Select the **Calendar** filter tab.
3. Use the quick navigation menu to jump to a point in time.
4. Select a date range using any of the following methods:
 - ▶ Click on a start and end date.
 - ▶ Type in the dates, or use a shortcut such as **Earliest Date** or **Today**.

Note: Earliest Date and Latest Date refer to the earliest and latest dates with data.



Note: Unlike other filters that you define, viewers will be able to change the type of date filter, providing more flexibility for viewing date ranges. For example, from Time Frame to Calendar, etc.

Creating a Dynamic Time Filter

This filter enables you filter dashboards to preset dynamic time frames, such as 'Last Year', 'Last 2 Years', '2 Year ago', etc.

To define a dynamic time filter:

1. Follow steps 1 - 3 of Creating Dashboard Filters. (Select a date field)
2. Select the **Time** filter tab.
3. Use the calendar to select a date range to filter by. Use the left calendar to select the starting date of the range, and the right calendar to select the end date.
4. Click **OK** to filter the data in the Dashboard and add a date filter to the FILTER panel of the Dashboard.

Advanced Filtering Criteria

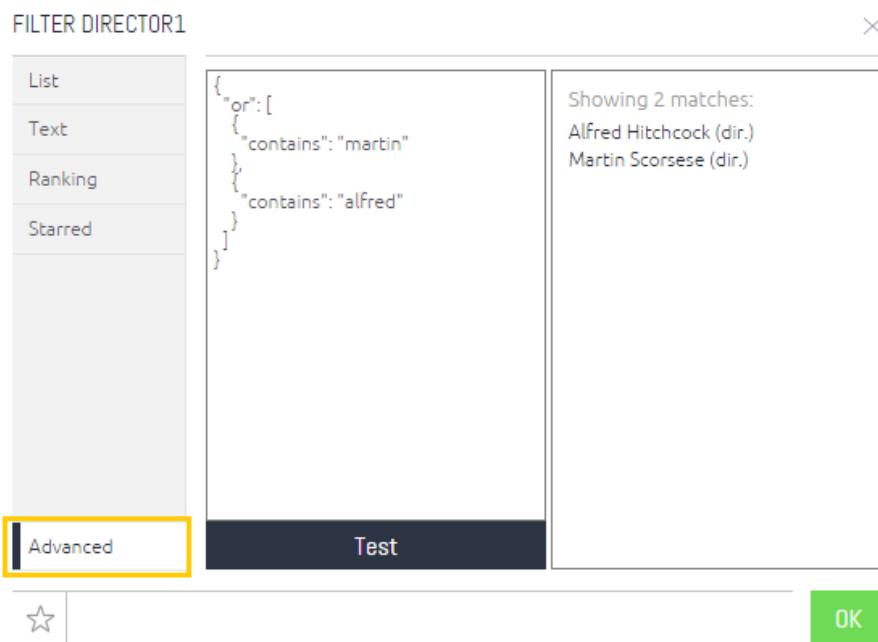
If you require advanced filtering options that are not included in the provided settings (for dashboard or widget), you can add or edit filters in the **Advanced** section.

To add or edit filtering criteria:

1. Click **Advanced** in the left panel.
2. Add or edit filtering criteria. Click **Test** to see the results of your new/edited filter.
3. Click **OK** to add the filter.

Example 1 – Adding an OR statement

The example below shows an OR statement being used to display multiple names. After clicking **Test**, the matches are displayed to the right of the script.



Examples using JAQL Queries

Example 2 – Filtering by Measures

This option lets you filter dimensions by measures, returning only the filtered members and associated values.

In this example, all products with a total price equal or greater than 1 million will be returned.

The query:

```
"datasource": "laptopsales",
  "metadata": [
    {
      "dim": "product"
    },
    {
      "dim": "price",
      "agg": "sum",
      "filter": {
        ">=": 1000000
      }
    }
  ]
```

Results include the relevant product names and price, for example:

- ▶ Product A, 1000000
- ▶ Product B, 1551246

Example 3 – Filtering by Dimension Attributes

This option lets you filter dimensions by filtered attributes of other dimensions, returning only the filtered values.

In this example, all customers that purchased *Product A* and *Product B* will be returned.

```
{
  "datasource": "laptopsales",
  "metadata": [
    {
      "dim": "customers",
      "filter": {
        "attributes": [
          {
            "dim": "products",
            "filter": {
              "members": ["Product A",
              "Product B"]
            }
          }
        ]
      }
    }
  ]
}
```

Note that you can add as many nested filters as needed.

Using Update on Every Change

When this option is on, any changes you make to the dashboard filters are updated immediately, and the dashboard is refreshed to reflect the changes.

If you are working with large or complex data sets that result in longer refresh times, then switch off this option. Make all your changes, and click the Update button when you are ready to update your dashboard.



Note: This option is available for designers only. Sisense viewers can make changes that will update automatically on every change.

Editing and Deleting a Dashboard Filter

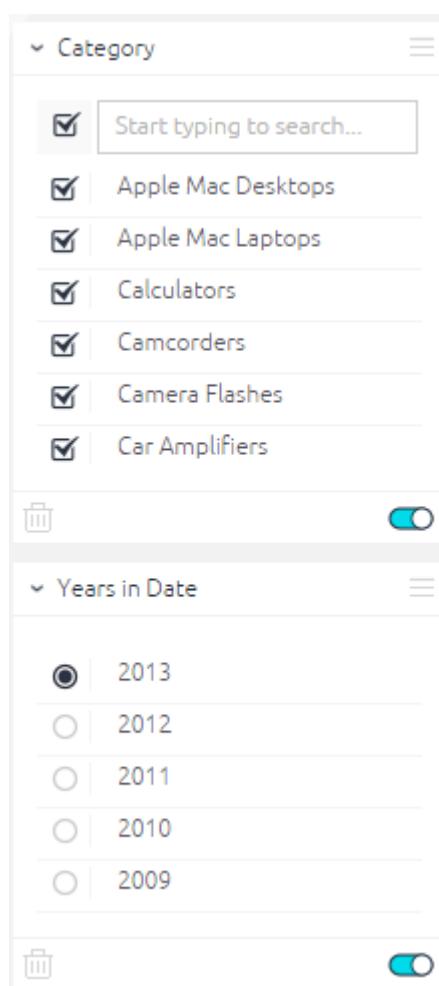
This section explains how to edit a dashboard filter and how to delete a dashboard filter.

Editing a Dashboard Filter

To edit a dashboard filter:

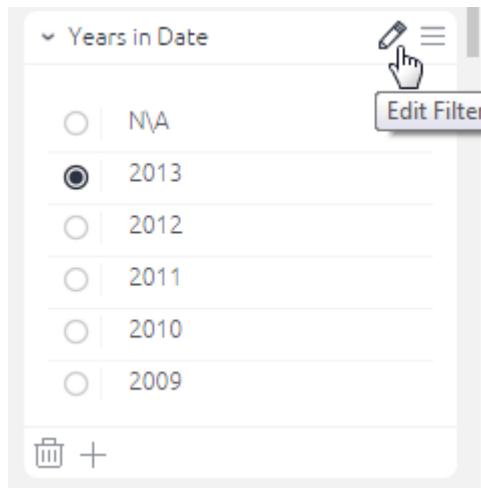
Do one of the following:

1. Make a different selection in the filter controls displayed in the FILTER panel on the right side of the dashboard, as shown below:



or

2. Click on the pencil icon next to the filter name (shown above) in the dashboard to display the Filter Definition window, as described in Creating Dashboard Filters.



Deleting a Dashboard Filter

To delete a filter:

Click the bin icon. To temporarily disable a widget's filter, you can toggle the filter on or off using the toggle switch. Delete the widget only if you do not need it any longer.

Creating Widget Filters

In the Widget Designer, you can define a filter for the data in a widget.

For example, you can create a filter that only shows the fields of relevant countries. The filters are not visible in the dashboard, nor can they be edited directly from the dashboard.

This section includes the following procedures:

- ▶ Creating a widget filter
- ▶ Adding a widget filter to an existing field

Note about duplicate filtering: When creating a widget filter for a field that is already filtered in a dashboard filter, the widget filter will override the dashboard filter.

Example: You have a dashboard filter, filtering the months field, with the values January and February selected. At the same time you create a filter for a widget, also for months, but this time March and April are selected. When both filters are applied in the dashboard, you will see data for March and April in the widget, according to the widget's filter.

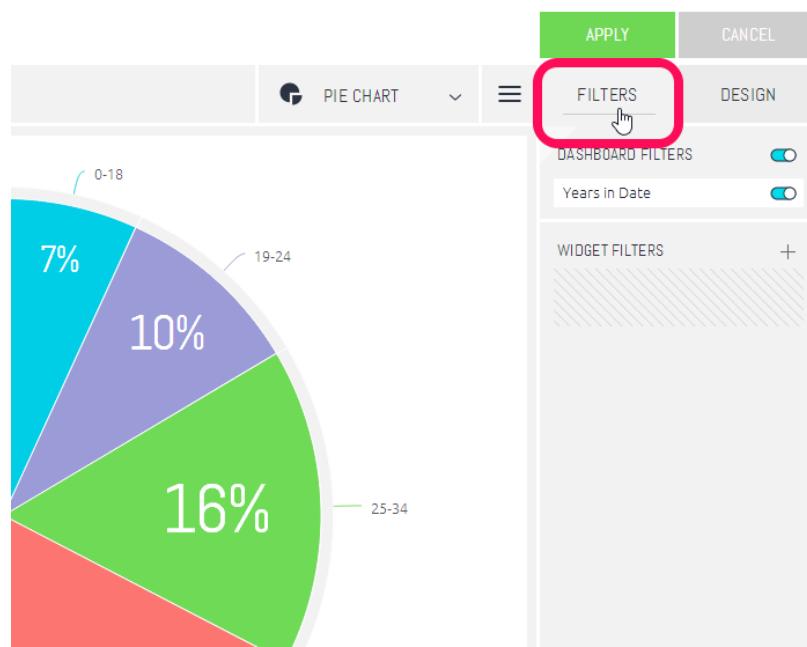
If your widget is filtered using measured values, then the measured value will override any other widget or dashboard filters you have for the same fields.

Creating a Widget Filter

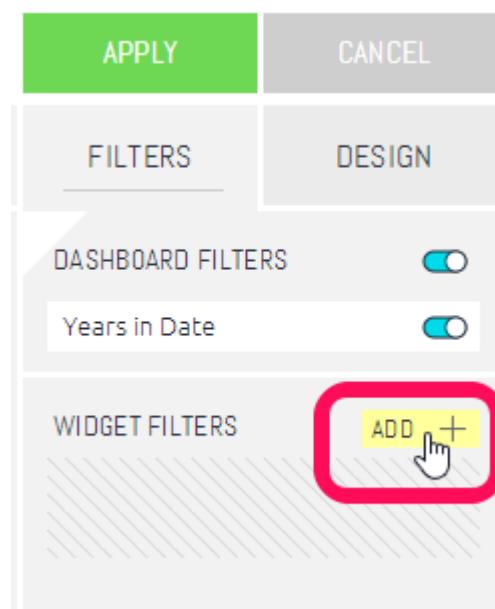
To add a widget filter:

1. Open (edit) this Widget in the Widget Designer, as described in Adding Widgets to a Dashboard.

2. Click the FILTERS tab in the widget designer.



3. The FILTERS panel is displayed showing the filters defined for this Widget. Click on the **ADD FILTER** button to display the Data Browser, which lists the fields in the ElastiCube. A Data Browser is displayed.



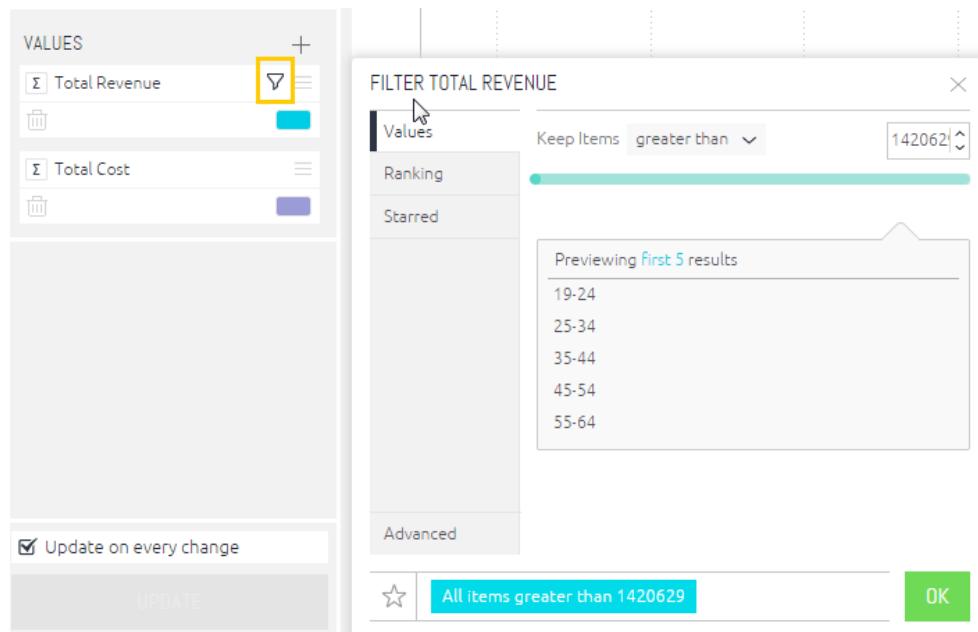
4. Click on a field in the Data Browser to add it as a filter. You can define multiple filters, each for a different field. These filters have an **AND** relationship between them. Each filter that you define appears on the **WIDGET FILTERS** panel on the right enabling you to easily edit it if needed. The procedure for defining a Widget filter is similar to the procedure for defining a Dashboard filter, as described in [Creating Dashboard Filters](#). The difference is that the widget filter only affects its hosting widget, and is accessible only via the widget designer, not directly from the dashboard.

Adding a Widget Filter to an Existing Field

Adding a new widget filter, as described above, lets you use any field as a widget filter. Sometimes you want to directly filter a field that is already part of the widget and visible in the data panel on the left.

To add a Widget Filter to an existing field:

- ▶ Hover over the field you want to filter, and click the FILTER icon.



Creating Dependent Filters

Use Dependent Filters to combine multiple dashboard filters into a hierarchy of dependent filters that affect each other from parent, to child, to grandchild and so on. When a user filters the parent filter, fewer options will be available in the child filter.

Example Use Case

The following example shows how using a Dependent Filter helps a user easily focus information in a dashboard.

This example describes an online travel agency with 250,000 hotels in its database. An agent wants to find a specific hotel, but does not remember its name. To narrow the results, the agent can define a Dependent Filter. This Dependent Filter is comprised of four sub-dependent filters defined with the following hierarchy:



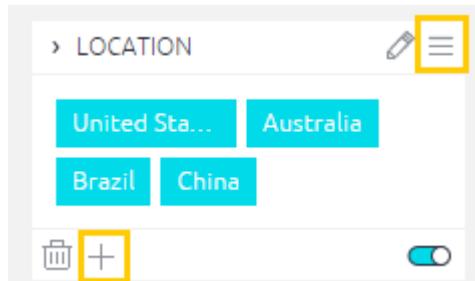
- ▶ **Destination:** Displays a list of States and Countries. In this example, the agent will select the Destination **Florida**.

- ▶ **City:** Because **Florida** was selected, only Destinations in Florida are displayed in the child filter, such as: Miami, Orlando and Tampa. In this example, the agent will select the City **Orlando**.
- ▶ **Resort:** Because **Orlando** was selected, only resorts in Orlando are displayed in the grandchild filter, such as Floridays, Buena Vista, Walt Disney World. In this example, the agent will select the Walt Disney World resort.
- ▶ **Hotel:** Because **Walt Disney World** was selected, only the Hotels in Walt Disney World are displayed in the great-grandchild filter.

Note: By default, filters are not dependent. Dependent Filters can be defined only from selection type filters.

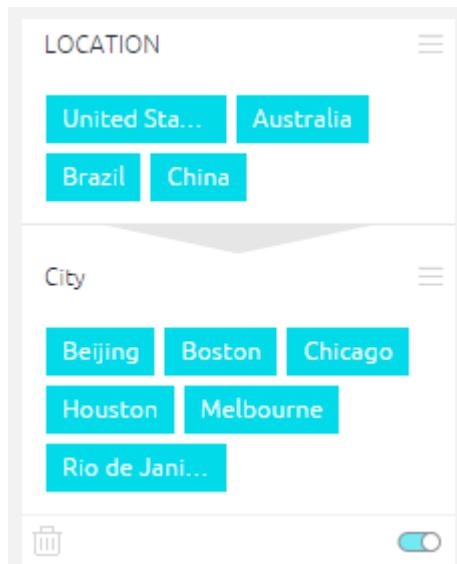
To define a Dashboard Dependent Filter:

1. The top level (most inclusive) filter of the Dependent Filter must first be defined in the usual manner (see Creating Dashboard Filters). This newly added filter then appears on the dashboard.



2. On the parent filter, click the **+** or click **≡** → **Add a dependent filter**.
3. Define the child filter in the usual manner (see Creating Dashboard Filters). The number of available options for selection in the child filter are less because of the selections in

the parent filter. The child filter appears on the dashboard with a down arrow indicating the dependent filter and its parent above.



4. Repeat this step for each new child filter to create as many sub-levels of Dependent Filters as you need.

Configuring How Filters Affect the Dashboard and Widgets

This section explains how dashboard filters work together with widget filters, and how you can configure different filtering behaviors and their effects on the dashboard.

The following procedures are explained below:

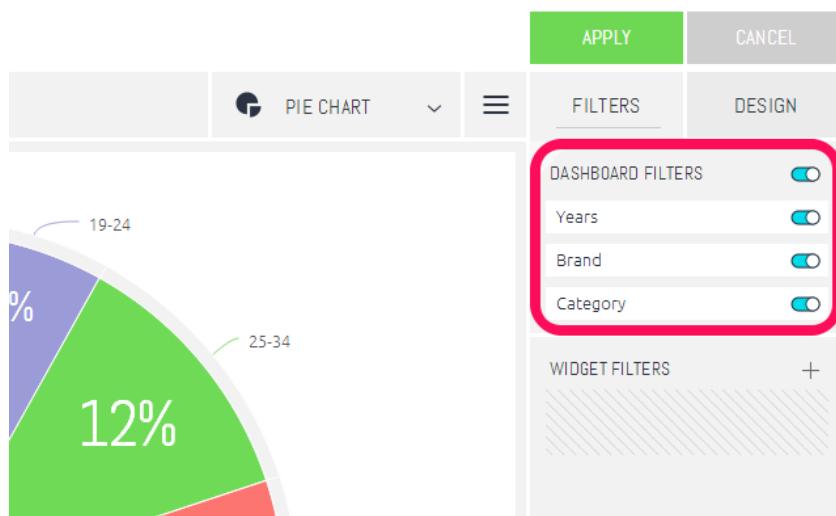
- ▶ Configuring how dashboard filters affect a widget
- ▶ Defining how a widget behaves when filtered
- ▶ Defining how a selection on the widget affects the dashboard

Configuring how Dashboard Filters Affect a Widget

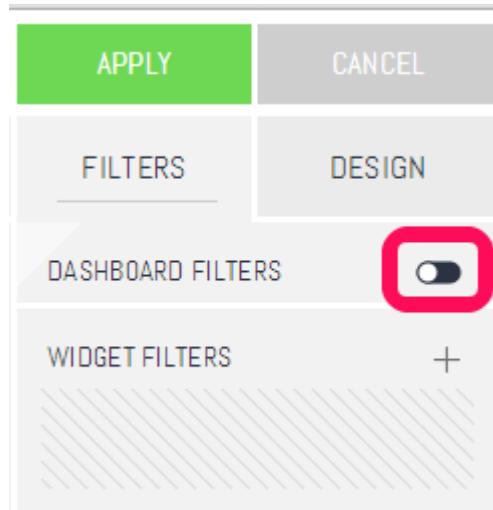
Dashboard filters affect all the widgets in the dashboard, except widgets that you have defined to be independent. An independent widget is not affected by dashboard filters.

To make a widget filter-independent:

1. Open (edit) this Widget in the Widget Designer, as described in Adding Widgets to a Dashboard. The top part of the **WIDGET FILTERS** panel on the right of the Widget Designer lists the Dashboard Filters that are defined for this Dashboard.

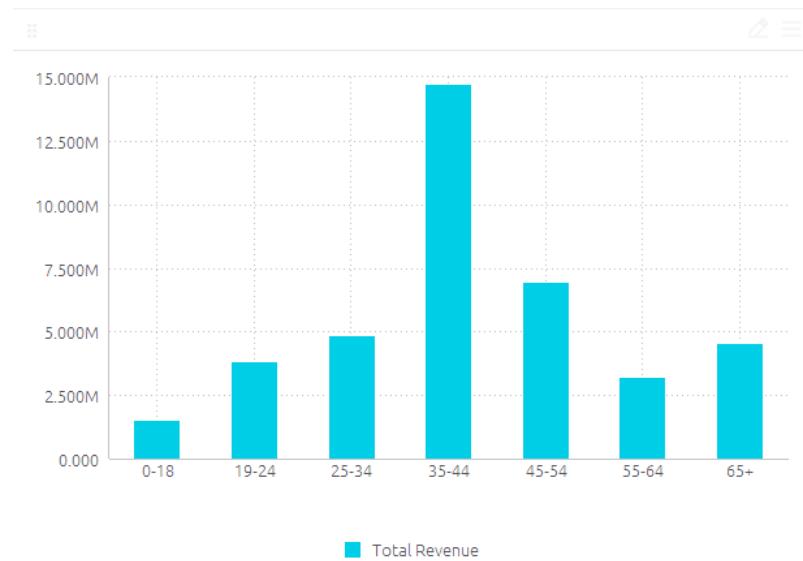


2. Turn off the Dashboard filters that you do not want to affect this Widget: The toggle on icon  (default) indicates that the the Widget is affected by this Dashboard filter. The toggle off  icon indicates that the widget is not affected by this Dashboard filter. You also have the option to make this Widget independent of all current and future Dashboard Widgets by turning the **DASHBOARD FILTERS** option off, as shown below:



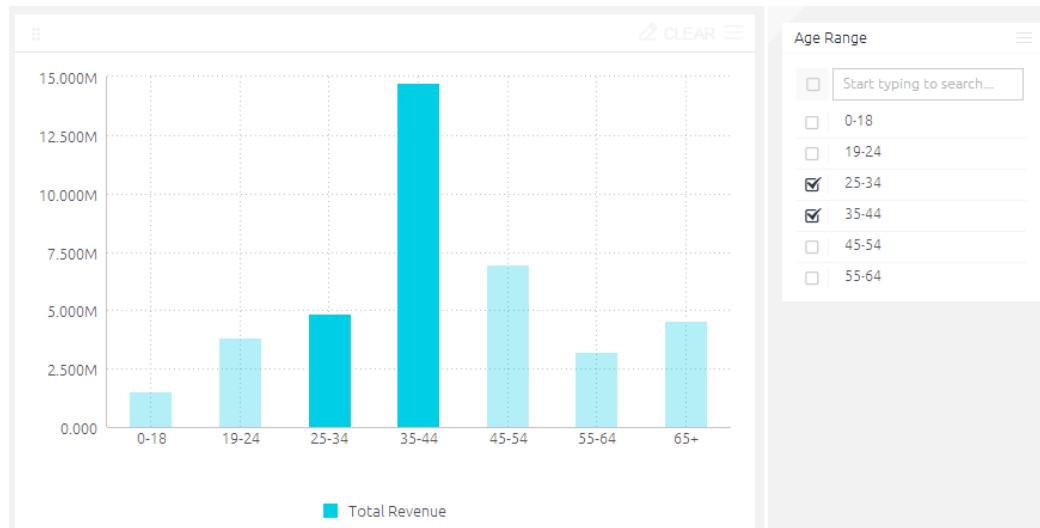
Defining how a widget behaves when filtered

When fields that are visible in a widget are filtered, it can be displayed in two ways – **SLICE/FILTER** or **HIGHLIGHT**. For example, let's look at this column chart, which shows revenue for different age categories:

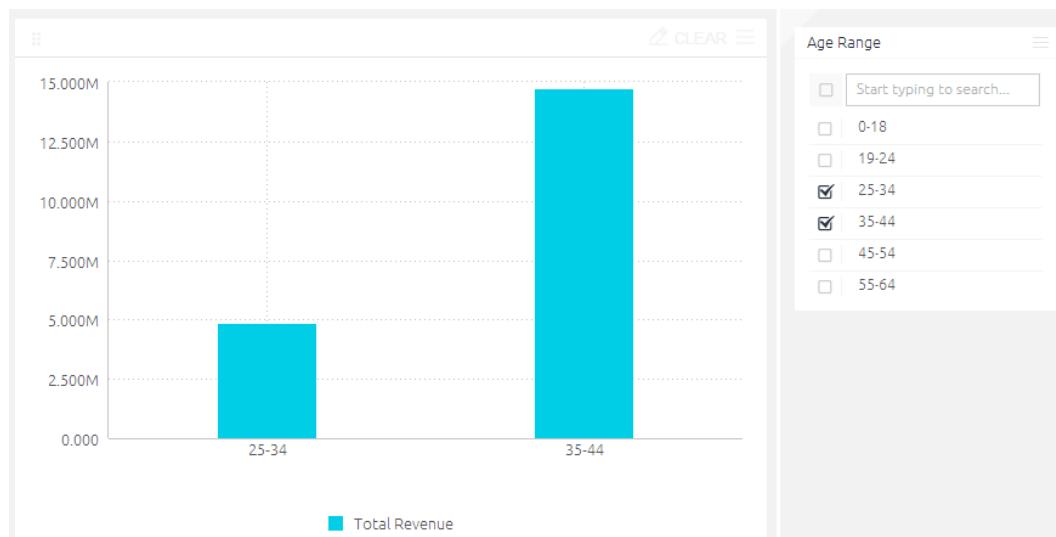


If the chart is set to **HIGHLIGHT** mode, filtering a subset of age categories will highlight them in the

chart:

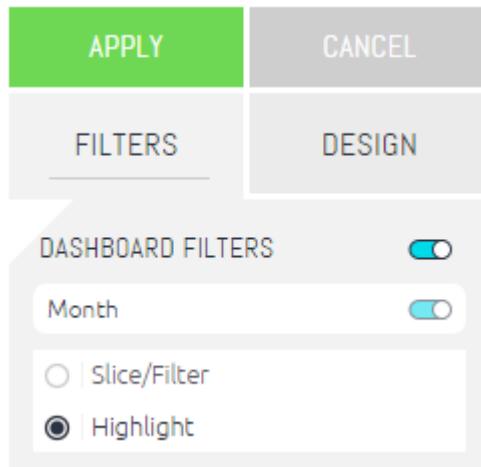


If the chart is set to **SLICE/FILTER** mode, the same filter will remove all non-selected age categories from the chart:



To define the filter behavior mode:

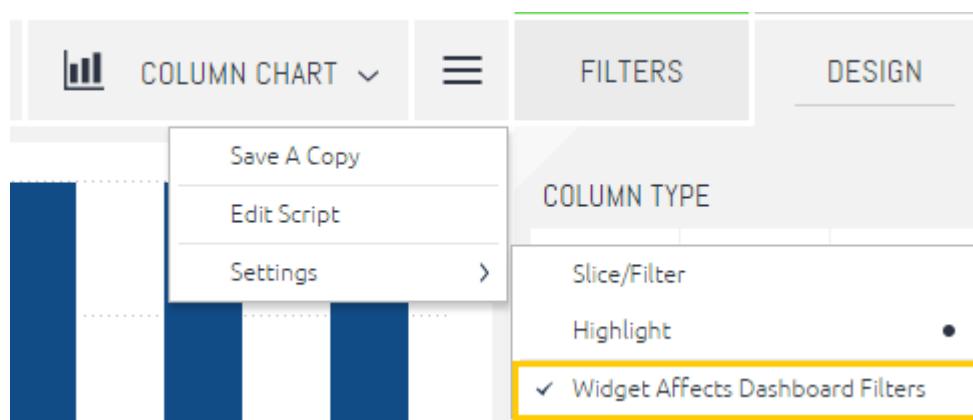
- ▶ Select **Slice/Filter** or **Highlight** in the Filters panel on the right side of the dashboard.



Defining how a selection on the widget affects the dashboard

When making a selection on a widget, a filter is added to the dashboard with the selected part of the chart as the applied filter.

To disable this option for a widget, open the widget in editing mode, and from the menu options, disable **Widget Affects Dashboard Filters**.



Limiting Filters to Specific Values with Background Filters

As a designer, you can define which of the fields in a filter will be visible to viewers. After defining which fields will be included in the

filter, viewers will be able to use the updated and restricted filter to further filter the dashboard or widget using the available fields.

This can be useful in cases where viewers only require selected data, and then they do not have to search through a long list of fields, or for security purposes in cases where you do not want to expose all the data.

Example 1 – Simple list filter

A filter includes a list of 10 countries. As a designer, you can limit the filter to include only five specified countries. Viewers will see the filter as having only five countries to include or exclude in the filter.

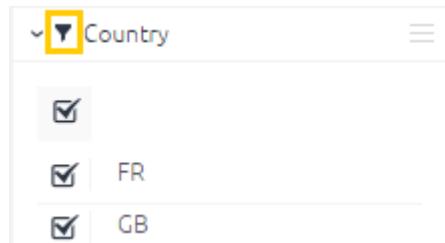
Example 2 – Conditional filter

Another example requires having only the top 5 countries by population included in the available list of countries. In this case, viewers will only see five countries, however, the list of countries will depend on the population sizes and any changes that may affect which countries are in the top 5 at the given time.

To create a background filter:

- ▶ Create a background filter using one of the following methods:
 - In a filter without selected values, open the filter's menu and click **Create background filter**. Define your filter settings, and click **OK**.
or
 - If you made changes to your filter (either by editing the filter or by selecting values directly in the filter), you can open the filter's menu, and click **Set as background filter**.

The background filter is indicated by a filter ▾ icon.



Note: The icon is not visible to viewers.

To edit a background filter:

1. In the filter's menu, click **Edit background filter**.
2. Update the filter settings, and click **OK**.

To remove a background filter:

- In the filter's menu, click **Remove background filter**.

Locking Filters

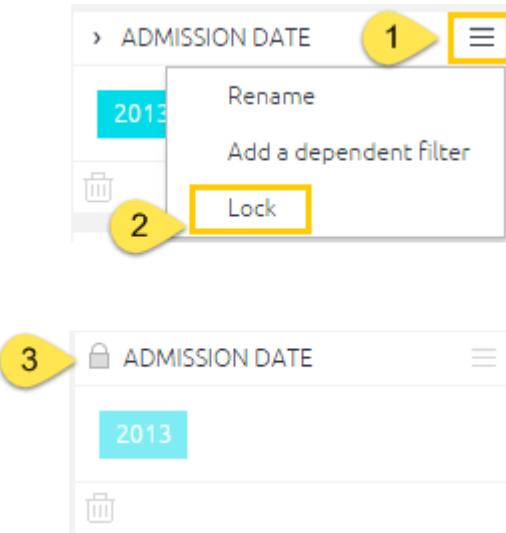
As a designer, you can lock a filter, preventing viewers from making any changes. The viewer will be able to see the defined filter settings, but the filter will be disabled for editing.

To lock a filter:

1. Click on the filter menu.
2. Click **Lock**.

A lock icon appears to indicate that the filter is locked.

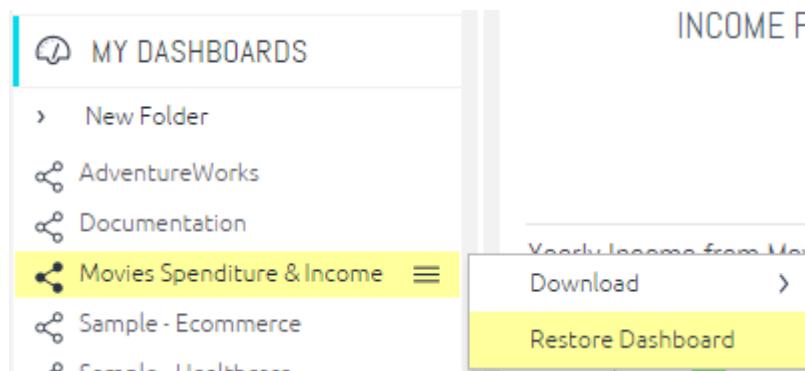
To unlock the filter, click on the menu, and select **Unlock**.



Restoring Filters

As a Designer, if you changed the dashboard's filters, you can revert to the latest copy shared by the dashboard's owner at any time. When you restore a dashboard, local changes to a dashboard by a Viewer are overridden and the default dashboard is restored for all users.

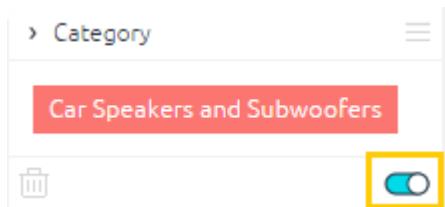
- ▶ Under the MY DASHBOARDS list, click on the menu of the dashboard and select **Restore Dashboard**.



Switching Filters On and Off

You can easily toggle filters on and off using the toggle switch. Use this option to compare states (with and without the filter), or to temporarily disable a filter, rather than deleting it.

Toggling a filter on or off in the dashboard affects the data in all the widgets, unless a widget is defined not to be affected by dashboard filters (see how dashboard and widget filters work together).



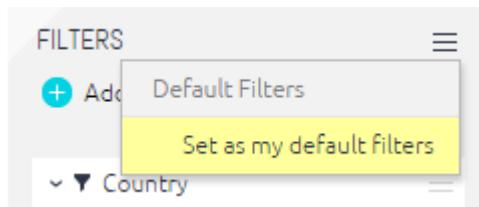
Saving Your Default Filters View

You can save the current state of your filters and their settings at any time. The current state includes the existing filters, their configuration, and the order in which they appear in the filters panel.

After making changes to any of the above settings, you will be able to restore your filters to their previously saved state.

To save your current set of filters:

- ▶ In the Filters menu, click **Set as my default filters**.

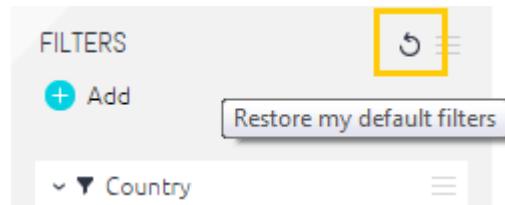


To restore a saved filters set:

- ▶ Click the restore icon next to the Filters menu.



SISENSE



Using Formulas

Formulas are custom calculations performed on one or more fields in the data. They offer an important way to analyze results and express business logic. This guide provides an overview to working with formulas along with important tips and examples. Sisense formula capabilities are designed around several principles;

- ▶ Create complex business calculations without IT or technical knowledge.
- ▶ Easily combine fields from different data sources together.
- ▶ Customize formulas to reflect specific criteria and conditions.
- ▶ Work with raw data without the need to summarize data sets before creating formulas.
- ▶ Instantly recalculate formulas based on any filter, variable or level of granularity.

The table below provides a reference to the main formula functions available in Sisense.

Goal	Function	Types and Syntax
Perform calculation based on criteria	Measured value	Value Filters: ≠ , =, >, <, between Text Filters: Contains, Doesn't Contain, Doesn't End With, Doesn't Start With, Ends With, Start With, Equals, Not Equal List Filter: Include, Exclude Ranking Filters: Top, Bottom Ranking Time Filter: Date and Calendar
Combine data/apply simple mathematics	Aggregate functions	Operator: +,-,*,/ Aggregate: Sum() Average: Avg() Count: Count(), DupCount() Range: Max()/Min()
Summarize data	Statistical function	Central Tendency: Median(), Model, Largest() Std Deviation and Variance: Stdev(), Stdevp(), Varp(), Var() Quartile and Percentile: Quartile(), Percentile()

Goal	Function	Types and Syntax
Accumulate data	Rolling sum/average	Sum to Date: YTDSum(), QTDSum, MTDSum() Avg to Date: YTDAvg, QTDAvg, MTDAvg()
Compare Time or Trends	Time functions	Past Periods: PastYear(), PastQuarter(), PastMonth(), Next(), Prev() Growth Trend: Growth(), GrowthRate() Time Difference: YDiff(), QDiff(), MDiff(), DDiff(), HDiff(), MnDiff(), SDiff(),

The following sections explain how to create formulas, and describe the features of the formula editor.

- ▶ Using the Formula Editor
- ▶ Creating and Editing a Formula
- ▶ Reusing Formulas
- ▶ Using Quick Functions
- ▶ Creating Formulas Based on Criteria and Conditions (Filters)
- ▶ Building Formulas with Functions
- ▶ Function Reference

Using the Formula Editor

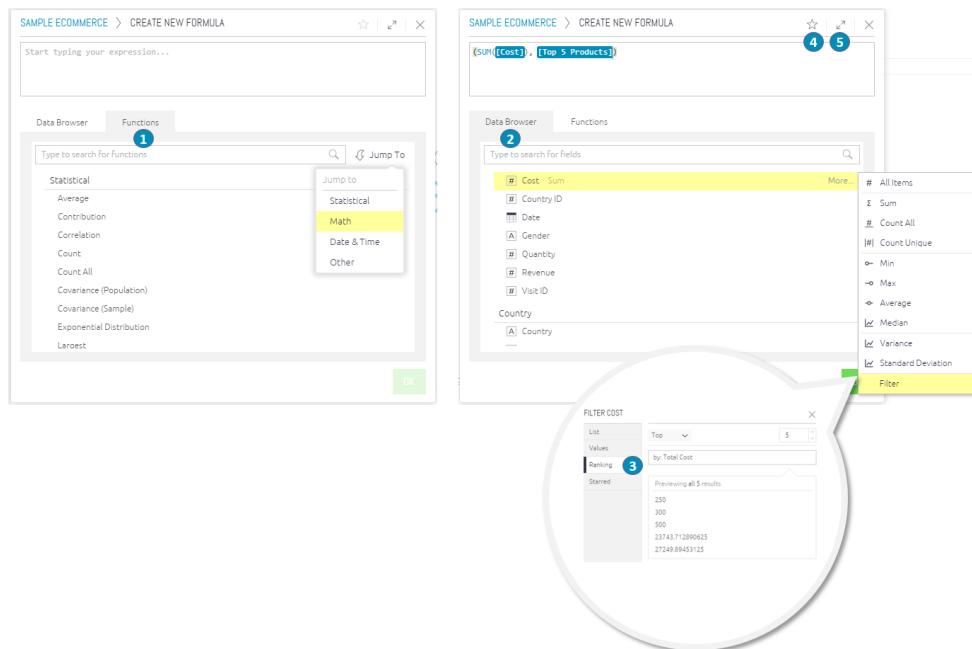
The Formula Editor is where you define formulas for your widgets.

If you are familiar with the Formula Editor, see the Quick Reference Guide. For a detailed guide with examples, continue reading below.

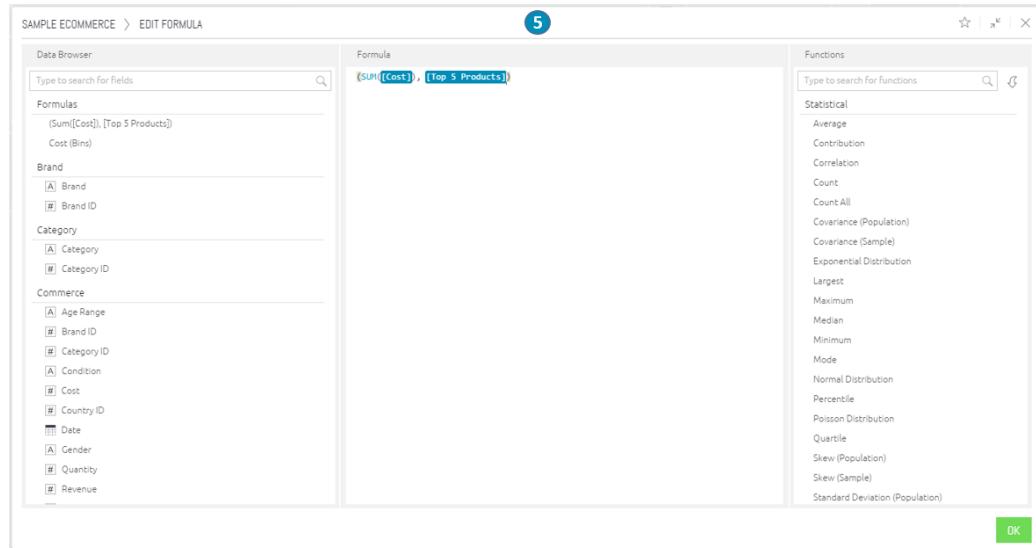
To open the formula editor:

- ▶ Do one of the following:
 - For a new widget, click **Select Data**, and then  to open the formula editor.
 - For an existing widget, click on the edit formula button  to open the formula editor.

The formula editor has two tabs, the **Data Browser** to select fields and the **Functions** tab to select formula operations. You can create a formula combining one or more *function*, *field* and *filters*. The diagram below highlights the main components of the formula panel.



1. **Functions** are operations which perform different calculations, for example a sum. Use the 'Jump To' menu or the search box to quickly find the formula you need.
2. Fields in the **Data Browser** are variables contained in the data set (ElastiCube). Clicking on a field in the data browser will include it as part of the formula.
3. **Filters** can be applied to restrict formulas based on criteria.
4. **Starring** is a way to save a formula for later use.
5. The formula editor window can be expanded by clicking the expand button at the top right.



Creating and Editing a Formula

The Data Browser enables you to define formulas (freeform expressions) that define the values and filters of a widget.

A rich variety of functions are provided for you to use in the formula that you define.

To define a formula:

1. Open the formula editor in the Data Browser:

- For a new widget, click **Select Data**, and then  to open the formula editor.
- For an existing widget, click on the edit formula button  to open the formula editor.

ECOMMERCE ×

Type to search for fields

Commerce

Age Range

Brand ID

Category ID

Condition

Cost

Country ID

Date

Gender

Quantity

Revenue

Visit ID

Brand

Brand ID

The Data Browser then changes to display the FORMULA EDITOR, which has two tabs: **Data Browser** and **Functions**.

- ▶ The **Data Browser** tab provides fields to choose from.
- ▶ The **Functions** tab lists the functions that you can include in your formula by selecting them. You can read a description of each function in a tooltip by hovering over it.

2. Define the formula as follows:

- ▶ From the **Data Browser** tab, select one or more fields.
- ▶ From the **Functions** tab, select the required functions.

SAMPLE HEALTHCARE > EDIT FORMULA ×

AVG [Cost Of Admission]

Data Browser
Functions

Type to search for fields

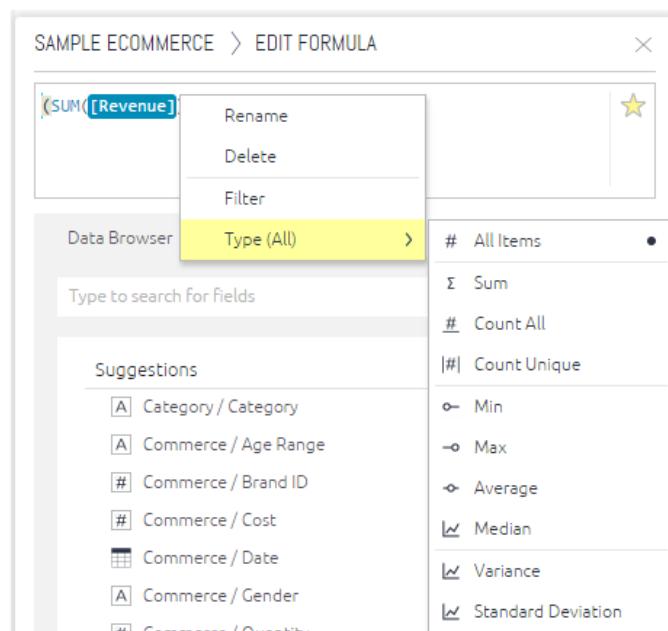
3. Type in the required parts of the formula. To see examples, see Formulas Based on Criteria and Conditions, and Functions to Build Formulas.
4. Click **OK**.

To edit a formula:

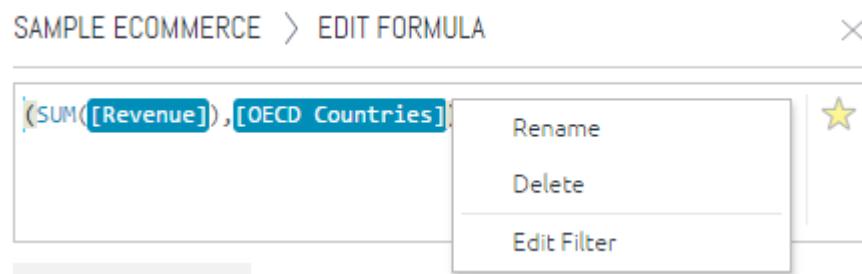
You can easily edit formulas using the right-click options. They include:

- ▶ **Rename:** Rename the formula, for example, give a name that represents a real-life task or expected result from the formula, or include in the name filters that you have added to the formula.
- ▶ **Filter:** Add filters to the formula.
- ▶ **Type:** Change the default aggregation method, for example, from *Sum* to *Average*.

The following image shows the right-click options.



The following image shows right-click options for a filtered formula.

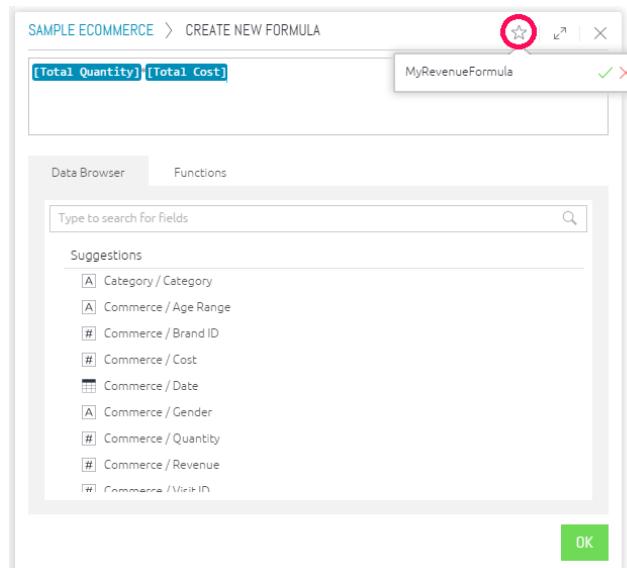


Reusing Formulas

If you added a new formula, and marked it as a Favorite (starred), then it can easily be reused. Formulas are saved per ElastiCube.

To mark a formula as a Favorite:

1. While defining a formula, click the Favorite (Star) button.
2. Enter a name for this Formula.
3. Click **OK**.



To reuse a favorite formula:

- ▶ Favorite Formulas appear in the Data Browser under the title **Formulas**, as shown below. Simply select it to use it.

ECOMMERCE ×

Type to search for fields

Suggestions

- [A] brand / Brand
- [A] category / Category
- [A] Commerce / Age Range
- [A] Commerce / Condition
- [#] Commerce / Cost
- [#] Commerce / Date
- [A] Commerce / Gender
- [#] Commerce / Quantity
- [#] Commerce / Revenue
- [#] Commerce / Visit ID

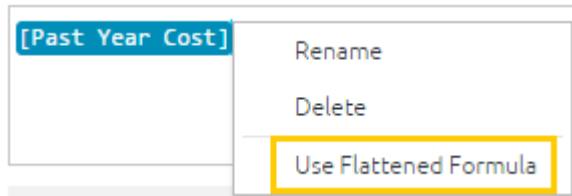
Formulas

MyRevenueFormula

Commerce

[A] Age Range

Note: To see the underlying building blocks of your favorite formula, right-click the favorite formula, and select **Use Flattened Formula**.



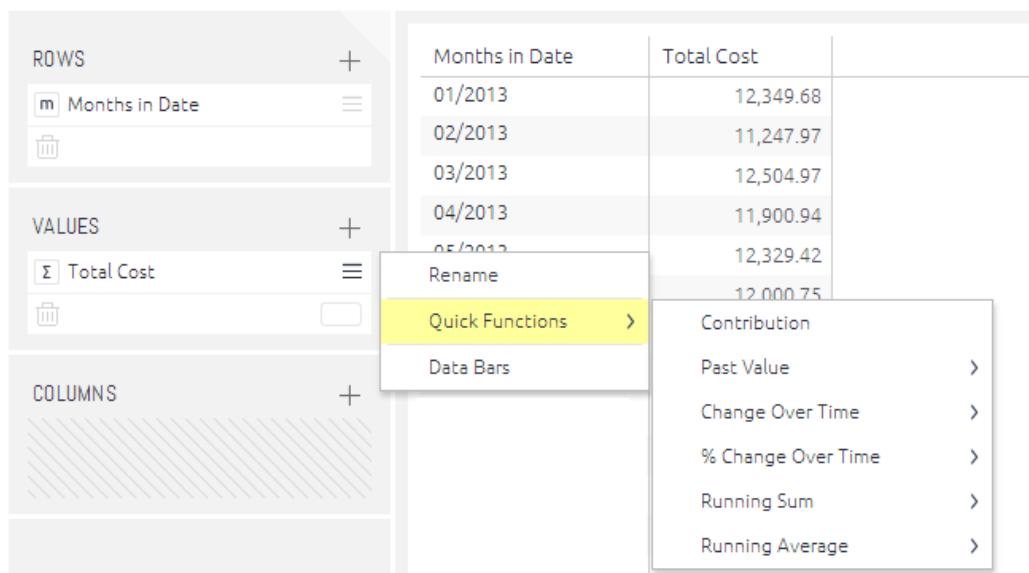
Using Quick Functions

Just like reusing formulas, quick functions is another feature to make working with formulas easier. The Widget Designer provides a variety of predefined commonly used functions that you can easily apply in the Data Browser when selecting a VALUE to be included in a Widget.

Quick Functions instantly add a time dimension to any existing value and formula. These functions include calculations for past values, change over time, contribution and running totals. Quick Functions

include all the Time Functions previously discussed but they can only be accessed by clicking on a formula that is already present in a widget.

A simple example of a Quick Function is a finance manager who is reviewing total costs per month but needs to track the accumulated annual costs. They can simply use a quick function to calculate the year to date total for costs.

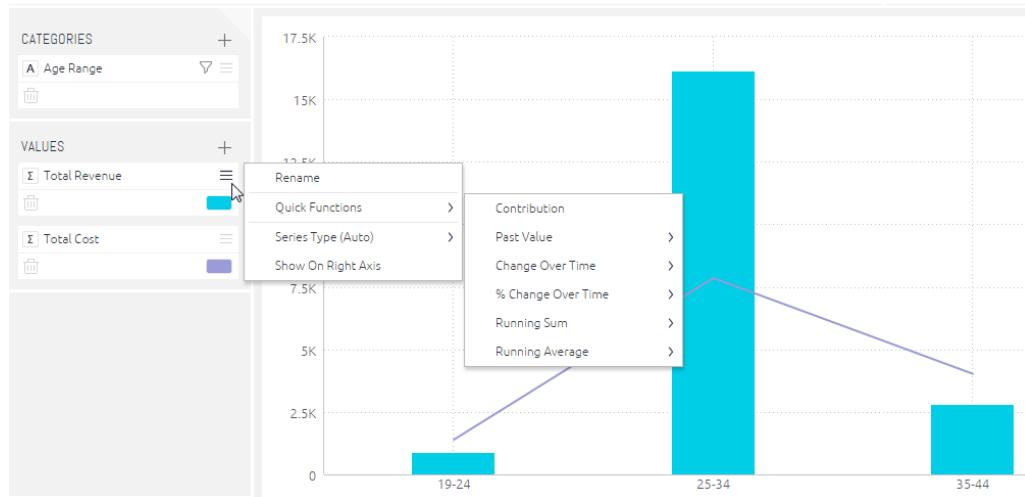


The screenshot shows the Sisense widget designer interface. On the left, there are three panels: ROWS, VALUES, and COLUMNS. The ROWS panel contains a single item: 'Months in Date'. The VALUES panel contains a single item: 'Total Cost'. The COLUMNS panel is empty. In the center, there is a data table with two columns: 'Months in Date' and 'Total Cost'. The data rows are: 01/2013 (12,349.68), 02/2013 (11,247.97), 03/2013 (12,504.97), 04/2013 (11,900.94), 05/2013 (12,329.42), and a total row (12,000.75). A context menu is open over the last row, showing options: Rename, Quick Functions (which is highlighted in yellow), Data Bars, Contribution, Past Value, Change Over Time, % Change Over Time, Running Sum, and Running Average.

Months in Date	Total Cost
01/2013	12,349.68
02/2013	11,247.97
03/2013	12,504.97
04/2013	11,900.94
05/2013	12,329.42
	12,000.75

To use a quick function:

1. Hover and click on the menu icon of a numeric field in the data panel of the widget designer, and select **Quick Functions** from the menu:



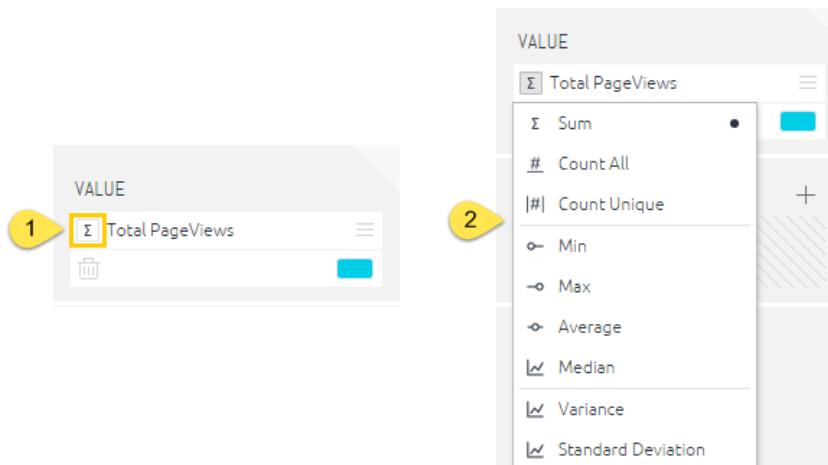
A list of commonly used functions is displayed.

2. Select a function. The widget will be updated immediately.

Quickly Adding Aggregate Functions

You can also add aggregate functions to your formula without opening the formula editor.

Click on the value icon to open a list of aggregate functions, and select the function to apply to your formula.



Starting Formulas with Quick functions

A more complex example uses starring with multiple quick functions. Let's say a finance manager also wants to compare year to date costs to the same period for the previous year. They can first apply the *year to date function* to total cost and then save it to the formula repository. They can then add the saved year to date formula but apply another quick formula for *past values* which will perform the same calculation but on data from the previous year.

Note: Starred (favorite) formulas will be shared with other users.

Using R in Formulas

R is a software environment for statistical computing and graphics. Sisense supports the integration of R functions in your formulas.

You can write R code directly in the formula editor, and send fields as parameters. You can also combine R functionality with the regular functions to create advanced and tailored formulas.

Before you can integrate R into your formulas, you must have an R server set up in your organization.

This document will cover the following:

- ▶ Understanding How R works in Sisense
- ▶ Connecting Sisense to Your R Server
- ▶ Using R functions in Formulas
- ▶ Guidelines for using R code in Sisense
- ▶ Example 1: Simple R Function –Kmeans Clustering
- ▶ Example 2: R Logistic Regression via Sisense
- ▶ Example 3: Loading an Existing R Model
- ▶ Example 4: Building and Saving a R Model using Data from Sisense

Understanding How R works in Sisense

Within a widget, R code is used and assigned fields as parameters, the following happens:

1. The selected fields are sent fully to R as a list of arrays.
2. The R code is executed on the R server, referencing the fields as needed.
3. The numeric results from R are sent back to the widget for visualization.

Each request from the R server is done in a self-contained namespace, thus R variables and results cannot be reused across requests unless they are loaded from a saved location.

R calculations are computed in the R server, and not in the Sisense ElatsiCube, this means that for larger data sets results will not return as fast as regular widgets. Sisense recommends Revolution Analytics ([Revolution R Open](#)) for enhanced R performance.

Running R on Windows

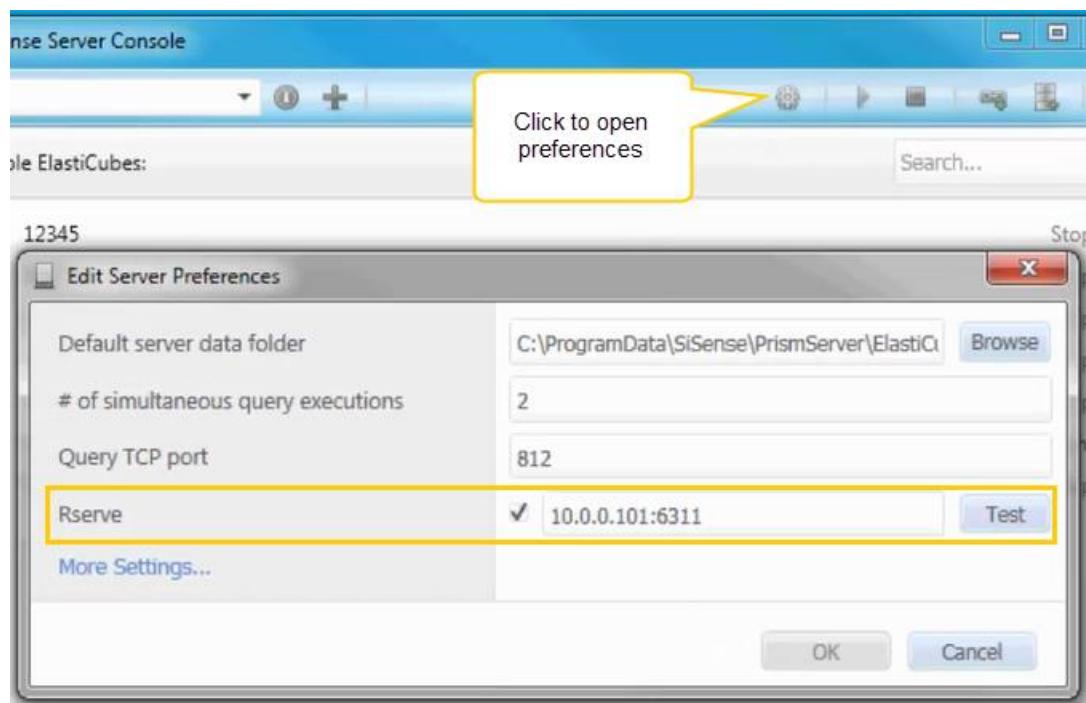
If your R server is running on Windows you will be bound to the concurrency limitations the R distribution has on Windows. This means that concurrent requests to R may return wrong results. Sisense recommends using R on a Linux machine for production environment with concurrent user activity.

Connecting Sisense to Your R Server

To connect your Sisense instance to your R server:

1. Open the Sisense Server Console from the start menu.
2. Click on the Server Preferences icon to open the server settings.

3. In the **Rserve** field, enter the IP address of your R server, and select the checkbox to enable the connection.



You can test the connection by clicking on the **Test** button.

Installing Rserve

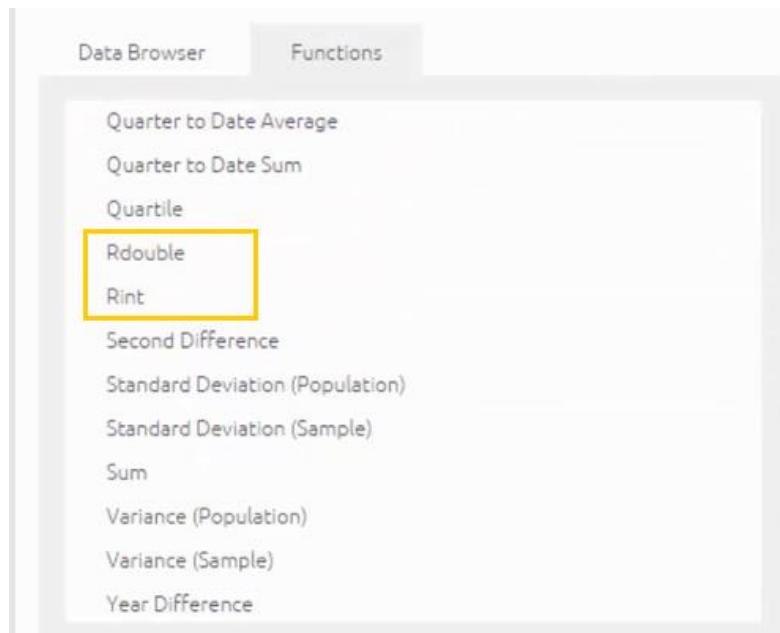
Sisense can only connect to a running Rserve instance. For more information about installing and running Rserve [follow this link](#).

Please note that if you're using RStudio, Rserve still has to be installed.

Using R functions in Formulas

Using R code in Sisense is achieved by using two dedicated formulas – RINT & RDOUBLE. Depending on the result type of your R code, use the appropriate function within the [formula editor](#).

For either formula, RINT or RDOUBLE, the return type has to be an exact match.



Multiple numeric field values can be passed as parameters to the R functions, which within your R code can be accessed via the 'args' argument (See K-means example below).

Syntax:

```
RINT([recycle (true)], [<Ordering>], <R expression>,
[<numeric value 1>, ..., <numeric value n>] )
RDOUBLE([recycle (true)], [<Ordering>], <R
expression>, [<numeric value 1>, ..., <numeric value n>] )
Sisense does not support the use of a final ';' in R statements. If you add a
semi-colon at the end of your R statement, the code will fail.
```

Parameters:

- ▶ **Recycle true/false (default = true):** This is an optional parameter that controls whether the results from R should be recycled (cached), so that consequent queries will not have to be recalculated unless they or the data have changed. Generally, this behavior is automatically managed by the ElastiCube automatically for all functions. However, since R code might have non-deterministic components to it (such as

randomality functions or date-specific functions), the ElastiCube cannot rely on a data-set and function that hasn't changed not to return a different result in multiple executions.

By default, the Recycle value is set to true. Use 'off' if your R code contains randomality or other non-deterministic content.

- ▶ **Ordering:** This is an optional parameter that defines the sort order in which numeric data is sent to R. The argument of the Ordering parameter can be an index in your data source or you can use the ORDERING() function to determine the order of your fields. This function arranges the values of the arguments into ascending or descending order, breaking ties by further arguments.

For example:

```
ORDERING([Total Sales], -1*[COUNT Salesman],  
MIN(<Office Name>))
```

For more information about the ORDERING() function, click [here](#).

- ▶ **R expression:** Your R code needs to be passed here, wrapped in double quotes. R expects the return type to be an array with the same size as widget's row count. Nulls will be used to make up for shorter arrays, and longer arrays will be trimmed
- ▶ Use **single quotes** to wrap strings within your R code when using the Rint/Rdouble functions, so that there will be no double-quote collision with the quotes wrapping your R code within the Rint/Rdouble function.
- ▶ **Numeric Value Arguments:** Numeric values can be passed as arguments to your R code.

All arguments are passed to R as a 1-based list named "args".

Each item in the list contains an array that represents the field.

For example:

- ▶ `args[[1]]` will return an array which represents the first field that was used as an argument.
- ▶ `args[[2]][3]` will return the 3rd data value within the 2nd field that was used as an argument.

Guidelines for using R code in Sisense

In order for your R code to work well in Sisense you need to make sure of the following:

1. The resultset returned from R into Sisense has to be equal in size to the dataset that is sent to R as parameters.
2. The order of the resultset return from R into Sisense has to match the incoming order. No assumptions can be made about the incoming order –This means that you cannot assume that the incoming order matches the order you visually see in the widget.

Example 1: Simple R Function – Kmeans Clustering

Description: In the example below the Total Cost and Total Revenue fields from Sisense are being used to cluster data via a Kmeans function. The result will cluster the data based on the kmeans settings, in this case 4 clusters.

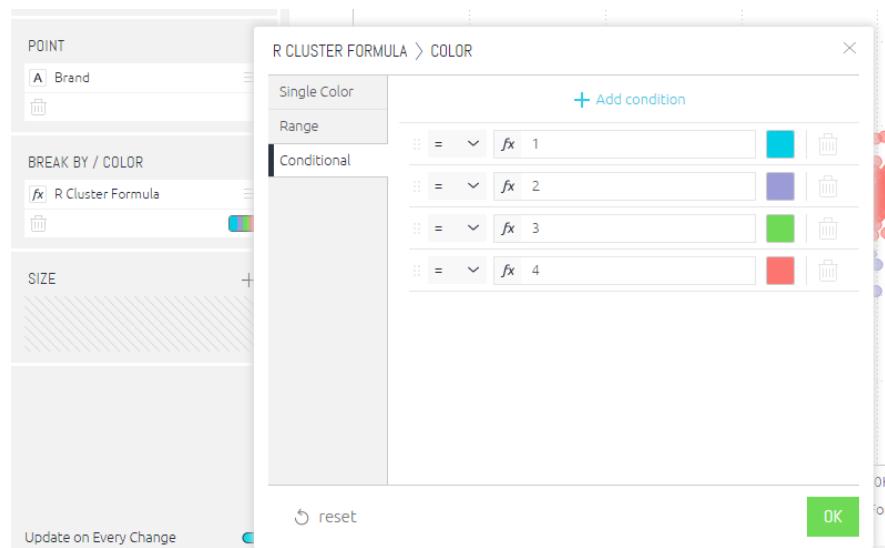
Sisense Syntax:

```
RINT(TRUE, "m<- log(matrix(unlist(args), ncol = 2));  
kmeans(m, 4) $cluster" ,[Total Cost],[Total Revenue])
```

Result: The R expression will return a result from 1 to 4. The widgets color settings can be configured to color corresponding values based on if they equal 1, 2, 3 or 4.

Tip: You can [save \(star\)](#) the complex formula above, and use it again in additional formulas and charts.

In the chart's BREAK BY/COLOR panel, you can change the color of the clusters as well as define the number of clusters and their break points.



Example 2: R Logistic Regression via Sisense

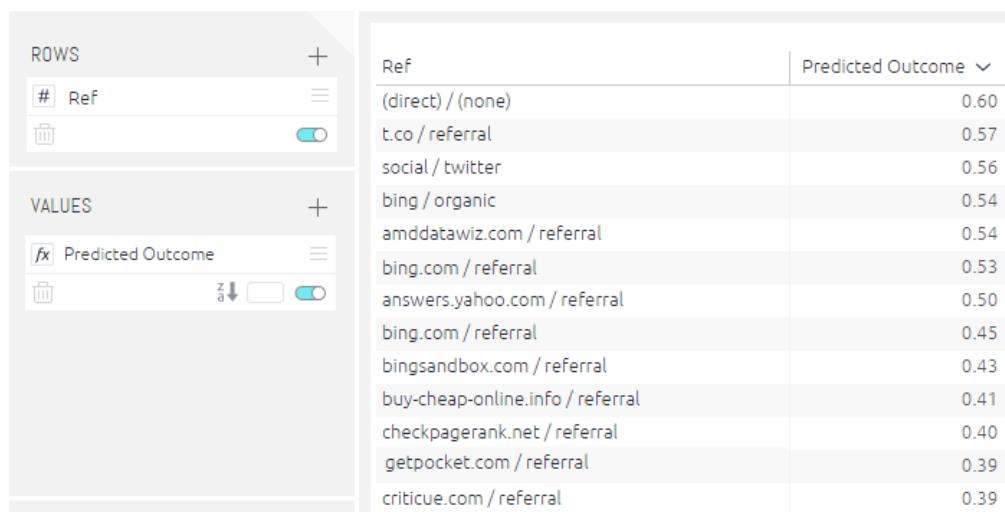
Description: In the example below website traffic data is used to predict if a conversion is likely to occur based on the source of web traffic. The following fields contained in the ElastiCube are passed to the R logistic model: Average Outcome (1=conversion, 0= no conversion), Average Time on Page, Average Pages (viewed) and Average Bounce Rate. This data is passed to a model to predict the outcome and likelihood the traffic will convert.

[For the sake of illustration both the training and test data set are the same in the example]

Sisense Syntax:

```
RDOUBLE (TRUE, "mydata<-data.frame(convert=args[[1]],
time=args[[2]], pages=args[[3]], bounce=args[[4]]);lrmmodel <-
glm(convert ~ time + pages + bounce, data = mydata, family =
'binomial');prob <- predict(lrmmodel, newdata = mydata, type =
'response')", [Average Outcome], [Average TimeonPage],
[Average Pages], [Average BounceRate])
```

Result: The R expression will return a result from 0 to 1 showing the likelihood of a conversion occurring. Note a similar process can be applied to create a linear regression.



The screenshot shows the Sisense interface with a table view. On the left, there are two sections: 'ROWS' containing a 'Ref' field and 'VALUES' containing a 'Predicted Outcome' field. To the right is a table with columns 'Ref' and 'Predicted Outcome'.

Ref	Predicted Outcome
(direct) / (none)	0.60
t.co / referral	0.57
social / twitter	0.56
bing / organic	0.54
amddatawiz.com / referral	0.54
bing.com / referral	0.53
answers.yahoo.com / referral	0.50
bing.com / referral	0.45
bingsandbox.com / referral	0.43
buy-cheap-online.info / referral	0.41
checkpagerank.net / referral	0.40
getpocket.com / referral	0.39
critique.com / referral	0.39

Predicted outcome of website conversion based on logistic regression.

Example 3: Loading an Existing R Model

Description: In the example below a saved logistic regression built on previous training data is called and used with new website traffic data to predict if a conversion occurs. The following fields contained in the ElastiCube are passed to the saved R logistic model: Average Outcome (1=conversion, 0= no conversion), Average Time on Page, Average Pages (viewed) and Average Bounce Rate. This data is passed to a model to predict the outcome and likelihood the traffic will convert.

Sisense Syntax:

```
RDOUBLE (TRUE, "mydata<-data.frame(convert=args[[1]],  
time=args[[2]], pages=args[[3]],  
bounce=args[[4]]);load('C:\rdata');prob <- predict(lrmodel,  
newdata = mydata, type = 'response')", [Average Outcome],  
[Average TimeonPage], [Average Pages], [Average BounceRate])
```

Result: The R expression will return a result from 0 to 1 showing the likelihood of a conversion occurring based on the loaded R model in this case stored in 'rdata'.

Example 4: Building and Saving a R Model using Data from Sisense

Description: In the example below a logistic regression to predict if a website conversion will occur is built using data from Sisense and then saved as an R model. The following fields contained in the ElastiCube are passed to create the R logistic model: Average Outcome (1=conversion, 0= no conversion), Average Time on Page, Average Pages (viewed) and Average Bounce Rate. This data is saved

as an rdata file and can be used on other data sets to predict outcomes.

Sisense Syntax:

```
RDOUBLE (TRUE, "mydata<-data.frame(convert=args[[1]],  
time=args[[2]], pages=args[[3]], bounce=args[[4]]);lrmodel <-  
glm(convert ~ time + pages + bounce, data = mydata, family =  
'binomial'); save(lrmodel, file = 'C:\rdata');1", [Average  
Outcome], [Average TimeonPage], [Average Pages], [Average  
BounceRate])
```

Result: The R expression will use Sisense data to build and save a logistic model in the specified file in this case 'rdata'.

Creating Formulas Based on Criteria and Conditions (Filters)

Often formulas must take into account specific criteria. To do this Sisense provides a feature called **Measured Value**, which like the SUMIF function in Excel, only performs a calculation when the values meet a set of criteria.

Criteria for Measured Values may be based on any logical operators in a filter.

Measure Value Syntax:
(Measure, Scope1 , Scope2...ScopeN)

Parameters

Measure: A field measure or formula.

Scope: A filter including *Value, Text, List, Ranking and Time filters*.

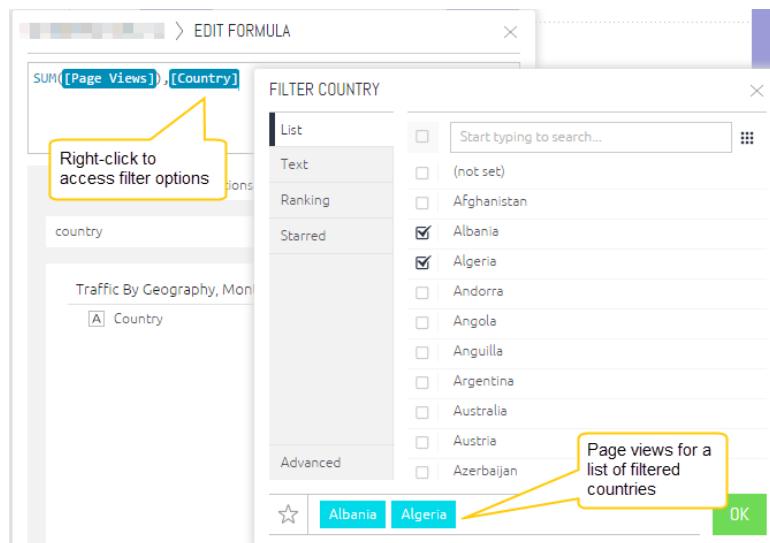
To filter the formula:

1. In the Data Browser, create your formula from the Data Browser and Functions, as explained in [Using the Formula Editor](#).

2. Add the field (criteria) by which you want to filter the formula.

Right-click the field and select **Filter**.

3. You can then filter the formula by listed items, text options, ranking, etc. When done, click **OK**.



A simple example of Measured Value is the **use of a list filter**.

A marketing team may need to count leads generated for a specific region such as North America. Even if leads come from many different countries, the measured value calculates leads generated only when the lead originates from the United States or Canada.

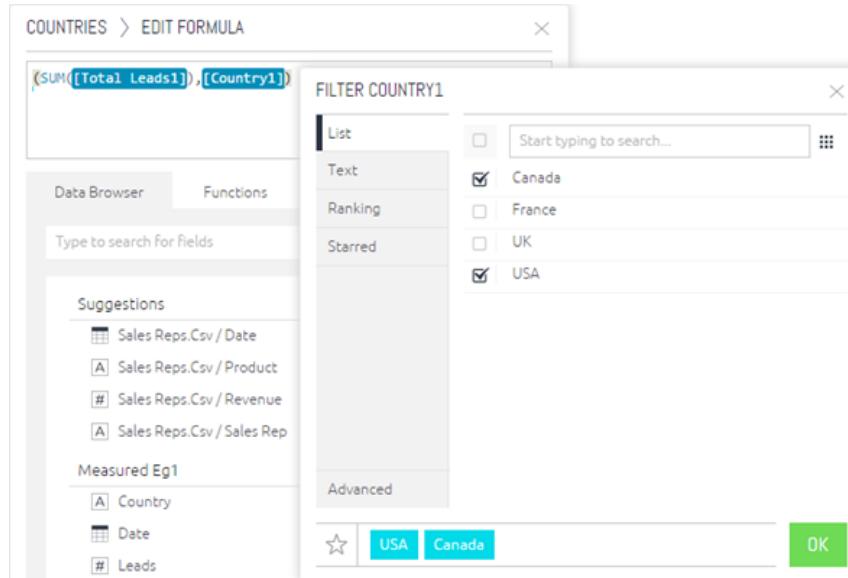
Country	# Leads
USA	15
France	20
Canada	5
UK	30

*Example 1: Measured Value with List Filter
 $(\text{Sum}(\text{Leads}), (\text{List Filter: Country} = \text{United States, Canada}))$*

Result

$15+5 = 20$

The above example as defined in the formula editor.



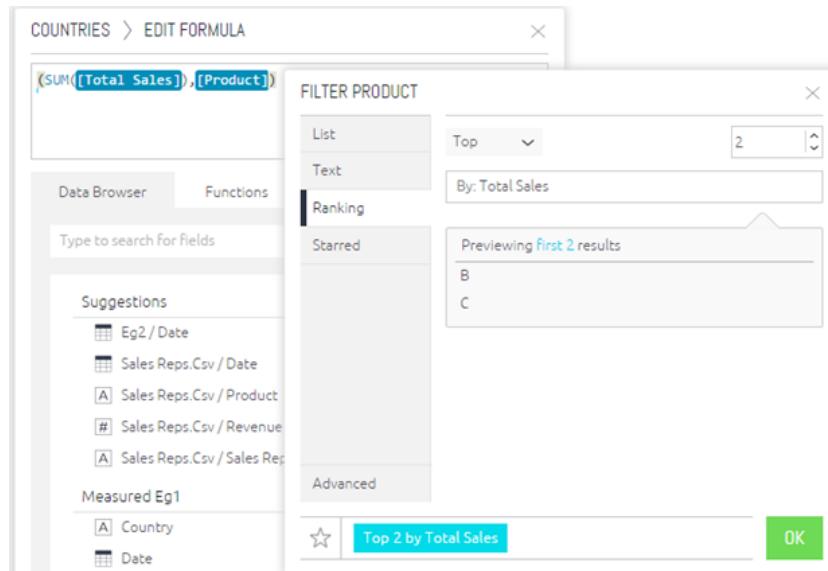
A more sophisticated case is the **use of a ranking filter**, for example a sales team may want to track the contribution of best-selling products to total revenue. However, what constitutes a popular product may change over time. A measured value can be created for sales which includes a condition that only shows sales for the top products for any month. This simultaneously filters the data but also takes into account changes in what classifies as a top product over time.

Month	Product	Sales
Jan	A	10
Jan	B	5
Jan	C	15
Feb	A	1
Feb	B	25
Feb	C	5

*Example 2: Measured Value with Top Ranking Filter
(Sum(Sales), (Top Ranking Filter: Top 2 Products by Sales))*

Results	
Month	Total Sales Top 2 Products
Jan	(A/C) 10 + 15 = 25
Feb	(B/C) 25 + 5 = 30

The above example as defined in the formula editor.



Measured Values are a powerful feature to take into account business logic and quickly perform calculations only when a specific set of criteria is met.

Note about duplicate filtering: If your widget is filtered using measured values, then the measured value will override any other widget or dashboard filters you have for the same fields.

Calculating Contributions Using the ALL Function

The All() function returns the total amount for a dimension, and can be used for various use cases. In the following example, we will use the All function to calculate how much each country contributed towards the total cost of a campaign.

Our final widget includes the following information:

Country	Total Cost	Total Cost per Countries	Contribution
Australia	9,643.09	283,755.53	3.40%
Brazil	9,638.80	283,755.53	3.40%
China	9,607.4	283,755.53	3.39%
England	19,492.8	283,755.53	6.87%
France	19,440.59	283,755.53	6.85%
Germany	19,377.61	283,755.53	6.83%
Greece	9,842.84	283,755.53	3.47%
India	9,603.16	283,755.53	3.38%
Kazakhstan	9,899.33	283,755.53	3.49%
Nigeria	9,764.44	283,755.53	3.44%
Norway	9,681.69	283,755.53	3.41%
Portugal	9,919.33	283,755.53	3.50%
South Africa	19,211.15	283,755.53	6.77%
Ukraine	9,693.03	283,755.53	3.42%
United States	99,241.60	283,755.53	34.97%
Vietnam	9,698.67	283,755.53	3.42%
Grand Total	283,755.53	283,755.53	100%

Step 1: The second column above represents a formula that sums up the total cost for all countries and does not represent the breakdown per country. The formula includes the calculation (total cost) followed by the `all` function (filter), followed by the dimension (country) in parenthesis. It looks like this:

```
([Total Cost], all([Country]))
```

We can save (star) the above formula and call it Total cost for Countries, which will be used in the next step.

Step 2: We can now use the above formula in another formula to calculate the contribution, like this:

```
SUM([Cost]) / [Total Cost for Countries]
```

The result is the third column above (plus formatting the results as percentages).

Building Formulas with Functions

Functions are operations that perform common types of calculations, and can be used to build formulas. In this section you can read about

four types of functions. The functions' syntax is explained and examples are provided.

Combine Data: Aggregate Functions

Aggregations are used to perform mathematical calculations on data.

Although this is an essential function Sisense offers an advantage in the ability to run multiple aggregations on several fields simultaneously –this makes it easy to summarize data based on multiple factors.

Aggregate Syntax:
Function(Numeric Field)

Parameters
Function: Sum(), Avg(), Count(), DupCount(), Max(), Min()
Numeric Field: A numeric field or formula.

An example **simple aggregation** is a sales manager who wants to calculate the average sales revenue for each sales rep. They can create a pivot table which shows the sales rep and their average sales revenue.

A more complex example is a **multi-pass aggregation** (or grouping) which is an aggregation that performs multiple calculations simultaneously. Following our first example let's assume the sales manager wants to also see *average sales per day* for each sales rep. Instead of having to add an additional column for day in the pivot table the manager can create a multi-pass aggregation that first performs a sum of sales per day and then averages the results for each rep. This requires two fields –a day from a date field and the revenue field, as well as two aggregations, sum of sales and average. This result is the sales manager does not need to add a column for days in the pivot.

Data Set

Day	Sales Rep	Revenue
1 Jan	John	10
1 Jan	Jane	20
1 Jan	Jane	40
2 Feb	John	30
2 Feb	John	10
2 Feb	Jane	5

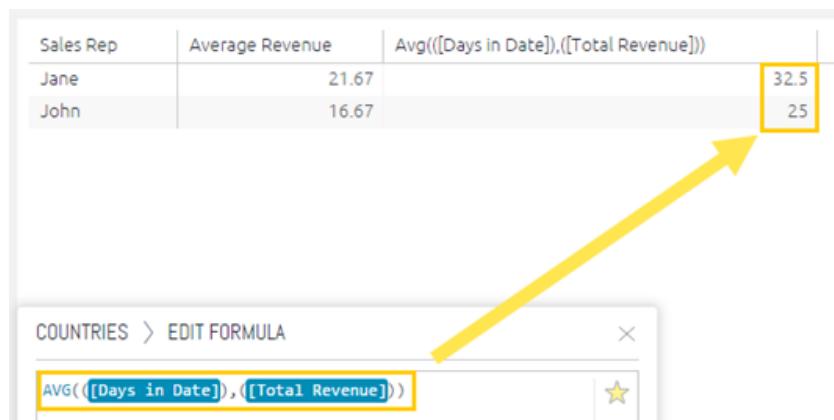
Results Example 1: Avg(Revenue)

Sales Rep	Avg Revenue
John	$(10+30+10)/3 = 16.67$
Jane	$(20+40+5)/3 = 21.67$

Results Example 2: Avg(Days, Sum(Revenue))

Sales Rep	Avg Daily Revenue
John	$(10+(30+10))/2 = 25$
Jane	$((20+40)+5) / 2 = 32.5$

The above example as shown in the formula editor.



The screenshot shows a table with two rows:

Sales Rep	Average Revenue
Jane	21.67
John	16.67

An arrow points from the value '32.5' in the bottom right corner of the table to the formula bar below. The formula bar displays the formula: `AVG([Days in Date], [Total Revenue])`.

Summarize Data: Statistical Functions

Descriptive statistics provide meaningful summaries of data and help make more informed decisions. This is particularly important for large data sets where descriptive statistics can help to focus analysis.

Statistical Syntax: Function(Numeric Field)

Parameters

Function: Median(), Mode(), Largest(), Stdev(), Stdevp(), Varp(), Var(), Quartile(), Percentile()
Numeric Field: A numeric field or formula.

An example of statistical functions is a marketing team that has a large data set on leads generated from various channels and want to understand where to focus their budget. Descriptive statistics can be used to summarize valuable insight about each channel such as the

central tendency or median leads generated along with standard deviations to assess typical lead volume.

Statistical Syntax: Function(Numeric Field)

Parameters

Function: Median(), Mode(), Largest(), Stdev(), Stdevp(), Varp(), Var(), Quartile(), Percentile()

Numeric Field: A numeric field or formula.

Accumulate Data: Running Total and Average

Often to measure performance data must be viewed in a continuous and accumulative format over extended periods such as years, quarters or months. Sisense provides functions to create running totals and averages over standard or custom time periods.

Accumulate Syntax: Function(Numeric Field)

Parameters

Function: YTDAvg(), QTDAvg(), MTDAvg(), YTDSum(), QTDSum(), MTDSum(), RPSum() / RP Avg()

Numeric Field: A numeric field or formula.

For example, a support team has a goal to reduce the average monthly cost to resolve open issues. A *Year to Date Average* can be used to track progress towards reducing the average cost of support.

Data

Month	Cost
Jan	30
Feb	35
Mar	40
Apr	20
May	25
Jun	10

Results Example 1: YTDAvg(Cost)

Month	YTD Avg Cost
Jan	30
Feb	32.5
Mar	35
Apr	31.25
May	30
Jun	26.67

Change over Time: Time Functions

Time is critical for business decisions, Sisense time functions make it easy to compare outcomes at different points in time, determine

growth rates and calculate the time differences. Functions can be set for common time periods such as year, month or day as well as for custom periods.

*Time Syntax: Function(Numeric Field) **

Parameters

Function:

PastYear(), PastQuarter(), PastMonth(), Growth(), GrowthRate(), Contribution(), YDiff(), QDiff(), MDiff(), DDiff(), HDiff(), MnDiff(), SDiff(), Next(), Prev(), DiffPastYear(), DiffPastMonth(), DiffPastQuarter(), DiffPastDay()

Numeric Field: A numeric field or formula.

** To calculate time functions require a time filter applied on the dashboard or widget.*

For example, an executive team wants to compare growth in revenue to the same period in the previous year. A *Difference in Past Year* function can be used to compare past values based on the current month to the same month in the previous year.

Data

Date	Revenue
01-2013	5
02-2013	10
03-2013	15
01-2014	25
02-2014	5
03-2014	10

Results Example 1: DiffPastYear(Revenue)

Month	Difference to Past Year
01-2013	
02-2013	
03-2013	
01-2014	25-5 =20
02-2014	5-10= -5
03-2014	10-15 =-5

Function Reference

This page contains a list of all the functions you can use in Sisense's formula editor.

Statistical Functions

Average

`Avg(<numeric Field>)`

Calculates the mean average of the given values.

For example –`AVG(Score)` will calculate the mean average of the given scores.

`Avg(<group by field>, <aggregation>)`

Calculates the average of the given aggregation grouped by another filed.

For example –`Avg(Product, Total Sales)` will calculates the average of the total sales per product.

Contribution

`Contribution(<numeric field>)`

Calculates the percentage of total.

For example –`Contribution(Total Sales)` will calculate the percentage of total sales per group (for example per day or per product) out of total sales (for all days or all products).

Correlation

`CORREL(<Numeric Field a>, <Numeric Field b>)`

Returns the correlation coefficient of two numeric fields.

For example –`CORREL(Revenue, Cost)` will return the correlation between revenue and cost.

`CORREL(<group by field>, <aggregation a>, <aggregation b>)`

Returns the correlation coefficient of two fields aggregations grouped by another field.

For example –`CORREL(Products, AVG(Revenue), AVG(Cost))`

will return the correlation between the average of revenue and cost per product.

Count

`Count (<Numeric Field>)`

Counts the number of unique values withing the given values.

Count All

`DupCount (<Numeric Field>)`

Returns the actual item count of the given list of items, including duplicates.

Covariance (Population)

`COVARP (<Numeric Field a>, <Numeric Field b>)`

Returns the population covariance of `<Numeric Field a>` and `<Numeric Field b>`.

For example –`COVARP(Revenue, Cost)` will return the population covariance of revenue and cost.

`COVARP(<group by field>, <aggregation a>, <aggregation b>)`

Returns the population covariance of two fields aggregations grouped by another field.

For example –`COVARP(Products, AVG(Revenue), AVG(Cost))` will return the population covariance of the average revenue and the average cost per product.

Covariance (Sample)

`COVAR(<Numeric Field a>, <Numeric Field b>)`

Returns the sample covariance of `<Numeric Field a>` and `<Numeric Field b>`.

For example –`COVAR(Revenue, Cost)` will return the sample covariance of revenue and cost.

`COVAR(<group by field>, <aggregation a>, <aggregation b>)`

Returns the sample covariance of two fields aggregations grouped by another field.

For example –`COVAR(Products, AVG(Revenue), AVG(Cost))` will return the sample covariance of the average revenue and the average cost per product.

Exponential Distribution

`EXPONDIST(<numeric value>, <lambd>, <Cumulative (true/false)>)`

Returns the exponential distribution for a given value and a supplied distribution parameter lambda. Cumulative: TRUE = Cumulative distribution function, FALSE = Probability density function. For example –`EXPONDIST(Count(Leads), 2, False)` will return the exponential distribution density of the number of leads per country where lambda is 2.

Intercept

`INTERCEPT(<field>, <numeric value>)`

Returns the intercept of the linear regression line through a supplied series of x- and y- values.

For example –`INTERCEPT(Date.Quarter, Total Sales)` will return the

intercept of the regression line that represents the trend over quarter of the sum of sales.

Largest

```
LARGEST(<Numeric Field>, <k>)
```

Returns the k-th largest value in a field.

Maximum

```
Max(<Numeric Field>)
```

Returns the maximum value among the given values.

Median

```
MEDIAN( <Numeric Field> )
```

Calculates the median of the given values. The median of a set of data is the middlemost number in the set. The median is also the number that is halfway into the set.

Minimum

```
Min(<Numeric Field>)
```

Returns the minimum value among the given values.

Mode

```
MODE(<Numeric Field>)
```

Returns the most frequently occurring value from the column.

Normal Distribution

```
NORMDIST(<Numeric Field>, <Mean>, <Standard Deviation>, <Cumulative (true/false)>)
```

Returns the standard normal distribution for a given value, a supplied distribution mean and standard deviation. Cumulative: TRUE = Cumulative Normal Distribution Function, FALSE = Normal Probability Density Function.

For example –NORMDIST(Score, (Mean(Score), All(Score)), (STDEV(Score), All(Score)), False) will return the normal probability density of a given score.

Percentile

`PERCENTILE (<Numeric Field>, <k>)`

Returns the k-th percentile value from the given field.
k is any number between 0..1 (inclusive).

Poisson Distribution

`POISSONDIST(<numeric value>, <mean>, <Cumulative (true/false)>)`

Returns the poisson distribution for a given value and a supplied distribution mean. Cumulative: TRUE = Cumulative distribution function, FALSE = Probability mass function.

For example –POISSONDIST(Score, (Mean(Score), All(Score)), (STDEV(Score), All(Score)), False) will return the poisson probability density of a given number of sales

Quartile

`QUARTILE (<Numeric Field>, <k>)`

Returns the k-th quartile for the given field.

- ▶ k = 0 returns the Minimum value
- ▶ k = 1 returns the first quartile (25th percentile)
- ▶ k = 2 returns the Median value (50th percentile)

- ▶ k = 3 returns the third quartile (75th percentile)
- ▶ k = 4 returns the Maximum value

Rank

`RANK(<numeric value>, [DESC/ASC], [Rank Type], [<group by field 1>,... , <group by field n>])`

Returns the rank of a value in a list of values.

[DESC/ASC] –Optional. By default sort order is descending.

[Rank Type] –Optional. By default the type is standard competition ranking ("1224" ranking). Support also modified competition ranking ("1334" ranking), dense ranking ("1223" ranking) and ordinal ranking ("1234" ranking).

[<Group by field 1>,...,<Group by field n>] –Optional. Rank partitions fields.

For example –`RANK(Total Cost, "ASC", "1224", Product, Years)` will return the rank of the total annual cost per each product were sorted in ascending order.

Skewness (Population)

`SKEWP(<numeric value>)`

Returns the skewness of the distribution of a given value in the population.

For example –`SKEWP(Revenue)` will return the skewness of the distribution of revenue in the population.

Skewness (Sample)

`SKEW(<numeric value>)`

Returns the skewness of the distribution of a given value.

For example –`SKEW(Revenue)` will return the skewness of the distribution of revenue.

Slope

`SLOPE(<field>, <numeric value>)`

Returns the slope of the linear regression line through a supplied series of x- and y- values.

For example –`SLOPE(Date.Quarter, Total Sales)` will return the slope of the regression line that represent the trend over quarter of the sum of sales.

Standard Deviation (Population)

`STDEVP(<Numeric Value>)`

Returns the Standard Deviation of the given values

(Population). Standard deviation is the square root of the average squared deviation from the mean. The standard deviation of a population gives researchers the amount of dispersion of data for an entire population of survey respondents.

Standard Deviation (Sample)

`STDEV(<Numeric Value>)`

Returns the Standard Deviation of the given values

(Sample). Standard deviation is the square root of the average squared deviation from the mean. A standard deviation of a sample estimates the amount of dispersion in a given data set, based on a random sample.

T Distribution

`TDIST(<numeric value x>, <degrees_freedom>, <Cumulative (true/false)>)`

Returns the student's T-distribution for a given value and a supplied number of degrees of freedom (must be ≥ 1). Cumulative: TRUE = Cumulative Distribution Function, FALSE = Probability Density Function.

For example –`TDIST(Score, 3, TRUE)` will return the student's T-distribution of a given score, with 3 degrees of freedom.

Variance (Population)

`VARP(<Numeric Value>)`

Returns the Variance of the given values (Population). Variance (Sample) is the average squared deviation from the mean, based on an entire population of survey respondents.

Variance (Sample)

`VAR(<Numeric Value>)`

Returns the Variance of the given values (Sample). Variance (Sample) is the average squared deviation from the mean, based on a random sample of the population.

Mathematical Functions

Absolute

`Abs(<Numeric value>)`

Returns the absolute value of the given value.

For example –`ABS(Cost)`, where the absolute result for the value '2' or '-2' is '2'.

Acos

`ACOS (<numeric value>)`

Returns the angle, in radians, whose cosine is the given numeric expression. Also referred to as arccosine.

For example –`ACOS(Total Revenue)` will return the angle, in radians, whose cosine is the given total revenue.

Asin

`ASIN (<numeric value>)`

Returns the angle, in radians, whose sine is the given numeric expression. Also referred to as arcsine.

For example –`ASIN(Total Revenue)` will return the angle, in radians, whose sine is the given total revenue.

Atan

`ATAN (<numeric value>)`

Returns the angle in radians whose tangent is the given numeric expression. Also referred to as arctangent.

For example –`ATAN(Total Revenue)` will return the angle in radians whose tangent is the given total revenue.

Ceiling

`CEILING (<numeric value>)`

Returns number rounded up, away from zero, to the nearest multiple of significance.

For example –`CEILING(Cost)`, where the result of ‘83.2’ rounded up is ‘84’.

Cos

`COS (<numeric value>)`

Returns the trigonometric cosine of the given angle (in radians).

For example –`COS(Average Angle)` will return the trigonometric cosine of the average angle.

Cosh

`COSH (<numeric value>)`

Returns the hyperbolic cosine of the given value.

For example –`COSH(Total Revenue)` will return the hyperbolic cosine of the total revenue.

Cot

`COT (<numeric value>)`

Returns the trigonometric cotangent of the given angle (in radians).

For example –`COT(Average Angle)` will return the trigonometric cotangent of the average angle.

Exp

`EXP (<numeric value>)`

Returns the exponential value of the given value.

For example –`EXP(Sales)` will return the exponential value of sales.

Floor

`FLOOR (<numeric value>)`

Returns number rounded down, toward zero, to the nearest multiple of '1'.

For example –FLOOR(Revenue), where the result of '88.6' rounded down is '88'.

Ln

`LN (<numeric value>)`

Returns the base-e logarithm of the given value.

For example –LN(Cost) will return the base-e logarithm of cost.

Log10

`LOG10 (<numeric value>)`

Returns the base-10 logarithm of the given value.

For example –LOG10(Revenue) will return the base-10 logarithm of revenue.

Mod

`MOD (<numeric value>, divisor)`

Returns the remainder after a number is divided by a divisor.

For example –MOD(Cost, 10), where the remainder of '255' divided by '10' is '5'.

Power

`Power (value, power)`

Returns the results of the given value raised to a supplied power.

For example –POWER(Revenue, 2) will return revenue raised by the power of 2.

Quotient

`QUOTIENT (<numeric value>, divisor)`

Returns the integer portion of a division.

For example –QUOTIENT(Cost, 2), where the integer portion of '5' divided by '2' is '2'.

Round

`ROUND (<numeric value>, num_digits)`

Returns number rounded to a specified number of digits.

For example –ROUND(Revenue, 2) will return the revenue rounded to two decimal places.

Sin

`SIN (<numeric value>)`

Returns the trigonometric sine of the given angle (in radians).

For example –SIN(Average Angle) will return the trigonometric sine of the average angle.

Sinh

`SINH (<numeric value>)`

Returns the hyperbolic sine of the given value.

For example –SINH(Total Revenue) will return the hyperbolic sine of the total revenue.

Square root

`SQRT (<Numeric value>)`

Returns the square root of the given value.

For example –SQRT(Cost) will return the square root of cost.

Sum

`Sum (<Numeric Field>)`

Calculates the total of the given values.

Tan

`TAN(<numeric value>)`

Returns the trigonometric tangent of the given angle (in radians).

For example –`TAN(Average Angle)` will return the trigonometric tangent of the average angle.

Tanh

`TANH(<numeric value>)`

Returns the hyperbolic tangent of the given value.

For example –`TANH(Total Revenue)` will return the hyperbolic tangent of the total revenue.

Time Related Functions

Day Difference

`DDiff(<Start Time>, <End Time>)`

Returns the difference between `<Start Time>` and `<End Time>` in days.

Growth

`Growth(<Numeric Value>)`

Calculates growth over time. The time dimension to be used is determined by the time resolution in the widget/dashboard.

Formula: $(\text{current value} - \text{compared value}) / \text{compared value}$.

For example:

- ▶ If this month your value is 12, and last month it was 10, your Growth for this month is 20% (0.2).

Calculation: $(12 - 10) / 10 = 0.2$

- ▶ If this year your value is 80, and last year it was 100, your Growth for this year is -20% (-0.2).

Calculation: $(80 - 100) / 100 = -0.2$

Growth Rate

`GrowthRate(<Numeric Value>)`

Calculates growth rate over time. The time dimension to be used is determined by the time resolution in the widget/dashboard.

For example:

- ▶ If this month your value is 12, and last month it was 10, your Growth Rate for this month is $12/10 = 120\% (1.2)$.

Calculation: $12 / 10 = 1.2$

- ▶ If this year your value is 80, and last year it was 100, your Growth for this year is $80/100 = 80\% (0.8)$.

Calculation: $80 / 100 = 0.8$

Hour Difference

`HDiff(<Start Time>, <End Time>)`

Returns the difference between <Start Time> and <End Time> in hours.

Prev

`Prev(<Time Field> [, <N>])`

Returns the Time period Member in <Time Field> which is N periods back from the current Member. This function only works as a scope function and not by itself.

For example – This formula will return the numeric value 2 months ago:

`(<Numeric Value>, Prev(<Month Field>, 2))`

Minute Difference

MnDiff(<Start Time>, <End Time>)

Returns the difference between <Start Time> and <End Time> in minutes.

Month Difference

MDiff(<Start Time>, <End Time>)

Returns the difference between <Start Time> and <End Time> in months. Returns whole numbers.

Month to Date Average

MTDAvg(<Numeric Value>)

Returns the running average starting from the beginning of the month up to the current time period member.

The time dimension to be used is determined by the time resolution in the widget/dashboard.

Returns 0 if the active time resolution is quarters or years.

Month to Date Sum

MTDSum(<Numeric Value>)

Returns the running total starting from the beginning of the month up to the current time period member.

The time dimension to be used is determined by the time resolution in the widget/dashboard.

Returns 0 if the active time resolution is quarters or years.

Next

Next(<Time Field> [, <N>])

Returns the Time period Member in <Time Field> which is N periods after the current Member. This function only works as a scope function and not by itself.

For example –This formula will return the numeric value 2 months ahead of now:

```
(<Numeric Value>, Next(<Month Field>, 2))
```

Now

```
Now(<Day from Date field>)
```

Returns the value for the current time period. Supports day, month, quarter or year. The Now function receives a date dimension and its level and returns all the members in that dimension which match the current query execution time.

Note: This function only works as a scope function and not by itself.

The following example will return the value for the current day.

```
([Total Sales], Now([Days in Datefield]))
```

Past Year

```
PastYear( <Numeric Value> )
```

Calculates the value for the same period in the past (previous) year.

For example:

- ▶ If you're looking at a specific day, you will see the value of the same day one year back.
- ▶ If you're looking at a specific month, you will see the value of the same month one year back.

Past Quarter

```
PastQuarter( <Numeric Value> )
```

Calculates the value for the same period in the past (previous) quarter.

For example:

- ▶ If you're looking at a specific day, you will see the value of the same day one quarter back.
- ▶ If you're looking at a specific month, you will see the value of the same month one quarter back.

Past Month

```
PastMonth( <Numeric Value> )
```

Calculates the value for the same period in the past (previous) month.

For example:

- ▶ If you're looking at a specific day, you will see the value of the same day one month back.

Quarter Difference

```
QDiff( <Start Time>, <End Time> )
```

Returns the difference between <Start Time> and <End Time> in quarters. Returns whole numbers.

Quarter to Date Average

```
QTDAvg( <Numeric Value> )
```

Returns the running average starting from the beginning of the quarter up to the current time period member.

The time dimension to be used is determined by the time resolution in the widget/dashboard.

Returns 0 if the active time resolution is years.

Quarter to Date Sum

```
QTDSum( <Numeric Value> )
```

Returns the running total starting from the beginning of the quarter up to the current time period member.

The time dimension to be used is determined by the time resolution in the widget/dashboard.

Returns 0 if the active time resolution is years.

Range

```
range( <Field1>, <Field2> )
```

Returns a graphical range selector for a data set where two members of the same dimension and level define the minimum and maximum values of the range.

Second Difference

```
SDiff( <Start Time>, <End Time> )
```

Returns the difference between <Start Time> and <End Time> in seconds.

Year Difference

```
YDiff( <Start Time>, <End Time> )
```

Returns the difference between <Start Time> and <End Time> in years. Returns whole numbers.

Year to Date Average

```
YTDAvg( <Numeric Value> )
```

Returns the running average starting from the beginning of the year up to the current time period member.

The time dimension to be used is determined by the time resolution in the widget/dashboard.

Year to Date Sum

`YTDSum(<Numeric Value>)`

Returns the running total starting from the beginning of the year up to the current time period member.

The time dimension to be used is determined by the time resolution in the widget/dashboard.

Other

All

`All (<Field>)`

Ignores the scope set on the dimension.

Ordering

`ORDERING(<expression1>, <expression2>)`

Returns the numeric order position of rows sorted into ascending or descending order, breaking ties with further arguments.

The expressions must be aggregated by applying the MIN/MAX functions as in the example below:

`ORDERING(MIN([Sales Person Name]), MIN([Days in Transaction_Date]), -1*Sum([Sales]))`

Rdouble

`RDOUBLE(<R expression>, <numeric value 1>, [<numeric value 2>, ..., <numeric value n>])`

Returns a numeric result for a given R expression and a list of numeric values (use 'args[[i]]' in the R expression to reference numeric values

parameters).

The R expression is passed to the running Rserve.

The optional ordering expression determines the order in which the rows are sent to R. The argument of the Ordering parameter can be an index in your data source or you can use the ORDERING() function to determine the order of the field. For more information about the ORDERING() function, click [here](#).

For example –RDOUBLE("m <- log(matrix(unlist(args), ncol=2)); kmeans(m,3)\$cluster", [Total Cost], [Total Revenue]) will return the k-means cluster (R expression) of the args: [Total Cost] and [Total Revenue].

```
RDOUBLE (<recycle>, <R expression>, <numeric value 1>,
[<numeric value 2>, ..., <numeric value n>] )
```

recycle = TRUE (default) –Results will be cached for unchanged functions and data.

recycle = FALSE –Results will not be cached. Use this option if your R code contains randomality.

Rint

```
RINT(<R expression>, <numeric value 1>, [<numeric value 2>,
..., <numeric value n>] )
```

Returns an integer result for a given R expression and a list of numeric values (use 'args[[i]]' in the R expression to reference numeric values parameters).

The R expression is passed to the running Rserve.

The optional ordering expression determines the order in which the rows are sent to R. The argument of the Ordering parameter can be an index in your data source or you can use the ORDERING() function

to determine the order of the field. For more information about the ORDERING() function, click [here](#).

For example –RINT("m <- log(matrix(unlist(args), ncol=2));
kmeans(m,3)\$cluster", [Total Cost], [Total Revenue]) will return the k-means cluster (R expression) of the args: [Total Cost] and [Total Revenue].

RINT(<recycle>, <R expression>, <numeric value 1>, [<numeric value 2>, ..., <numeric value n>])

recycle = TRUE (default) –Results will be cached for unchanged functions and data.

recycle = FALSE –Results will not be cached. Use this option if your R code contains randomality.

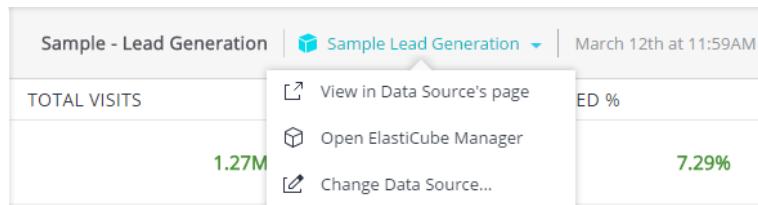
Changing a Dashboard's ElastiCube

From your dashboard, you can change your dashboard's data source. This is useful when you have recently changed your server and you need to reassign your dashboard's ElastiCube or in cases where you have deleted an ElastiCube Set and you want to reassign the dashboards in that set to your remaining ElastiCubes.

Dashboards can also support multiple data sources, which means you can have widgets built on several data sources within a single dashboard. For example, if you have widgets from an ElastiCube on a dashboard, you can add a data set to the dashboard and create widgets built on that data set's live data source. The widgets from both data sources continue to function independently of each other in the same dashboard. In addition, filters applied to fields from each data source, do not affect the fields from another data source. For more information about filtering, click [here](#).

To change your dashboard's ElastiCube:

1. In the top-left corner of your dashboard, click the ElastiCube link.



2. Hover over Change Data Source and select the new data source.

The source is added to your list. You now add new widgets from any of your data sources to the dashboard.

Embedding Dashboards and Widgets

You can embed Sisense dashboards and widgets in non-Sisense environments, such as in your own websites or applications by concatenating parameters to the URL of your dashboard or widget.

In addition, you can customize what aspects of the dashboard and widget are displayed in your environment.

For example, if you want to remove the Sisense header from the dashboard, you can concatenate the “?embed=true&h=false” parameter in the URL of the dashboard to hide the header.

`mysite.com/app/main#/dashboards/5541dc7a80a4e2181e00011a?embed=true&h=false`

In this example, the “embed=true” embeds the dashboard into your environment. When you embed a dashboard or widget, viewers can only view the dashboard or widget and do not have any editing privileges. The “h=false” parameter hides the Sisense header allowing you to whitelabel the dashboard.

This article provides a full list of the parameters you can concatenate, their functionality, and examples.

The tutorial below shows how you can embed dashboards and widgets in your environment.

Embedding Dashboards

To embed a dashboard, add “?embed=true” to the URL of the dashboard, and refresh the page. If your dashboard is within a folder,

then first delete "?folder=" and everything that follows, before adding "?embed=true".

This will open the dashboard without the surrounding Sisense environment.

For example:

Dashboard URL:

`mysite.com/app/main#/dashboards/5541dc7a80a4e2181e00011a`

Embedded URL:

`mysite.com/app/main#/dashboards/5541dc7a80a4e2181e00011a?embed=true`

Dashboard URL for dashboard within folder:

`mysite.com/app/main#/dashboards/550952417404b2981a000029?folder=550955a27404b2981a00003b`

Embedded URL for dashboard within folder:

`http://localhost:8081/app/main#/dashboards/550952417404b2981a000029?folder=550955a27404b2981a00003b`

You can use this URL to embed the dashboard in an iframe, for example:

```
<iframe id="ifm" name="ifm" width="100%" height="100%"  
frameborder="0"  
src="http://mysite.com/app/main#/dashboards/536f2b70d093e26c2  
80000d5?embed=true" scrolling="auto"></iframe>
```

Dashboard Embedding Configuration Options

You can control which aspects of the Sisense environment are available when embedding by concatenating the following parameters to the embedded URL:

Show/Hide the left navigation panel

Parameter: "h" (left)

Default: false

Example: ?embed=true&l=true

Show/Hide the right filter panel

Parameter: "r" (right)

Default: true

Example: ?embed=true&r=false

Show/Hide the toolbar

Parameter: "t" (toolbar)

Default: false

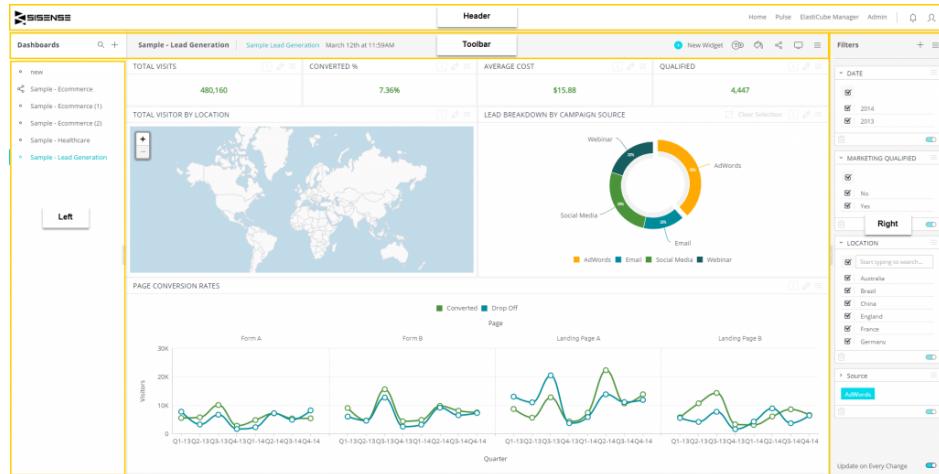
Example: ?embed=true&t=true

Show/Hide the environment header

Parameter: "l" (header)

Default: false

Example: ?embed=true&h=true



Embedding Widgets

To embed a single widget, add "?embed=true" to the URL of the widget when it edit mode.

This will open the widget without the surrounding Sisense environment.

For example:

Dashboard

URL: <http://mysite.com/app/main#/dashboards/536f3a54d093e26c28000114/widgets/536f3a54d093e26c2800011b>

Embedded URL: <http://mysite.com/app/main#/dashboards/536f3a54d093e26c28000114/widgets/536f3a54d093e26c2800011b?embed=true>

You can use this URL to embed the widget in an iframe, for example:

```
<iframe id="ifm" name="ifm" width="100%" height="100%"  
frameborder="0"  
src="http://mysite.com/app/main#/dashboards/536f3a54d093e26c28000114/widgets/536f3a54d093e26c2800011b?embed=true"  
scrolling="auto"></iframe>
```

Widget Embedding Configuration Options

You can control which aspects of the Sisense environment are available when embedding by concatenating the following parameters to the embedded URL:

Show/Hide the left data panel

Parameter: "l" (left)

Default: false

Example: ?embed=true&l=true

Show/Hide the right design panel

Parameter: "r" (right)

Default: false

Example: ?embed=true&r=false

Show/Hide the toolbar

Parameter: "t" (toolbar)

Default: false

Example: ?embed=true&t=true

Show/Hide the environment header

Parameter: "h" (header)

Default: false

Example: ?embed=true&h=true

Authentication in Embedded Dashboards and Widgets

All Sisense dashboards and widgets require authentication. When embedding dashboards and widgets outside of the Sisense environment, usually the use case is to not enforce the Sisense environment authentication, but rather use SSO (Single Sign On) with existing corporate authentication.

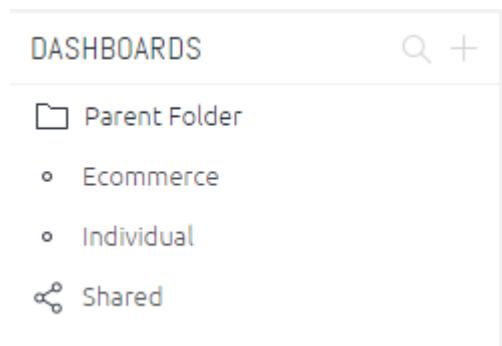
Using Dashboards

The following sections describe how to explore and interact with your dashboards:

- ▶ Viewing dashboards
- ▶ Using dashboards on mobile devices
- ▶ Filtering dashboards
- ▶ Drilling down in a dashboard to get an in-depth view for a specific field
- ▶ Making selections and focusing in on an area in the widget
- ▶ Zooming in and out of charts
- ▶ Resizing the x-axis to display labels
- ▶ Downloading a dashboard to PDF
- ▶ Downloading a dashboard as an image
- ▶ Exporting widgets to CSV/Excel
- ▶ Downloading a widget as an image

Viewing Dashboards

You can view dashboards that you created and dashboards that were shared with you. A dashboard that was shared with you appears with this  icon in the Dashboards list.



You may receive an email with a link or you may simply see a new dashboard in your Dashboards list on the HOME page.

To view a dashboard:

- ▶ Click on the link in the email. If you do not have a Sisense account, you will be automatically guided through the activation process.
OR
- ▶ Browse to your Sisense Web page, log in and click the name of the dashboard in the Dashboards list on the HOME page.

Interacting with Filters as a Viewer

Viewers can interact with filters for analyzing data. Through the Filters pane, you can modify filters in a dashboard and change the way values are sorted in a visualization. This is useful for focusing on data displayed in dashboards that have been shared by the dashboard's owner.

As a Viewer, you can select and deselect filter visuals on a dashboard. Each time you interact with a filter, for example by selecting or entering a value, the filter is immediately applied to your dashboard. After you have configured the relevant filters, you can save it as your default filters view.

Viewers can interact with filters as follows:

- ▶ Make a different selection in the filter controls displayed in the FILTER panel on the right side of the dashboard, as shown below:

Category

 Start typing to search...

- Apple Mac Desktops
- Apple Mac Laptops
- Calculators
- Camcorders
- Camera Flashes
- Car Amplifiers

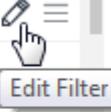
Years in Date

- 2013
- 2012
- 2011
- 2010
- 2009

OR

- ▶ Click on the pencil icon next to the filter name (shown above) in the dashboard to display the Filter Definition window.

Years in Date



- N/A
- 2013
- 2012
- 2011
- 2010
- 2009

 +

In addition, Viewers can perform the following procedures:

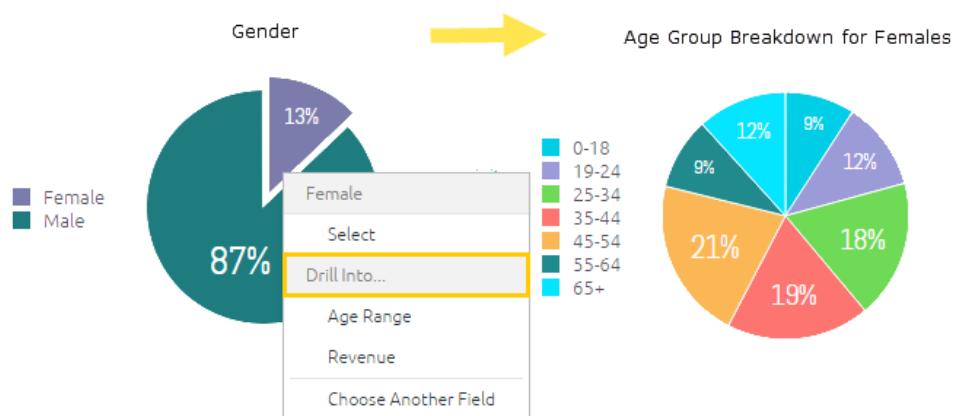
- ▶ Switching Filters On and Off
- ▶ Saving your Default Filters View

To create you must be a Designer.

Drilling Down in a Widget

You can drill down in most widgets to get an in-depth view of a selected value.

In the following example, the pie chart on the left shows a breakdown by gender. This is the original chart as created by the dashboard's designer. On the right side, the chart shows a breakdown or drill down showing age groups of the 'female' segment from the original pie chart.



With Sisense you can drill down from any field to any field, unless disabled by the dashboard's designer.

To drill down into a chart:

1. Right-click on the item in the widget into which you want to drill down.
2. To manually select the drill hierarchy path, select **Drill Into**, and then in the In the Data Browser, select the field into which to drill down. If you have already drilled into this chart, then you will have shortcuts to previously selected fields, or select **Choose Another Field** to select a different field for the first time.

or

Select a predefined drill hierarchy (if available).

To drill up to a higher level, click on a breadcrumb. To drill all the way up, click on the X icon.



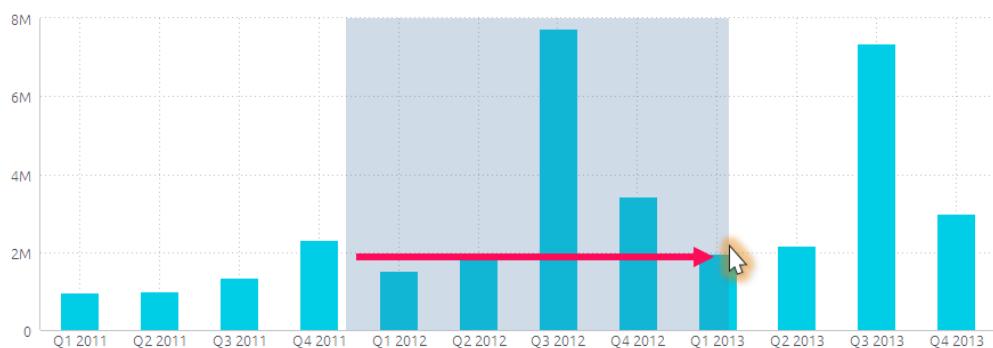
Making Selections in a Widget

You can click on a specific portion of a widget to select it. Doing so filters the dashboard according to the selected data by adding a filter to the dashboard's FILTERS panel.

You can also use the selection to drill down in the widget (see Drilling Down in a Widget).

To select an item in a widget for filtering:

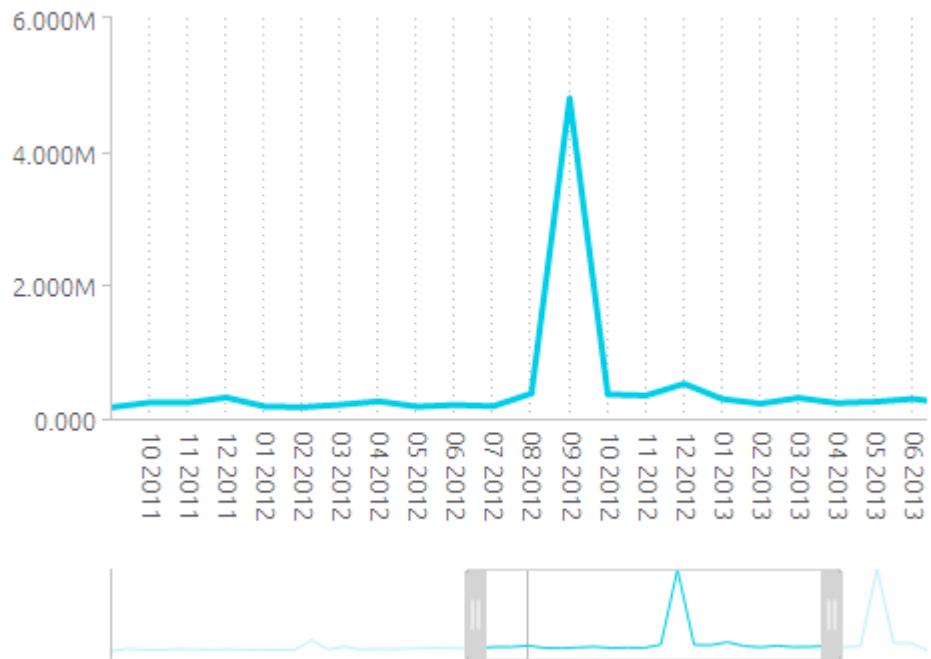
1. Left-click an item in a widget to automatically select it and add a dashboard filter according to the selection.
2. For multiple selection, use the Ctrl key. When released, a menu will appear. Click **SELECT**.
3. In chart widgets, you can drag and draw the area to be selected, as shown below:



Zooming In/Out in Charts

When a widget contains more data than can comfortably be displayed in one view, a zoom bar may be displayed under the widget. The zoom bar enables you to scroll right and left and to zoom in/out of different parts of the widget.

The dark gray vertical bars to the left and right of the scroll bar represent what is shown in the widget:



To zoom into an area of the chart:

- ▶ Drag the bars closer together and position them immediately to the right and left of the area to be displayed.

To zoom out:

- ▶ Drag the bars further apart.

Resizing the X-AXIS to Expose Labels

The x-axis on the chart can be resized to reveal long or hidden labels.

Drag the x-axis up or down to reveal more or less of the labels.

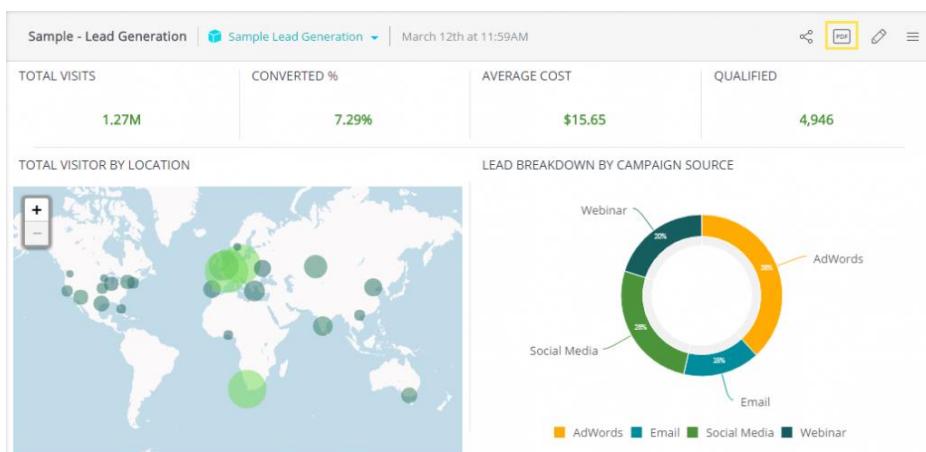
Creating PDF Reports

When you need to take copies of your dashboards with you for meetings or sharing with others, you can generate a PDF report of your dashboard.

By clicking the  icon on your dashboard, you can customize your dashboard's appearance through the PDF Report Settings page. After you have defined how your dashboard is to be displayed, you can download your dashboard locally as a PDF.

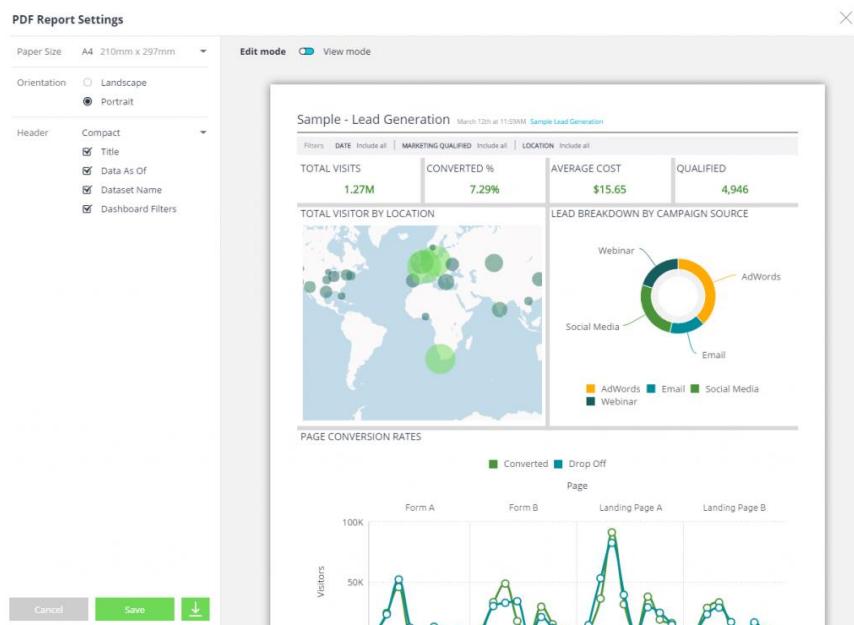
To create a PDF report:

1. From your dashboard menu, click the PDF icon .



The PDF Report Settings page is displayed.

2. Customize how your dashboard is displayed in the PDF. For more information, see [Customizing PDF Reports](#).



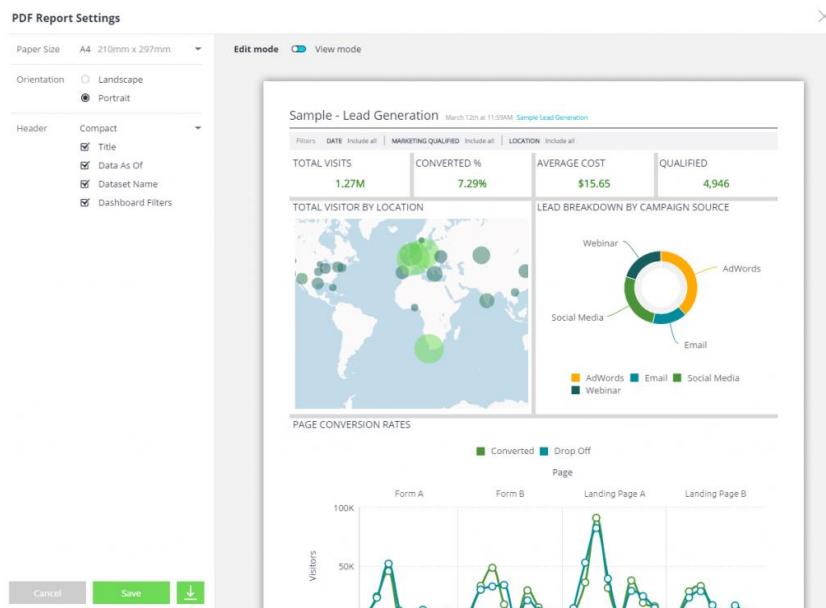
3. Click . The dashboard is downloaded locally as a PDF file.

Customizing PDF Reports

If you need to share dashboards with other users, or create a hardcopy for yourself, Sisense allows you to send an email report to your users or export your dashboard to PDF. Dashboards in email reports or PDFs though look different compared to online dashboard as they serve different purposes. Through the Sisense PDF Report Settings, you can customize and create nicely formatted email reports and PDFs according to your requirements. The Sisense PDF Report Settings allow you to define both the content and design of your report quickly and easily.

From the PDF Report Settings page, you have two modes, Edit mode, for customizing your PDF, and View mode, for seeing how the

dashboard will be displayed in the PDF. In Edit Mode, all the settings you can apply to your report are displayed in the left menu.



In Edit mode, you can set the size of your report, your dashboard's orientation, its layout, and what is displayed in the header and footer on your report.

After you have customized your report, you can save the settings and use them when sharing reports, or download the report as a PDF.

To customize a PDF report:

1. In your dashboard's menu, click . The PDF Report Settings page is displayed.
2. Toggle the mode to **Edit Mode**.
3. Edit the dashboard by selecting any of the following options:

Setting	Description
Paper Size	The size of the email report in the PDF.

Setting	Description
Orientation	<p>The orientation of the dashboard in the PDF, landscape or portrait. Landscape orientations display the dashboard horizontally while portrait orientations display the dashboard vertically.</p>
Header	<p>The design and content of your report's header. The header is displayed on the top of each page of your PDF report.</p> <p>From the Header list, you can define how your header is to be displayed: Compact, Medium, Large, or No Header.</p> <p>After you have set the design, you can determine what information is included in the header.</p> <p>Title: Select to display the dashboard name at the top of the PDF.</p> <p>Dashboard As Of: Select to display the ElastiCube name and last build time at the top of the PDF.</p> <p>Dataset Name: Select to display the name of the ElastiCube that contains the dashboard's data.</p> <p>Dashboard Filters: Select to display dashboard filter selections.</p>

- Click **Save** to save your settings when sharing reports

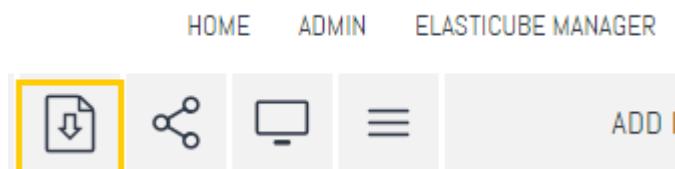
or  to download your report.

Downloading a Dashboard as an Image

You can download the dashboard as an image. This can be useful for including the dashboard in a presentation, web site, etc.

To download the image:

1. Click the download button at the top right of the screen, and click **Download Image**.



2. You can also define the following additional settings:

- ▶ **Show Dashboard Title:** Turning this on will display the dashboard name at the top of the PDF.
- ▶ **Show Dashboard Filters:** Turning this on will display a summary of the dashboard filter selections at the top of the PDF.
- ▶ **Show ElastiCube Info:** Turning this on will display the ElastiCube name and last build time at the top of the PDF.



SISENSE

Download Dashboard

File Type PNG (Better Quality) ▾

Size (Width) Current (1334 Pixels) ▾

Data

Show Dashboard Title

Show Dashboard Filters

Show ElasticCube Info

[Download](#)

[Cancel](#)

Exporting Widgets to CSV/Excel

You can export widgets in the CSV format. This is also useful for using with Excel.

Only the displayed (filtered) data is exported according to the currently selected filters.

To export a widget to csv:

- ▶ In dashboard view, click on the widget's menu, and select **Download > CSV File**.
- ▶ In edit widget view, click on the download icon, and select **Download CSV**.

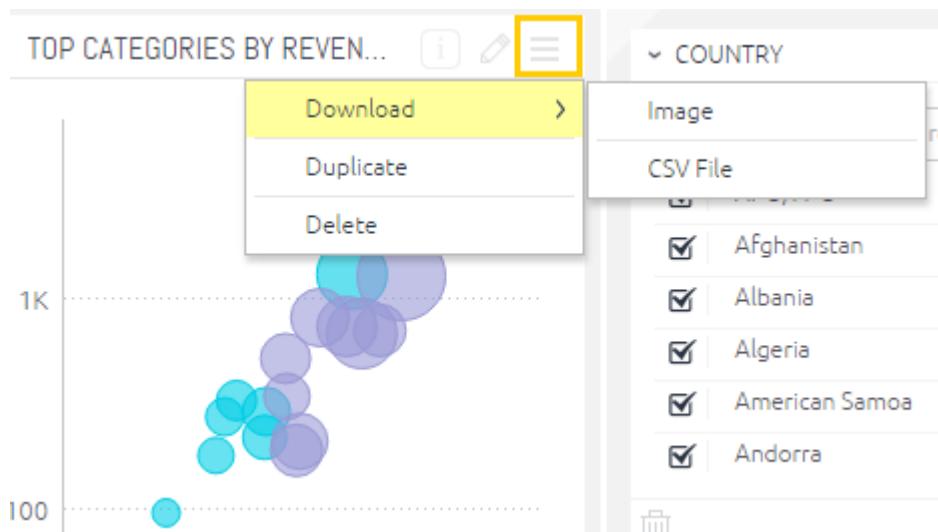
Downloading a Widget as an Image

You can download a widget as an image (png format).

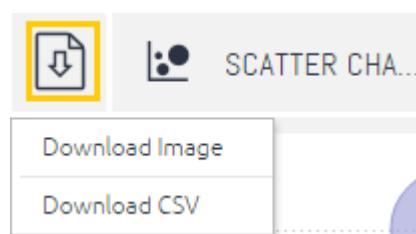
To download the widget as an image:

- ▶ In dashboard view, click on the widget's menu, and select **Download > Image**.
- ▶ In edit widget view, click on the download icon, and select **Download Image**.

When viewing the dashboard



When viewing the widget



Tip: The image size will reflect the size of the widget on the screen when you download it. To create a larger image, we recommend



SISENSE

enlarging the widget space in the dashboard, or to open the widget in edit view, to get the bigger picture.

Managing and Sharing Dashboards

Organizing Dashboards

This chapter describes how to manage and share Dashboards.

This chapter is only for Sisense users who have dashboard editing rights (Designers).

Organizing Dashboards in Folders

You can organize your dashboards into folders and sub-folders within any of your folders to improve how your viewers navigate the Dashboard list.

To create a folder:

1. In the Dashboard list, click + and select **New Folder**.



The new folder is added to the dashboards list.

2. Hover over the new folder and click  > **Rename** to name the folder.

To create additional sub-folders within your folders, hover over the folder and select > **New Folder** and repeat Step 2.



Locating a Dashboard

You can quickly and easily locate dashboards without having to navigate through your entire list of dashboards. When you enter any characters in the **Search** field, the list of dashboards is filtered according to the characters found within your dashboard titles.

To locate a dashboard:

1. In the Dashboard list, click  to display the **Search** field.



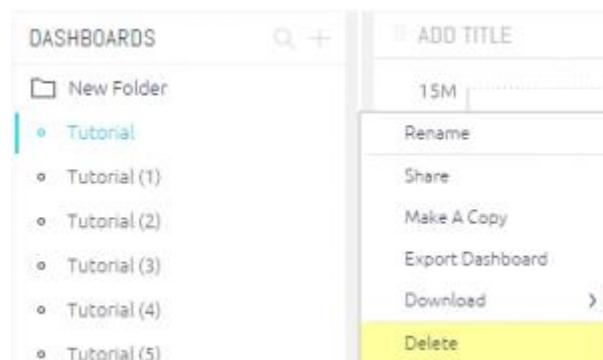
2. In the **Search** field, enter the title of your dashboard. Any dashboards that contain the characters you entered are displayed.



Deleting a Dashboard

To delete a Dashboard:

- Click on the menu  button of the Dashboard to be deleted in the list in the HOME page, and select **Delete**.



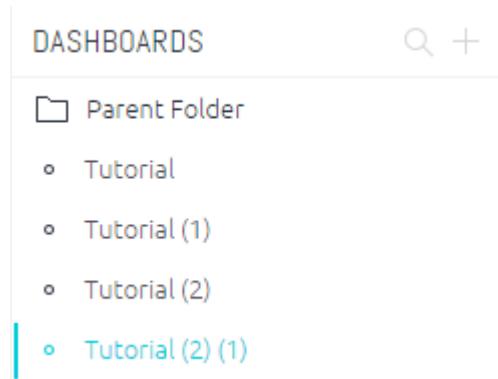
OR

Click the Dashboard Menu  button and select **Delete Dashboard**.

Copying a Dashboard

Copying a dashboard (even one that was shared with you) creates a new copy of the dashboard with a different name. This new dashboard is totally separate from the original and you are its owner, meaning that you have full editing rights. When you copy a dashboard, the new dashboard is added below the copied dashboard in your Dashboard list with the same name plus a number in the order it was created.

For example, Tutorial(1) and Tutorial(2) are copies of Tutorial, while Tutorial(2)(1) is a copy of Tutorial(2):



To copy a Dashboard:

- ▶ Right-click the dashboard's name in the Dashboard list in the HOME page and select **Make a Copy**.

Exporting and Importing Dashboards

Sisense supports exporting and importing of dashboards. This is useful for copying dashboards between separate Sisense installations, and also for making backups.

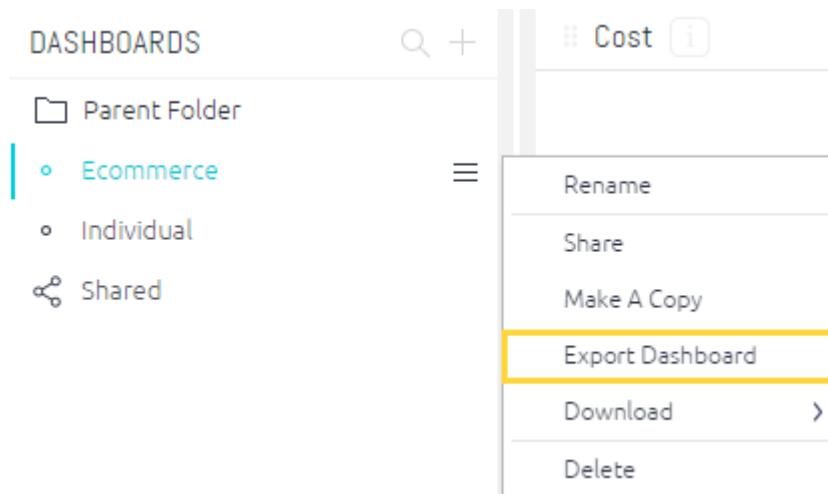
Dashboards are exported as files with a 'dash' extension, which can then be imported.

Exporting a Dashboard

To export a dashboard, do one of the following:

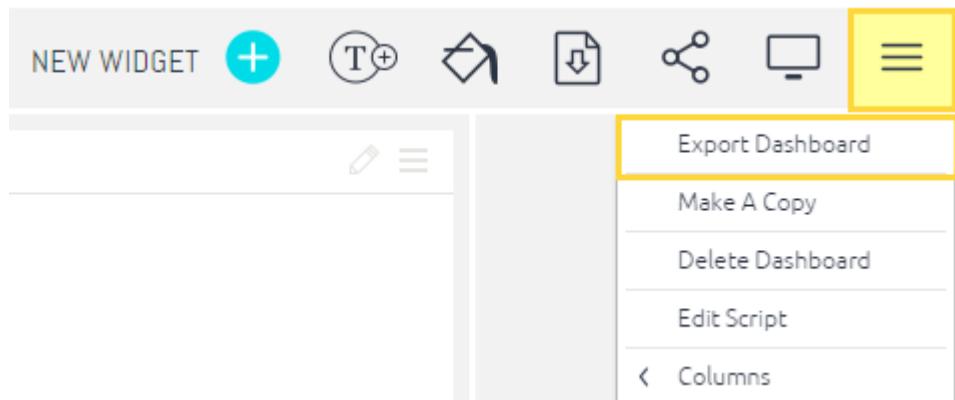
Option #1

In the Dashboard list on the left of the Sisense environment, click the dashboard's menu and click Export Dashboard.



Option #2

Open the dashboard's menu and click **Export Dashboard**.

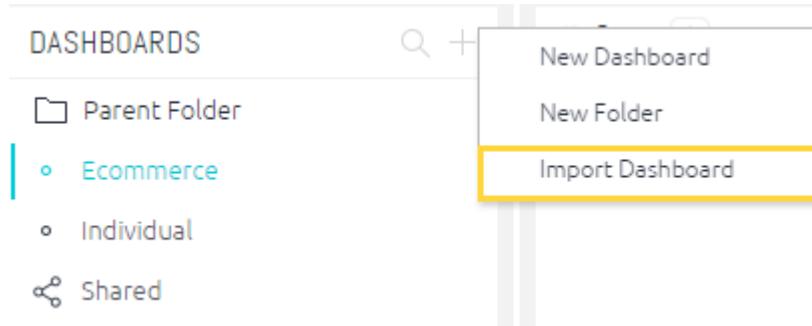


Both of these methods will prompt you to choose a name for the exported dash file and save it to disk.

Importing a Dashboard

To import a dashboard:

1. Hover over Dashboards list in the home page and click the '+' symbol. Select **Import Dashboard** from the menu. You can also right-click anywhere in the dashboards list, or click on the menu of one of your existing folders to import a dashboard into a folder.
2. Select your exported dash file. The dashboard will be added to your list of dashboards or folder. You can drag it from there to your desired location in the tree.



Sharing Dashboards

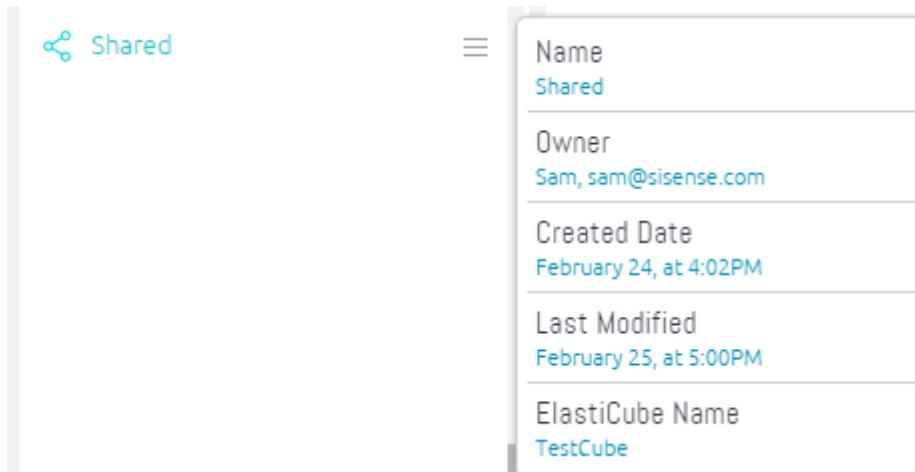
Owners can share dashboards that they have created with other Sisense users(Viewers and Designers). An owner is any Designer who has created a dashboard. When you share a dashboard, you can define whether the recipient has editing rights:

- ▶ Designers have editing rights.
- ▶ Viewers have viewing/filtering rights.

After you share a dashboard, or for any dashboard shared with you, in the Dashboards list the  icon appears next to the dashboard.

-  [Tutorial \(2\)](#)
-  [Tutorial \(3\)](#)

You can hover over the dashboard in the Dashboards list to view ownership details.



The screenshot shows a dashboard list item for "Tutorial (3)". A blue "Shared" badge with a share icon is displayed next to the list item. To the right, a detailed card provides ownership information:

Name	Shared
Owner	Sam, sam@sisense.com
Created Date	February 24, at 4:02PM
Last Modified	February 25, at 5:00PM
ElastiCube Name	TestCube

When you share your dashboard, you can configure Sisense to send periodic email notifications containing the dashboard inside the email's body to your dashboard's recipients.

Working with Shared Dashboards

A dashboard recipient can edit the dashboard if you grant them permission, **but the changes they make in the dashboard do not affect your copy of the dashboard.** The recipient's changes are automatically performed on a separate copy of the dashboard (to which only they have access).

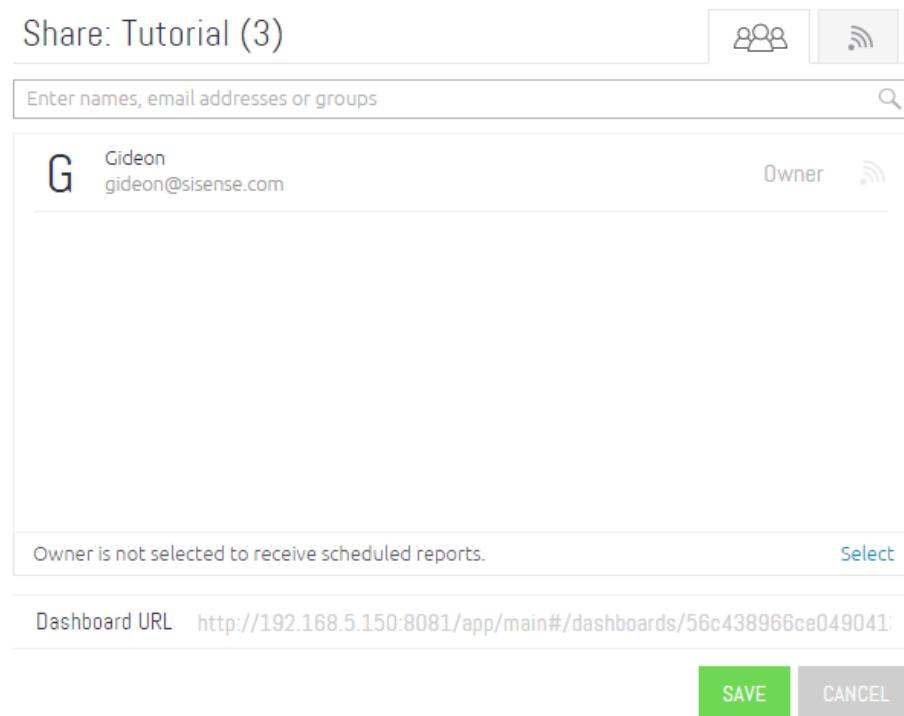
Dashboard recipients can revert to your latest shared copy at any time by right-clicking the dashboard's name in the Dashboards list on the left of the Sisense environment and selecting **Restore Dashboard**.

When you share a dashboard that's contained within a folder, the dashboard is shared using the same structure (under the same folder).

When a Designer shares a dashboard with you, you become a recipient and all of the above applies to you.

To share a Dashboard:

In the heading of your dashboard, click the **Share**  button.



OR

In the Dashboards list, select  > **Share** for the dashboard you want to share.

Enter the emails or groups of users with whom to share the dashboard. Selecting **Everyone** specifies that this dashboard is shared with all Sisense users and user groups.

The contact information of the people you add is listed in the center of the window and is displayed the next time you open this window.

If a person does not have a Sisense account, they receive a link that guides them through the activation process.

Sisense users are defined by your Sisense administrator, as described in [Managing Dashboard Users](#).

In the dropdown menu next to each user/user group select whether the user is a designer with editing rights (**Can edit**) or a viewer (**Can view**).

Toggle the Email Reports  button next to each user/user group on/off to define whether the user receives scheduled email reports of the dashboard. Read more on email reports below.

The **Dashboard URL** field at the bottom displays a direct link to this dashboard that you can copy/paste.

Click **Save**.

Publishing Dashboards to Recipients

When an owner modifies a dashboard, for example, by adding a new widget, the dashboard is not automatically updated for recipients. The owner of a dashboard must publish their updated dashboard to update the dashboards of their recipients and the dashboard recipients must refresh the dashboard before any of the owner's changes are applied.

To publish a dashboard to your recipients:

- ▶ In the heading of your dashboard, click **REPUBLISH**.

OR

In the Dashboards list, right-click the dashboard's name and select  > **REPUBLISH**.

Updating a Shared Dashboard

When a recipient modifies a shared dashboard, those changes remain local unless the recipient updates their dashboard to match the

dashboard of the owner. If the recipient updates their shared dashboard to match the dashboard of the owner, any of their local changes to the dashboard are overwritten. Sisense recommends that recipients save any local versions of a shared dashboard as a new dashboard before updating a dashboard.

To update a shared dashboard:

1. In the heading of your dashboard, click  > **Restore Dashboard**.

OR

In the Dashboards list, right-click the dashboard's name and select  > **Restore Dashboard**.

2. Click **YES** to confirm that you want to update the shared dashboard.

Changing Ownership of a Dashboard

The owner of a dashboard can transfer ownership of the dashboard to an Admin or another Designer. Transferring ownership of dashboards can be useful when the original owner leaves your company and you do not want to lose their dashboards.

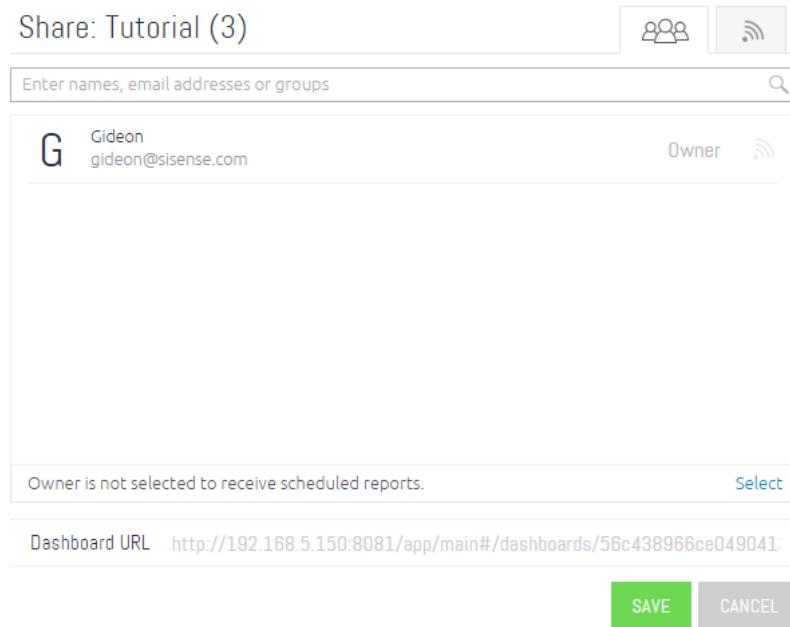
When you transfer ownership, the dashboard remains as a shared dashboard within your Dashboards list, and you become a dashboard recipient. As a dashboard recipient, you are limited to how you can modify the dashboard as described in Working with Shared Dashboards.

To transfer ownership of a dashboard:

1. In the heading of the dashboard you want to transfer ownership of, click the **Share** button.

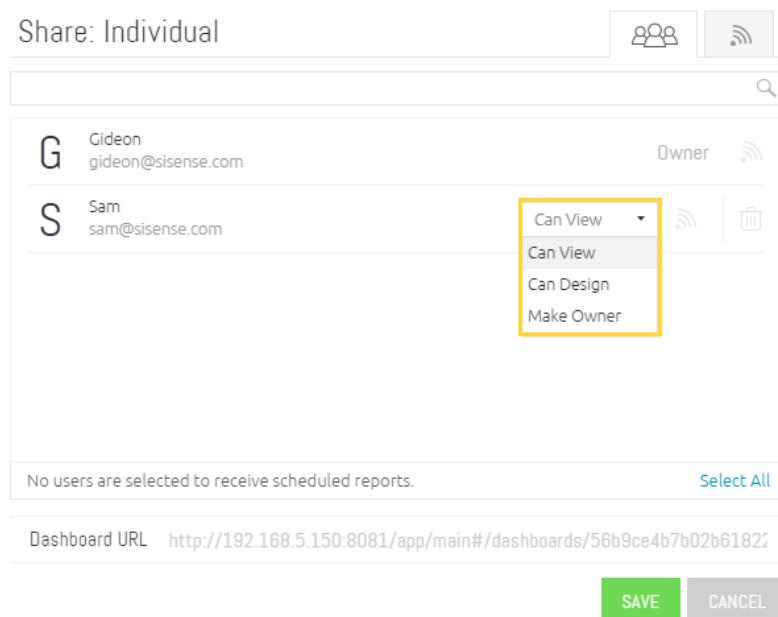
OR

In the Dashboards list, select  > **Share** of the dashboard.
The Share window is displayed



2. Enter the name of the Admin or Designer you want to transfer ownership to.

3. In the User Privilege list of the user you want to make an owner, select **Make Owner**.



The screenshot shows the 'Share: Individual' interface. At the top, there are icons for users and a search bar. Below that, a list of users is shown: Gideon (gideon@sisense.com) and Sam (sam@sisense.com). For each user, there is a dropdown menu under 'Owner' with options: Can View, Can View, Can Design, and Make Owner. The 'Make Owner' option is highlighted with a yellow box. At the bottom, it says 'No users are selected to receive scheduled reports.' and has 'Select All' and 'Dashboard URL' (http://192.168.5.150:8081/app/main#/dashboards/56b9ce4b7b02b61822). There are 'SAVE' and 'CANCEL' buttons at the bottom right.

4. Click **Save**. A Warning message is displayed.
5. Click **OK** to confirm that you want to transfer ownership to the user.

Sending Dashboard Email Reports

After you share your dashboard, you can configure the Sisense Scheduler to send periodic email reports containing a static version of the dashboard that can be clicked to access the dashboard or PDFs containing images of your dashboard.

From your dashboard, you can configure how many emails the Scheduler sends and when.

To test how the report is to be displayed to your shared users, in the Shared Dashboard window, click the **Send me a report now** link, which generates an email report sent to your email address immediately.

To schedule email notifications:

1. In your dashboard, click the Share icon to display the Share window.



2. In the Shared Dashboard window, click the tab.

Share: Dashboardname

Email Report Content

Email report

PDF attachment [Edit / View PDF Report](#)

Schedule

Every ElastiCube update, no more than emails per day

Daily: at Eastern Time (US and Can...)

[Send me a report now](#) Save Cancel

3. Select the type of report you want to send. There are two types of email reports you can send:

Email Report: The email report is attached as an image file.

PDF Attachment: The email report is attached as a PDF. You can define how the dashboard is to be displayed in the PDF through the PDF Report Settings page by clicking Edit/View PDF Report. For more information, see Customizing PDF Reports.

4. In the Schedule area, select the frequency for sending email notifications. You can select one of the following options:

Option	Description
Every ElastiCube update, no more than X emails per day	<p>Select to send a report each time you update the ElastiCube up to the amount of emails you define in the Number box.</p> <p>In the Number box, type or select the amount of emails that can be sent a day, regardless of the amount of builds.</p>
Daily	<p>Select to send one email report according to the time you define.</p> <p>In the Everyday box, select the days the scheduled report is to be sent. You can select multiple days by clicking the day.</p> <p>Select the Hour and the Time Zone that the email is to be sent on the days you selected.</p>

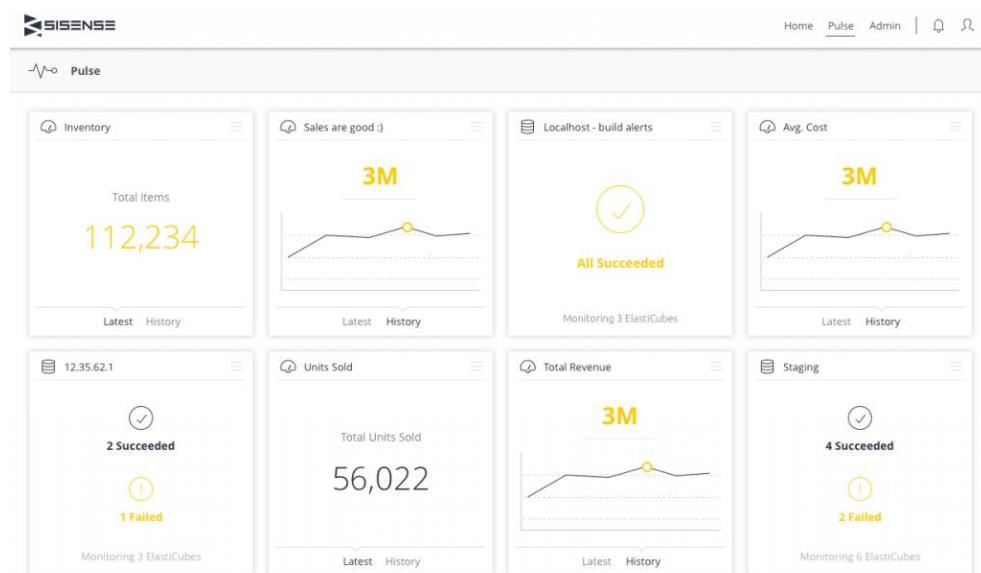
5. Click **Save**.

Sisense Pulse and Alerting

Sisense Pulse is a centralized location where you can stay on top of your most important KPIs across multiple dashboards or manage your data and build alerts.

For example, you can consolidate and monitor important aspects of your business by adding Indicator widgets from various dashboards to the Sisense Pulse page.

Then, you can create alerts to notify you when certain thresholds are met or anomalies in your data are detected. Sisense Pulse provides you with access to your most important data and notifies you when to take action.



The following sections describe Sisense alerts and managing alerts and KPIs in your Sisense Pulse page.

Sisense Alerts

Sisense offers two types of alerts you can define, data alerts and system alerts.

Data alerts are triggered by changes to your data. Sisense provides several options for defining the type of changes that trigger data alerts including threshold based alerts, automatic anomaly detection, and Always which is triggered by any change to your data. For example, if you have an Indicator widget that represents total sales, Sisense can notify you when a threshold you defined is met or exceeded.

System alerts can be configured to be sent following build attempts.

When creating an alert, you can customize your notifications, the channel through which they are sent, and how often a notification is sent following a data or system event.

After you create an alert, the alert is added to the Sisense Pulse page where you can manage the alert and its notifications. For example, you can delete alerts or enable and disable notifications.

If you want to make a similar alerts, for example, alerts with different notifications for different groups of users, you can duplicate the alert and modify its settings.

When an alert is triggered, you and your users are notified through the channel you defined when creating the alert and through the Pulse Feed. The Pulse Feed is displayed in the Sisense Web Application and shows a list of your most recent triggered alerts.

Creating Data Alerts

Data alerts are triggered by conditions you define for your Indicator widgets. For example, if you have an Indicator widget that represents your company's revenue, you can define an alert that notifies you when this revenue reaches or falls below a certain threshold. When creating data alerts, you define the condition that triggers the alert. Sisense provides three alert conditions that can trigger an alert:

Threshold

Sisense notifies you when a threshold is reached following a build. This condition is useful for monitoring metrics that follow a regular, cyclical schedule.

Automatic

Sisense automatically notifies you when the Sisense alert engine identifies an anomaly in your data based on machine learning algorithms. The Sisense algorithm actively learns from your data and eliminates noise to identify anomalies in your data. Sisense's anomaly detection is based on a smoothing algorithm that takes the latest values into account. The more times an ElastiCube is built, the more accurate Sisense becomes at identifying any anomalies and notifying you.

Always

Sisense notifies you any time there is a change to your widget's data. This condition is useful for metrics that do not frequently change.

In addition, you can configure advanced settings that determine who receives the alert and what channel the alert is sent through. For more information see Advanced Settings.

To create a Data alert:

1. From a indicator widget's menu, select **Add To Pulse**.
2. The Add to Pulse window is displayed.

Add to Pulse

<p>Name</p> <input type="text" value="Total Quantity"/>	<p>Condition</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  Automatic </div> <div style="text-align: center;">  Threshold </div> <div style="text-align: center;">  Feed </div> </div>	<p>We will monitor this value for you and let you know when there are irregularities.</p>
		Advanced ADD CANCEL

3. In the **Name** field, enter a meaningful name that represents your alert.

In the Alert Condition area, select the condition that triggers the alerts.

Threshold: Sisense defines the threshold value as the current value displayed in the widget. You can increase or decrease this value and define the equality and relational operators that determine when the threshold is met.

Add to Pulse

Name

KPI

Alert Condition



Threshold



Automatic



Always

You will be notified when the value is:

Greater than ▾

0.072920551856 ▾

[Advanced](#)

Add

Cancel

Automatic: Sisense automatically identifies anomalies in your data and alerts you.

Alert Condition



Threshold



Automatic

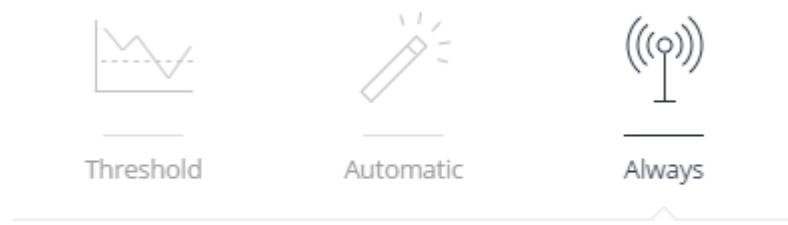


Always

The system will monitor this value and notify you when anomalies are detected.

Always: Sisense notifies you every time the data changes.

Alert Condition



You will be notified about this measure
whenever it updates.

- Click **ADD** to create the alert. The alert is added to Sisense Pulse.

For additional advanced configuration options, see Advanced Settings.

Creating Build Alerts

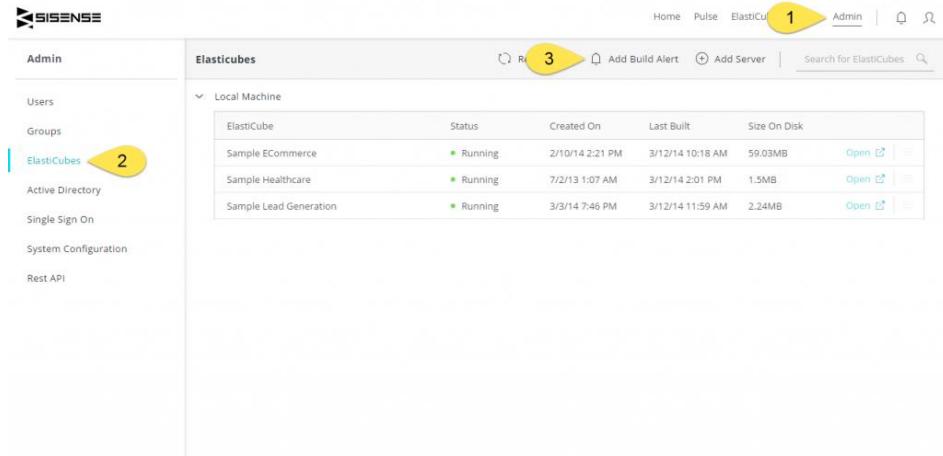
Administrators can set build alerts to notify themselves or other users when a build fails or succeeds. Build alerts are created from the Sisense Web Application and are sent following an ElastiCube build.

Build alerts allow you to monitor the status of ElastiCube builds.

Administrators can create build alerts from the Admin section of the Sisense Web Application. After a build alert is created, it is added to Sisense Pulse from where you can manage it.

To create a build alert:

- In the Sisense Web Application, select **Admin > ElastiCubes > Add Build Alert**.



The screenshot shows the Sisense Admin interface. On the left, there is a sidebar with the following navigation items:

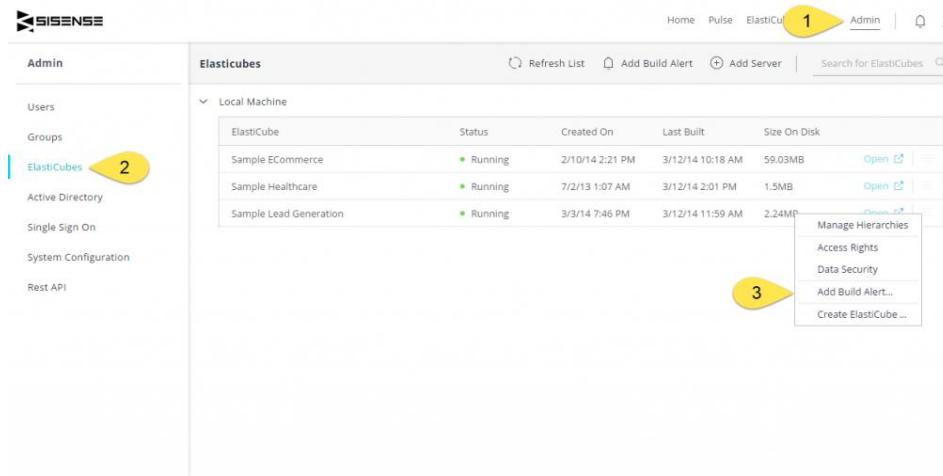
- Admin
- Users
- Groups
- ElastiCubes** (highlighted with a yellow arrow labeled 2)
- Active Directory
- Single Sign On
- System Configuration
- Rest API

The main content area is titled "ElastiCubes" and displays a table of "Local Machine" ElastiCubes. The table has columns for ElastiCube, Status, Created On, Last Built, and Size On Disk. Three rows are listed:

ElastiCube	Status	Created On	Last Built	Size On Disk
Sample ECommerce	Running	2/10/14 2:21 PM	3/12/14 10:18 AM	59.03MB
Sample Healthcare	Running	7/2/13 1:07 AM	3/12/14 2:01 PM	1.5MB
Sample Lead Generation	Running	3/3/14 7:46 PM	3/12/14 11:59 AM	2.24MB

At the top right of the main area, there are buttons for Refresh List (yellow arrow labeled 1), Add Build Alert, Add Server, and Admin. A search bar for "Search for ElastiCubes" is also present.

OR
From the ElastiCube menu, Add Build Alert.



This screenshot is similar to the one above, showing the Sisense Admin interface with the ElastiCubes menu selected (arrow 2). However, a context menu is open over the first row of the ElastiCubes table (arrow 3). The menu options are:

- Manage Hierarchies
- Access Rights
- Data Security
- Add Build Alert...
- Create ElastiCube ...

The Add Build Alert window is displayed.

2. In the **Name** field, enter a name for the alert.

Add Build Alert





Name	<input style="width: 100%; height: 25px; border: 1px solid #ccc; padding: 5px;" type="text" value="Type name..."/>
Server	localhost
ElastiCube	Sample Healthcare
Notify when	<input checked="" type="checkbox"/> Build failed <input type="checkbox"/> Build succeeded <input checked="" type="checkbox"/> Build succeeded after failure

Cancel

3. From the **Server** list, select the server where the ElastiCube is hosted.
4. From the **ElastiCube** list, select the ElastiCube(s) that triggers an alert following a build.
5. Select when to notify you following a build. There are three conditions that when met, can trigger an alert:
 - Build failed:** The alert is sent after the ElastiCube build fails.
 - Build succeeded:** The alert is sent after the ElastiCube is built successfully.
 - Build succeeded after failure:** The alert is sent following a successful build after the ElastiCube previously failed to build.
6. Click **SAVE**. The Alert is added to your Sisense Pulse page.

7. For additional advanced configuration options, see Advanced Settings.

Advanced Settings

By default, when an alert is triggered, you are notified via your email address registered with Sisense.

Through the Advanced Settings options, you can notify additional users when an alert is triggered, notify them across multiple channels, and customize the message displayed in the notification.

The following pages describe the functionality available in the Advanced Settings options:

- ▶ Notifying Additional Users
- ▶ Setting Notification Channels
- ▶ Customizing Notification Messages
- ▶ Configuring Additional Options

Notifying Additional Users

When an alert is triggered, Administrators and Designers can notify additional Sisense users and groups by configuring the alert's advanced settings. Viewers cannot share alerts with additional users.

Users who are notified receive a notification through Sisense Pulse and any other additional channels you define.

To notify additional users:

1. For Build Alerts, select the  tab.

Add to Pulse

Name	Value
Converted (%)	Converted (%)

Filters

3 active filters [Show](#)

Alert Condition - Threshold *

Greater than *

[Basic](#) [Add](#) [Cancel](#)

For Data Alerts, select **Advanced Options >** 

Add to Pulse

Name

Total Revenue

Condition



Automatic



Threshold



Feed

We will monitor this value for you and let you know when there are irregularities.

[Advanced](#)

[ADD](#)

[CANCEL](#)

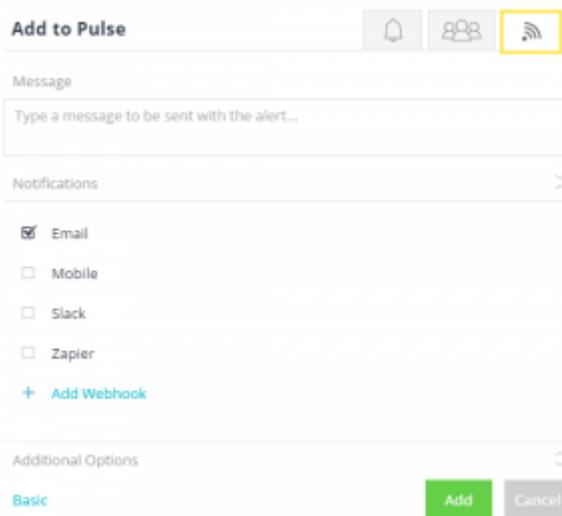
2. Enter the email address, Sisense username, or group name of the users you want to notify when an alert is triggered. Entering **Everyone** specifies that all Sisense users and user groups associated with your account are notified. The contact information of the people you add is listed in the center of the window and is displayed the next time you open this window.
3. Click **Add** to add the alert to Pulse.

Setting Notification Channels

By default, your users are notified via their email addresses registered with Sisense. In addition, Sisense supports a variety of workflows by sending notifications across multiple applications and services. For example, your users can receive notifications through the Sisense mobile application, Zapier, Slack, and other 3rd services via webhooks.

To define how users are notified:

1. For Build Alerts, select the  tab.



For Data Alerts, select **Advanced Options** > .

Add to Pulse

Name

Total Revenue

Condition



Automatic



Threshold



Feed

We will monitor **this value** for you and let
you know when there are irregularities.

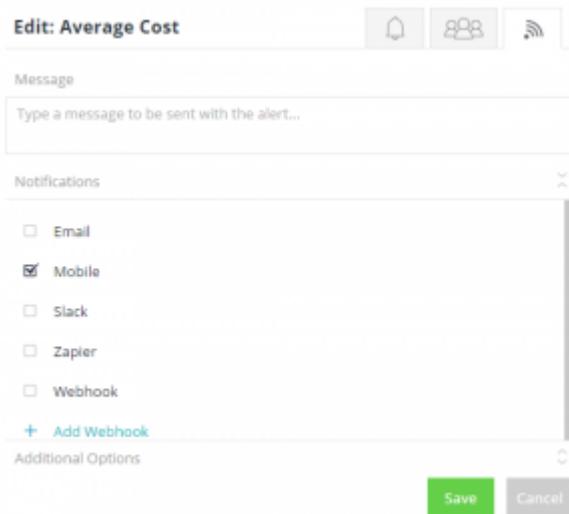
Advanced

ADD

CANCEL

2. In the Notifications area, select the relevant channel for your notification. The default channel, Email, is already selected. You can select one or more of the following channels:

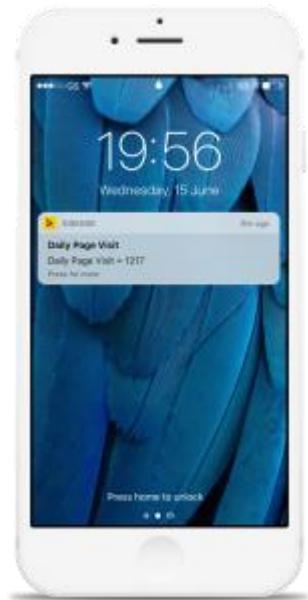
- ▶ Email
- ▶ Mobile
- ▶ Slack
- ▶ Zapier
- ▶ Webhook



3. Click **Save**.

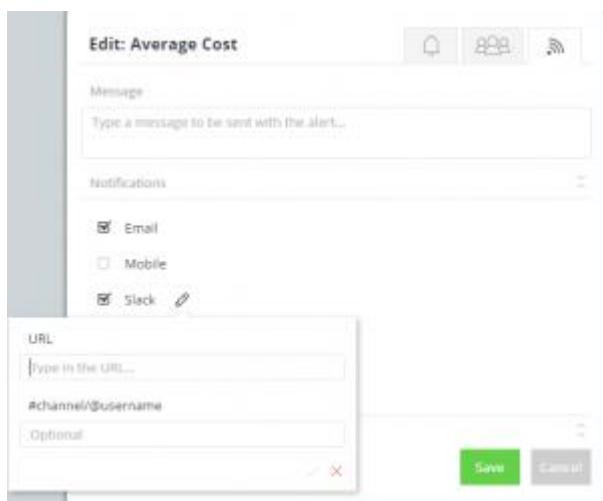
Sisense Mobile App

The Sisense Mobile app enables users to view Sisense dashboards through their mobile devices. You or your users can receive notifications through the Sisense Mobile app. After selecting the users you want to notify, any users with a Sisense user account and the Sisense Mobile app will receive a notification when an alert is triggered.



Slack

Slack is a real-time team collaboration messaging platform that allows you to receive incoming Webhooks from external sources such as Sisense. To send notifications through Slack, you need to enter the URL and optionally, your Slack channel or username, where the notification is to be displayed.



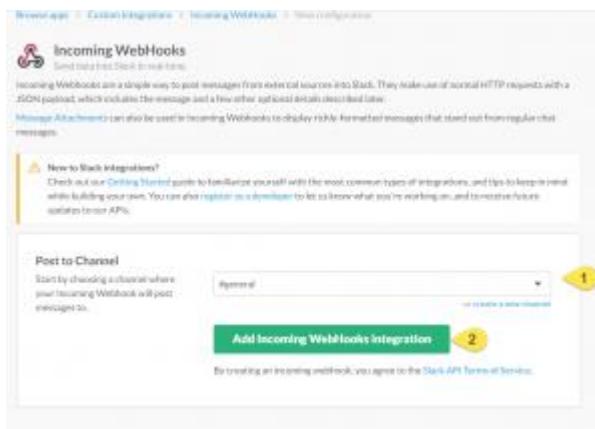
You can obtain these details by adding the Incoming WebHooks app into your Slack configuration.

To obtain a Webhook URL and channel from Slack:

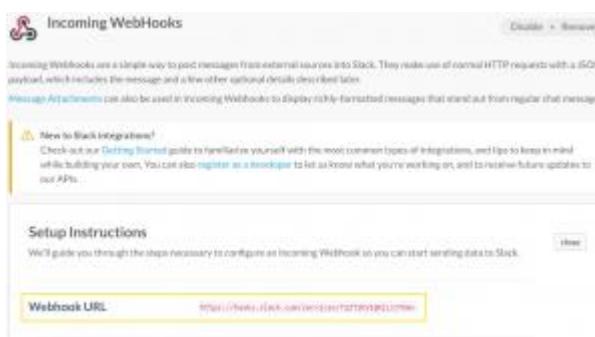
1. Add the Incoming WebHooks app to Slack. You can find this app in Slack's App Directory.
2. In the Incoming WebHooks app page, click **Add Configuration**.



3. After you add the Incoming WebHooks app, you can select which channel to post the notifications to. The channel name is optional and displays the notification in the channel associated with your WebHook URL. In the Post to Channel area, select the channel where the notification is to be displayed and click **Add Incoming WebHooks integration**.



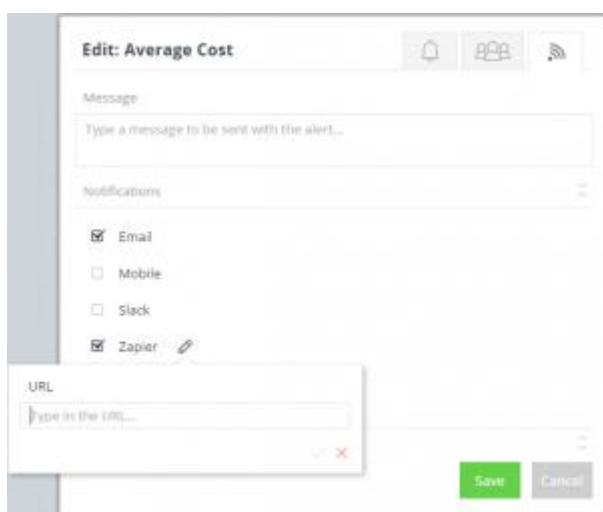
4. In the Setup Instructions the Webhook URL is displayed. Copy the Webhook URL.



5. In Sisense when sending notifications through Slack, in the **URL** field, enter the URL copied in the previous step and in the **#channel/@username** field, enter the channel name if you have defined one.

Zapier

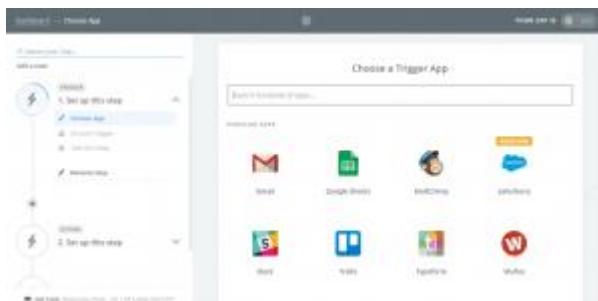
Zapier is an online service that lets you create automated actions connecting disparate business apps and services such as Sisense. To receive notifications through Zapier, you must create a Zap and provide Sisense with the relevant URL when configuring your alerts' notifications.



When an alert is triggered, Sisense sends the notification to the Webhook you define in Zapier.

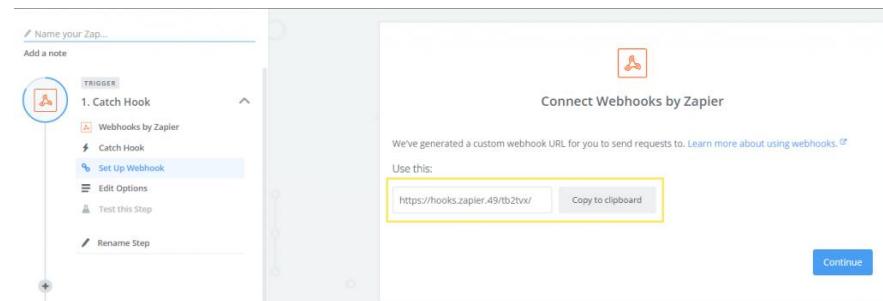
To obtain a Webhook URL from Zapier:

1. Log in to Zapier and click **Make a Zap**. The Trigger App page is displayed.



2. From the Trigger App page select Webhooks. The Connect to Webhooks by Zapier page is displayed.

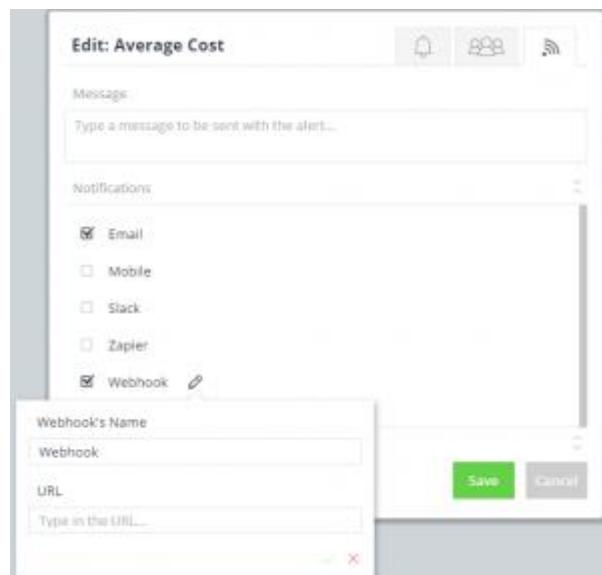
3. In Step 1 > Set Up Webhook, click **Copy to clipboard** to copy the Webhook URL you need to provide to Sisense.



4. In Sisense, when sending notifications through Zapier, in the **URL** field, enter the URL copied in the previous step.

Webhooks

Webhooks are a system of automated notifications that indicate that an event occurred, in this case an alert being triggered. To send notifications through additional 3rd party channels, you can select **Webhook** and enter the Webhook name and URL.



Your Webhook server listens for incoming messages from Sisense across the URL specified in the URL field.

When Sisense sends notifications to your custom Webhook, they are sent within a JSON payload. Your Wehbook provider must interpret the payload and display the notification to your users. The following are examples of notifications sent in a JSON payload and a description of the contents of the payload:

Data Events

```
{
  type: 'kpi',
  name: 'Total Sales',
  message: 'Hello World',
  originDashboardLink:
  '58128745677f52dc0f000019',
  measure: 'Revenue',
  value: 39759625.51502721,
  condition: 'none',
  filters:
  [
    { asDimension: false,
      title: 'Category',
      value: 'Include all',
      jaql: [Object],
      panel: 'scope' },
    { asDimension: false,
      title: 'Gender',
      value: 'Include all',
      jaql: [Object],
      panel: 'scope' } ]
}
```

Build Events

```
{
  type: 'build',
  name: 'Success Event',
  measure: null,
  value: 1,
  condition:
  'none',
  filters: null }
```

Name	Type	Description
Type	String	Type of alert. 'kpi' for data events and 'build' for build events.
Name	String	Name of the alert.
Message	String	Notification message.

Name	Type	Description
originDashboardLink	String	Dashboard ID provided by Sisense. You can find the dashboard ID from the URL of the dashboard. For example, the ID of the dashboard below is: http://localhost:8081/app/main#/dashboards/573b4322048e5e382a000001
value	Double	For data events when a threshold condition is applied, this is the value of the threshold. For build events, the value is 0 for failed builds and 1 for successful builds.
condition	String	Type of condition that generated the alert.
filters	Object	Object that defines filters applied to the dashboard.

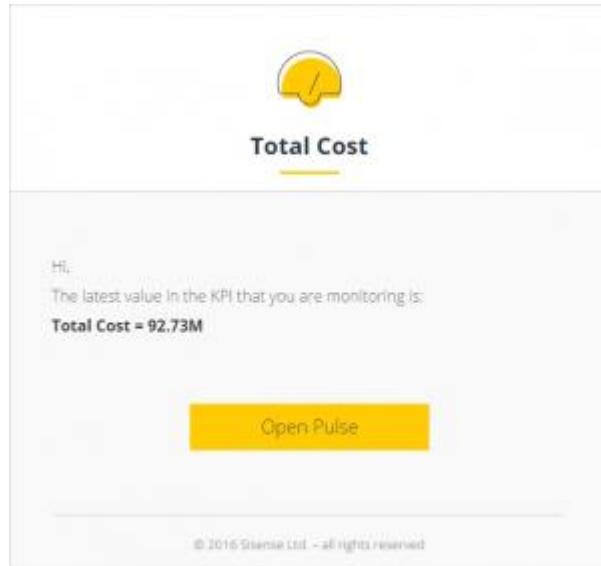
Customizing Notification Messages

By default, the following message is provided in your email alert notifications:

Hi,

The latest value in the KPI that you are monitoring is:

Value



You can provide customized messages for your notifications when configuring your alert through the alert's Advanced Settings.

The message you provide replaces the string: The latest value in the KPI that you are monitoring is:

For example, if your message is "The last value of my KPI is:" the message in your alert's notification will be:

Hi,

The last value of my KPI is:

Value

To customize your notification's text:

- For Build Alerts, select the  tab.

Add to Pulse

Message
Type a message to be sent with the alert...

Notifications

Email
 Mobile
 Slack
 Zapier
[+ Add Webhook](#)

Additional Options

[Basic](#)  

For Data Alerts, select **Advanced Options > **.

Add to Pulse

Name

Condition


Automatic


Threshold


Feed

We will monitor this value for you and let you know when there are irregularities.

- In the Message area, enter your custom text.

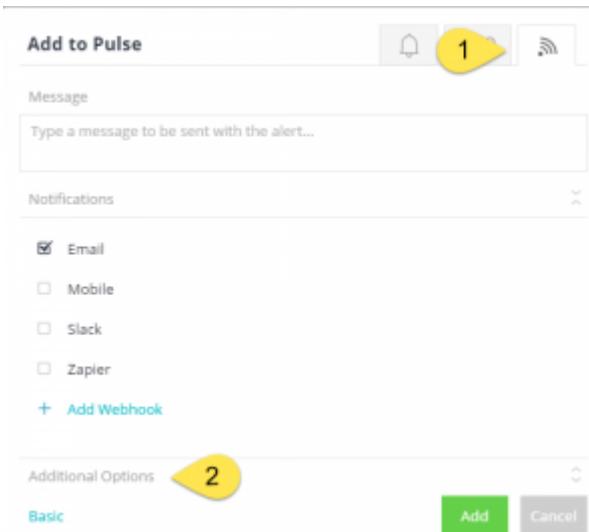
3. Click **Save**. The next time a notification is sent, your custom message is displayed in the notification.
4. You can modify your alert notifications further including customizing the text and images displayed. For more information, see Rebranding Sisense Automated Emails.

Configuring Additional Options

You can configure how often and when a notification is sent after an event is triggered. For example, you can limit the amount of notification sent to one when a data alert is triggered. This is useful if you build your ElastiCubes frequently, but do not need multiple notifications.

To configure additional options:

1. When adding or editing a widget to Pulse, in the Advanced Options window, select  >Additional Options.



2. Select any of the following options:

For data alerts:

Select **Notify only once after a condition is met** to limit how many notifications are sent to notification after an event occurs.

Select **Notify when a condition is no longer met** to notify users after the event is no longer triggered.

For build alerts:

Select **Notify once when there is a status change** to notify users only once when the latest build status is different than the previous build status.

3. Click **Add** to save the alert.

Sisense Pulse

Sisense Pulse can be personalized to your needs so you can stay on top of your most important KPIs across all your dashboards and your latest ElastiCube builds. By adding important KPIs from your dashboards to Sisense Pulse, you can get a comprehensive picture of your data from a single location.

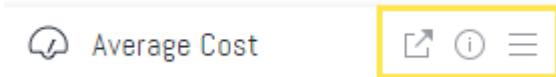
Sisense Pulse contains tiles that display information from your dashboards and the status of your ElastiCube builds.

Within Sisense Pulse, there are two types of tiles, data tiles that display your Indicator widgets' current status and recent history and build tiles that display the latest status of your ElastiCube builds.

You can access Sisense Pulse from the menu bar of the Sisense Web Application.

Home Pulse ElastiCube Manager Admin |  

By hovering over the header of any tile in Sisense Pulse, you can view additional options that provide more information.



The  icon opens a new tab in your browser. For data tiles the dashboard where the related Indicator widget originated is displayed and for build tiles the ElastiCubes page in the Admin area is displayed.

The  icon is a tooltip that describes the tile in more detail. For data tiles the tooltip describes who added the tile to Pulse and when it was added. Build tile tooltips describe when the tile was created, which ElastiCube is being monitored, and under what conditions an alert is triggered when alerts are enabled.

The  icon displays a menu for managing tiles in Sisense Pulse. For more information regarding managing tiles in Sisense Pulse, see [Managing Tiles in Sisense Pulse](#).

Adding Tiles to Sisense Pulse

To keep track of your important KPIs and ElastiCube builds, you add the relevant data and build tiles to Sisense Pulse.

The flow for adding tiles to Sisense Pulse is the same for creating alerts for Indicator widgets and ElastiCube builds. For more information about these flows, see the following topics:

For data tiles, see [Creating Data Alerts](#).

For build tiles, see [Creating Build Alerts](#).

Each of these flows adds tiles to Sisense Pulse and creates an alert. If you want to monitor your data and ElastiCube builds without alerts, see [Deactivating Notifications](#).

Managing Tiles in Sisense Pulse

After you have created alerts or added tiles to Sisense Pulse, you can manage those tiles from Sisense Pulse. Through the Sisense Pulse page, you can duplicate alerts and modify them, disable notifications, and delete alerts.

Duplicating Tiles in Sisense Pulse

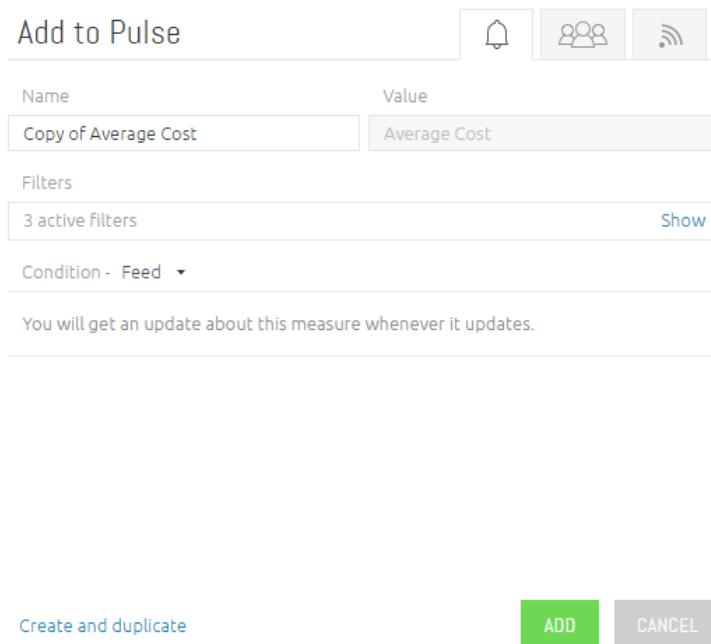
You can create copies of alerts and modify them to quickly generate new alerts with different notification messages or notify various groups across separate channels.

For example, if you want to notify Group 1 via email and Group 2 via Slack when the same event occurs, you can duplicate an alert and modify the notification channel.

To duplicate a tile:

1. In the Sisense Web Application, select **Pulse**.

2. In the relevant alert's menu, select **Duplicate**. A copy of the alert is created and the Add to Pulse window is displayed.



The screenshot shows the 'Add to Pulse' configuration window. At the top, there are three icons: a bell for notifications, two people for sharing, and a Wi-Fi signal for connectivity. Below these are sections for 'Name' (containing 'Copy of Average Cost') and 'Value' (containing 'Average Cost'). Under 'Filters', it says '3 active filters' and has a 'Show' link. A dropdown menu shows 'Condition - Feed'. A note at the bottom states: 'You will get an update about this measure whenever it updates.' At the bottom right are 'Create and duplicate' (blue), 'ADD' (green), and 'CANCEL' (grey) buttons.

3. You can modify the alert's settings from the Add to Pulse window.

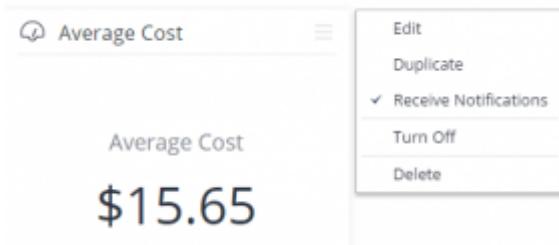
Deactivating Your Notifications

Each time you add a tile to Sisense Pulse or create an alert, by default, you will receive notifications when the event is triggered. If you want to just monitor KPIs from Sisense Pulse, or you do not want to receive any notifications, you can disable notifications from being sent by toggling the notifications. When you toggle the notifications, you no longer receive notifications when an alert is triggered, however, other users with whom the alert was shared will continue to be notified. See Deactivating Notifications for All Users to turn off notifications for everyone associated with an alert.

To deactivate notifications:

1. In the Sisense Web Application, select **Pulse**.

From the Tile menu of the relevant alert, toggle **Receive Notifications**.



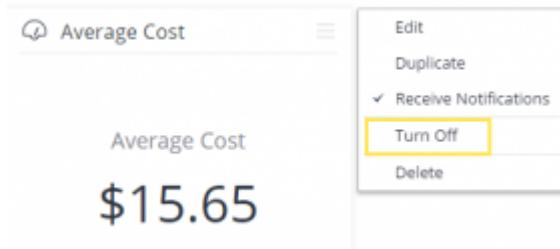
Deactivating Notifications for All Users

If you have added an alert that is no longer relevant or you want to deactivate the alert for all users, you can turn off the alert.

When you turn off an alert, no users are notified when the event that triggers the alert occurs.

To deactivate notifications for all users:

1. In the Sisense Web Application, select **Pulse**.
2. From the Tile menu of the relevant alert, select **Turn Off**. A confirmation dialog box is displayed.



3. Click **Yes** to confirm that you want to turn off all notifications for the alert. The alert is turned off.

4. Alerts that have been turned off appear in the Pulse page as shaded tiles. You can reactivate the alert by selecting Turn On from the Tile menu.

Deleting Tiles in Sisense Pulse

When a tile is no longer needed, you can delete the tile through the Sisense Pulse page.

Deleting an alert removes it from the Sisense Pulse page and users are no longer notified.

To delete an alert:

1. In the Sisense Web Application, select **Pulse**.
2. In the relevant alert's menu, select **Delete**.

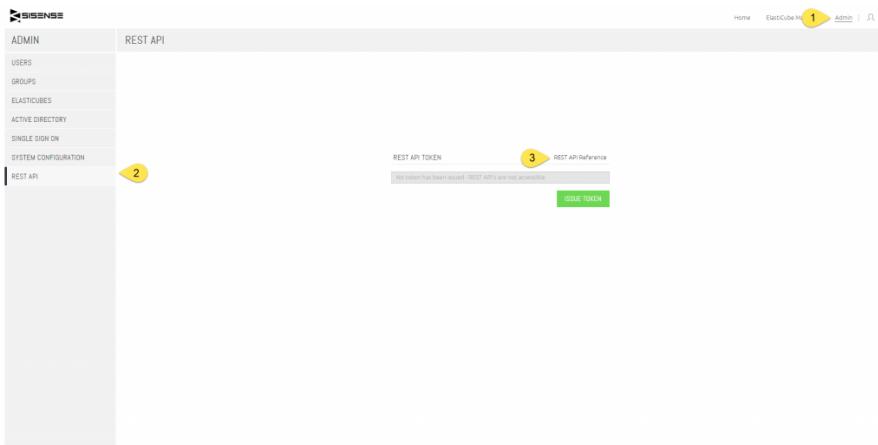
Removing Sisense Pulse

You can remove the Sisense Pulse through the Sisense REST API.

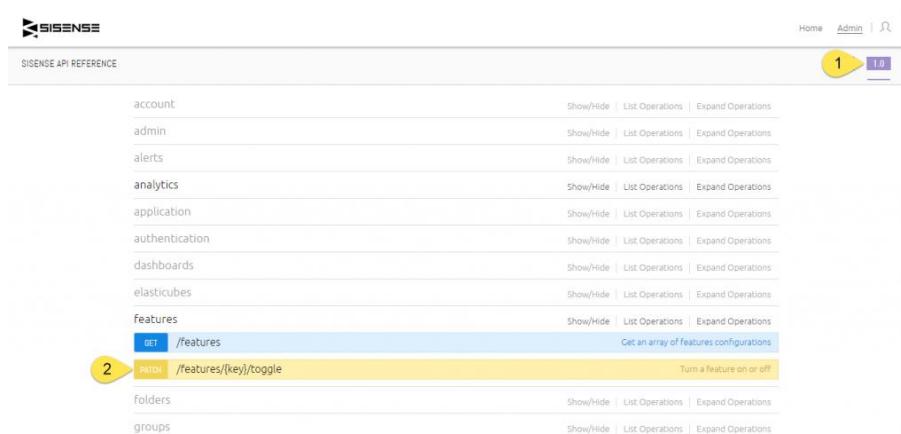
Removing Sisense Pulse removes the Pulse page from the Sisense Web Application and deactivates all the alerts defined in Sisense Pulse.

1. To remove Sisense Pulse:

2. Access the Sisense REST API.



3. In version 1.0 of the REST API, select the PATCH Dashboard endpoint, /features/{key}/toggle.



4. In the Authorization field, enter your Sisense authorization key. The value of this field is the user's API token, preceded by the keyword Bearer (with a space between it and the token). For information about how to retrieve this key, see Using the REST API.

5. In the key field, enter the name of the feature to be enabled or disabled. In the case of Sisense Pulse, the key is alert. You can retrieve the available features that you can toggle from the GET /features endpoint.

6. In the **toggleFeature** field, set the active key value as true or false to enable (true) or disable (false) a feature.
7. Click **RUN** to toggle the alert.

Administration

Managing Dashboard Users

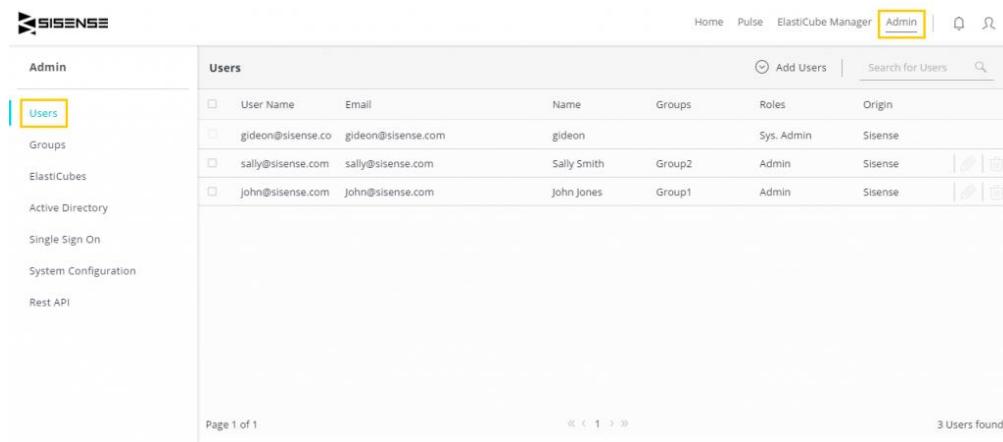
This section describes how to add and manage Sisense users that can access Sisense dashboards.

You can add users using the following methods:

- ▶ Add new users by entering them one by one or in bulk.
- ▶ Add new users from your Active Directory database.

To access the user management settings:

- ▶ Click **ADMIN** in the top-right corner and select the **USERS** tab on the left. By default, the only available user after installing Sisense is the user that installed Sisense. By default, this user is an Admin.



User Name	Email	Name	Groups	Roles	Origin
gideon@sisense.co	gideon@sisense.com	gideon	Sys. Admin	Sisense	
sally@sisense.com	sally@sisense.com	Sally Smith	Group2	Admin	Sisense
john@sisense.com	John@sisense.com	John Jones	Group1	Admin	Sisense

Adding a New User

To add a new user:

1. Click the **ADD USERS** button. If a connection with Active Directory also exists, an additional **ADD ACTIVE DIRECTORY**

USERS option is available. Click it to select a user from Active Directory.

Add Users

Single User
Import Many

Email	<input type="text"/>
First Name	<input type="text" value="Optional"/>
Last Name	<input type="text" value="Optional"/>
Language	<input type="text" value="Group/System Langua..."/> ▼
Roles	<input checked="" type="radio"/> Viewer <input type="radio"/> Designer <input type="radio"/> Admin

Define Password
OFF ON

* The user will receive an email containing a password activation link.

Cancel

2. Fill in the user's details in the window.
3. Select the **Role** of the user:
 - ▶ **Sisense Viewers** can view, explore, drill down, make selections and filter the Dashboards that Designers share with them. Viewers only need a standard web browser; no plugins or downloads. They cannot create new Dashboards or edit existing ones.

- ▶ **Sisense Designers** create, design, edit and share Dashboards. Sisense Designers determine whether the user with who they share a Dashboard has editing rights (is a Designer) or only viewing rights (Viewer).
 - ▶ **Sisense Admin:** Admins have all the rights of Designers and Viewers. In addition, only admin users can access the MANAGE section which enables them to manage Elasticubes, users, user groups and more. Sisense supports multiple Admin users.
4. Click **SAVE**. The user then receives an email from Sisense that redirects the user to activate his/her account and set a password.

To manually set a password for the new user:

Set the **Define Password** option to **ON** and enter a password before saving.

Define Password	OFF ON
Set Password	<input type="password"/>
Confirm Password	<input type="password"/>
<small>* An email will not be sent. Please notify the user of his account and password.</small>	
<input type="button" value="CANCEL"/>	

In this case, the user receives an invitation email. You must personally notify the user of the password that you set for him/her.

Adding Multiple Users

To add multiple users:

1. Click the **ADD MANY** option at the top right of the Add users window.

In the displayed text box, paste a comma separated list of emails and click **NEXT**.

The next screen enables you to assign Viewer or Designer rights to the users that you entered and to verify that the user count and permissions match your license.

2. After you click **ADD**, each user receives an activation email.

Editing an Existing User

To edit an existing user:

1. Click **ADMIN** in the upper right corner and select the **USERS** tab on the left.



2. Click the **Pencil** button on the right of the user's entry in the list.

Deleting a User

To delete an existing user:

1. Click **ADMIN** in the upper right corner and select the **USERS** tab on the left.



2. Click the **Trash** button at the right of the user's entry in the list.

Managing User Groups

This section describes how to add and use user groups in Sisense.

How Do User Groups Work?

- ▶ When building and sharing dashboards, wherever you can select a user, you can also select a user group. For example, you can share a dashboard with an entire user group at once.
- ▶ When you add an additional user to an existing user group, then that user is automatically awarded the permissions of that user group.
- ▶ When users are added in Active Directory to a previously imported Active Directory user group, then those users are automatically added to the corresponding Sisense user group.
- ▶ User groups can only contain users. They do not support nested user groups.
- ▶ Additional users cannot be added to a user group that was imported from Active Directory.

You can add users to a user group using the following methods:

- ▶ Add existing Sisense users to a user group
- ▶ Add a user group from Active Directory

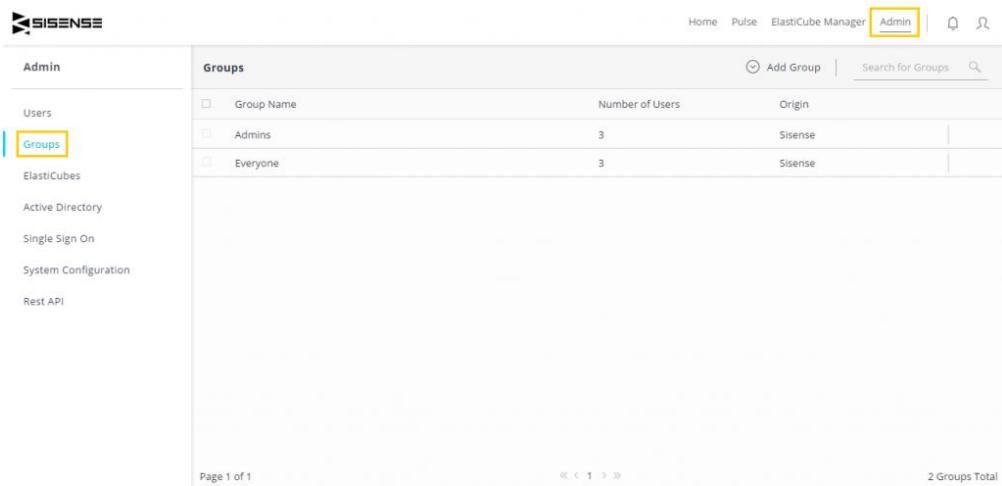
Accessing User Group Management

You can add, edit and delete user groups, and assign users to them.

To access the user group settings:

- ▶ Click **ADMIN** in the top-right corner and select the **GROUPS** tab on the left. A list of the previously defined user groups is displayed. The **Origin** column displays **Sisense** or **Active**

Directory to indicate whether the user group was defined in Sisense or imported from Active Directory.



Group Name	Number of Users	Origin
Admins	3	Sisense
Everyone	3	Sisense

If no user groups have yet been defined, the following message is displayed: *No Groups Found*

Adding a New User Group

The following procedure describes how to add a new user group by selecting users from existing users in Sisense.

See Adding a User Group from Active Directory for a description of how to add a user group from Active Directory.

To add a new user group:

1. Click **ADMIN** in the top-right corner and select the **GROUPS** tab on the left.
2. Click **ADD GROUP**.

Create a New Group

Group Name

Group Users

No members added yet

Default Language:

Choose default members role: Viewer Designer

3. In a **Group Name** field, enter the name for this user group.
4. In the **Group Users** field select each of the users that belong to this group. To add each user, start typing in the user's name. An

auto-complete list is displayed showing the users that already exist in Sisense. Select a user name to add to the group.

Note: If you want, you can leave the group empty and add members to it later.

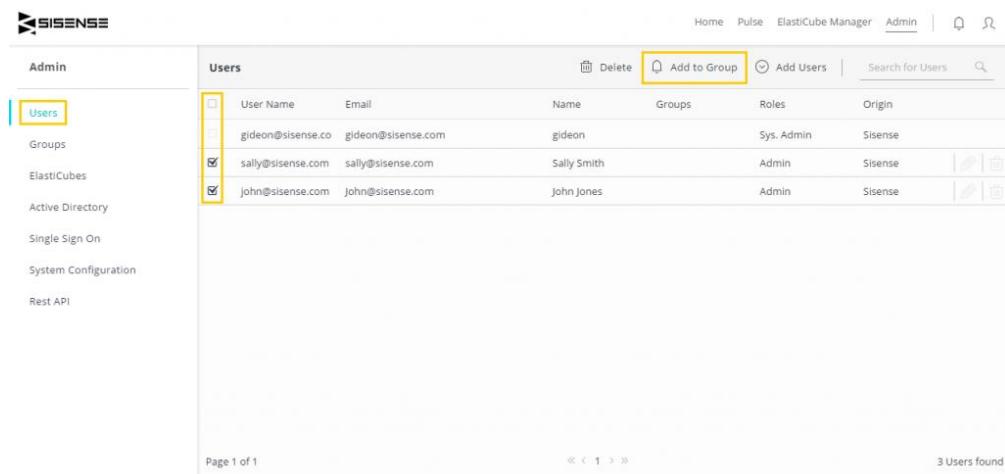
3. Click **Save**.

Adding Users to an Existing User Group

The following procedure describes how to add one or more users to an existing user group.

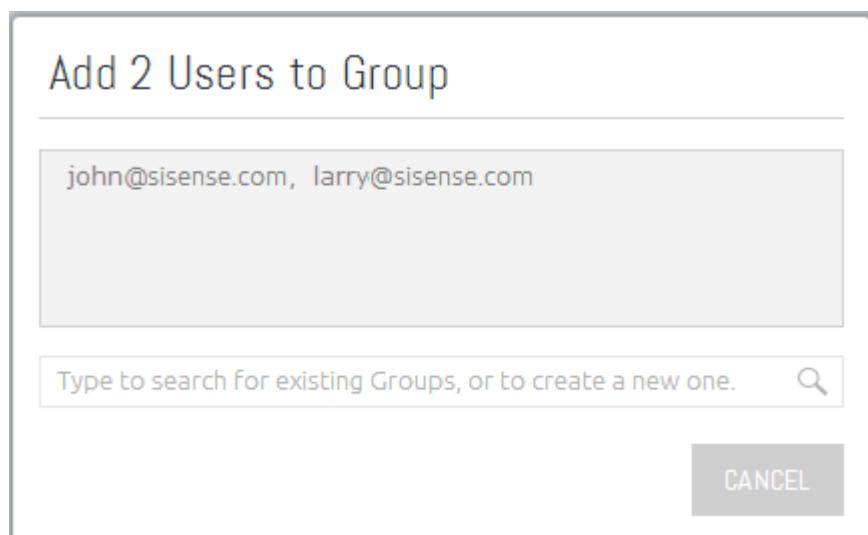
To add multiple users to a user group:

1. Click **ADMIN** in the top-right corner and select the **USERS** tab on the left.
2. Select the checkboxes next to the users' names for the users that you want to add to the user group.



User Name	Email	Name	Groups	Roles	Origin
gideon@sisense.co	gideon@sisense.com	gideon		Sys. Admin	Sisense
<input checked="" type="checkbox"/> sally@sisense.com	sally@sisense.com	Sally Smith		Admin	Sisense
<input checked="" type="checkbox"/> john@sisense.com	john@sisense.com	John Jones		Admin	Sisense

3. Click **ADD TO GROUP**.



Add 2 Users to Group

john@sisense.com, larry@sisense.com

Type to search for existing Groups, or to create a new one.

CANCEL

4. Type in the field at the bottom of the window to select an existing user group or to create a new one.
5. Click **ADD**.

Editing an Existing User Group

To edit an existing user group:

1. Click **ADMIN** in the upper right corner and select the **GROUPS** tab on the left.

2. Click the **Pencil**  button on the right of the user group's entry in the list.

Note: To remove a user from the Group, hover over the user name in

the list and click the **Trash**  button.

Members

Members	
+ Add members	
barbara@sisense.com	

Deleting a User Group

When a user group that was entered in Sisense is deleted, its users are not deleted. They remain active users in Sisense.

When a user group that was imported from Active Directory is deleted, a prompt is displayed asking whether you want to also delete all the users in this user group from Sisense.

To delete an existing user group:

1. Click **ADMIN** in the upper right corner and select the **GROUPS** tab on the left.

2. Click the **Trash**  button at the right of the user group's entry in the list.

Managing User Sessions

When a user signs into the Sisense Web Application, a session cookie is stored in their browser. The session cookie allows the user to remain logged in and authenticated even after ending a session by closing the browser.

For security reasons, you may want to logout your users or end their session when they close the browser. Through the Sisense REST API, you can add an additional property, sessionOnlyCookie, to the POST, /api/settings/security of REST API v.9.

When the Boolean value of sessionOnlyCookie is **true**, the login page does not display the "Remember Me" checkbox in the user's browser and logging in to Sisense generates a session cookie.



When the user logs out or closes their browser, the session ends and they must log in again to access Sisense.

The following is an example of a cURL command in which the Remember Me button is no longer displayed in the Login page.

```
<pre>curl -X POST -header "Content-Type: application/json" -  
header "Accept: application/json" -d "{  
\"sessionOnlyCookie\": \"true\"  
}" "http://localhost:8081/api/settings/security"</pre>
```

Customizing User Roles

This section describes how to customize the permissions given to the Sisense user roles (designer and viewer), using Sisense's [REST API](#). At the bottom of the page, you can find a video tutorial demonstrating the following procedure.

A reminder about default roles

Sisense users are assigned one of the following default roles:

- ▶ Administrators
- ▶ Designers: Can create, design, edit and share dashboards.
- ▶ Viewers: Can view, explore, drill down, and filter the dashboards that designers have shared with them.

Note: The API uses a previous terminology for Designers and Viewers. When entering the role name into the operation parameters, use **Contributor** (for Designer), and **Consumer** (for Viewer).

Customizing roles

You can customize the three default roles with Sisense's REST API, by defining which permissions to enable or disable for each role. For example, you can prevent a user from drilling into widgets, or exporting data to a csv file. On the other hand, you may want to let viewers change the widget type and format the style.

The following table lists all the actions that you can customize using the API. *True* and *False* indicate values that you can change. *Never* indicates values that you cannot change.

Dashboards

Dashboards	Admin	Designer	Viewer	Description
create	True	True	Never	Create a new dashboard.
duplicate	True	True	False	Duplicate an existing dashboard.
toggle_edit_mode	True	True	False	Select between Layout and View modes in the dashboard. Layout mode is the default mode in which you can add new widgets and rearrange them on the dashboard. In View mode, the dashboard cannot be rearranged or edited.
edit_script	True	True	Never	Edit a dashboard using the Sisense JavaScript API.
export_dash	True	True	Never	Export a dashboard as a .dash file that can be imported back into Sisense.

Dashboards	Admin	Designer	Viewer	Description
export_jpg	True	True	True	Download a widget as an image in PNG format. The downloaded image represents the current view.
export_pdf	True	True	True	Export the dashboard to PDF format. The exported file includes the current view.
restore	True	True	True	Revert changes made in the dashboard to the latest dashboard that was shared.
import	True	True	Never	Import a .dash file into the Sisense Web Application.
select_palette	True	True	Never	Change the selected color palette.
Filters				
create	True	True	False	Create a new filter.
on_off	True	True	True	Switch filters on or off.

Dashboards	Admin	Designer	Viewer	Description
toggle_expansion	True	True	True	Enable the expansion of filter settings to see more filter parameters.
modify_type	True	True	False	Include in the filter options 'Ranking' and 'Starred', in addition to 'List' and 'Text'.

Widgets

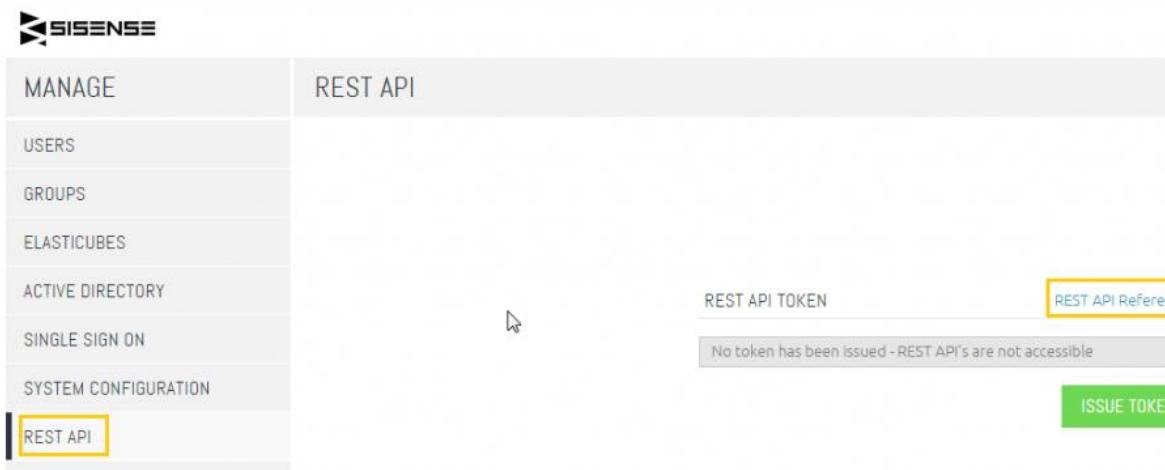
Widgets	Admin	Designer	Viewer	Description
create	True	True	Never	Create a new widget.
delete	True	True	Never	Delete a widget.
rename	True	True	False	Rename a widget.
duplicate	True	True	Never	Duplicate a widget.
edit_script	True	True	Never	Edit a widget using the JavaScript API.
change_type	True	True	False	Change the chart type.
export_csv	True	True	True	Export a widget's data to a CSV file.

Widgets	Admin	Designer	Viewer	Description
export_png	True	True	True	Download a widget as an image in the PNG format. The downloaded image represents the current view.
modify_selection_mode	True	True	False	Enable the option to change the filter behavior between 'Slice' (only filtered items are included in the chart) and 'Highlight' (filters items are highlighted).
drill_to_anywhere	True	True	True	Enable the option to drill down into a field to get an in-depth view of a selected value. If False, the user can only drill down with predefined drill hierarchies (if defined).
Items				
reorder	True	True	False	Reorder the fields and values in the data panel and their representation in the chart.

Widgets	Admin	Designer	Viewer	Description
modify_type	True	True	Never	Enables the option to edit values and categories in the data panel.
on_off	True	True	True	Enable the option to turn fields on or off in the widget.
Filters				
on_off	True	True	False	Switch filters on or off.
toggle_expansion	True	True	True	Enable the expansion of filter settings to display more filter parameters.
modify_type	True	True	False	Include in the filter options 'Ranking' and 'Starred', in addition to 'List' and 'Text'.
use_starred	True	True	False	Enables the option to star (bookmark as favorite) a filter selection.

1. To change permissions for a user role:

2. In the Sisense web app, click **ADMIN** at the top right of the screen, and then REST API in the left menu.
3. Click REST API Reference to view the API documentation.



4. Click on **/roles** to access the different API operations.
5. Find the operation that you require (see examples below), and type in the required parameters in the operation. You can click on **Model Schema** to see and add example code for the parameters.

When changing a permission for a role, you must use the **Path** parameter, which defines the permission you want to update and its relative path in the hierarchy that appears in the table above. For example, to change the 'Create' setting for dashboard filters, enter `dashboards/filters/`. For creating dashboards, enter `/dashboards`. See the reference table above

for more settings.

Dashboards	Admin	Designer
create	true	true
duplicate	true	true
toggle_edit_mode	true	true
edit_script	true	true
export_dash	true	true
export_jpeg	true	true
export_pdf	true	true
restore	true	true
import	true	true
select_palette	true	true
Filters		
create	true	true
on_if	true	true
tr_expansion	true	true
path	dashboards/filters/	
mount-type		

6. Click **RUN** to apply the changes.
7. Here are some examples of what you can do.

See all permissions for all roles

Method: **Get /roles**

Parameters

- ▶ **includeManifest** –Use the default setting (true)
- ▶ **compiledRoles** –Use the default setting (true)

Example API Call

/api/roles?includeManifest=true&compiledRoles=true

See all settings for a specific role

Method: **GET /roles/{idOrName}**

Parameters

- ▶ **idOrName** –The role name can be one of the following: admin, contributor, consumer.
- ▶ **compiledRole** –Use the default setting (true)

Example API Call

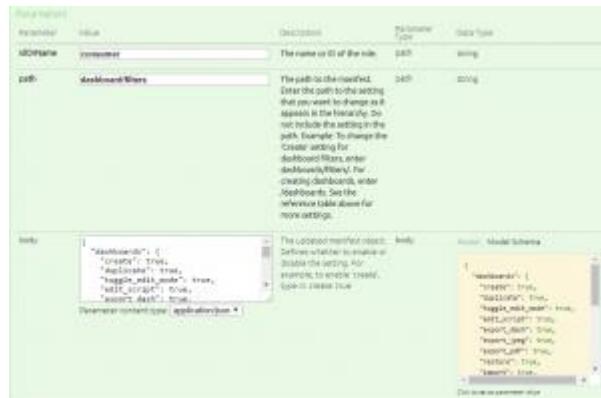
/api/roles/consumer?compiledRole=true

Change a specific setting for a specific role

Method: **PUT /roles/{idOrName}/manifest/{path}**

Parameters

- ▶ **idOrName** –The role name can be one of the following: admin, contributor, consumer.
- ▶ **Path** –Enter the path to the setting that you want to change as it appears in the hierarchy. Do not include the setting in the path. Example: To change the 'Create' setting for dashboard filters, enter *dashboards/filters/*. For creating dashboards, enter */dashboards*. See the reference table above for more settings.
- ▶ **Manifest** –Defines whether to enable or disable the setting. For example, to enable 'create', type in: *{"create": true}*



Example API Call

/api/roles/viewer/manifest/dashboards/filters

Note: You can use POST to do the same action, but for POST all other options will be reset to their default settings.

Restore a specific setting for a specific role

Method: **DELETE /roles/{idOrName}/manifest/{path}**

Parameters

- ▶ **idOrName** –The role name can be one of the following: admin, contributor, consumer.
- ▶ **Path** –Enter the full path of the setting you want to change, as it appears in the hierarchy. Example: dashboards/filters/modify_type. See the reference table above for more settings.

Example API Call

/api/roles/viewer/manifest/dashboards/filters/modify_type

Restore all settings for a specific role

Method: **DELETE /roles/{idOrName}/manifest/{path}**

Parameters

- ▶ **idOrName** –The role name can be one of the following: admin, contributor, consumer.
- ▶ **Path** –/Enter just a slash to restore all settings.

Example API Call

/api/roles/viewer/manifest/

The following video tutorial demonstrates the above procedure.

Integrating Active Directory

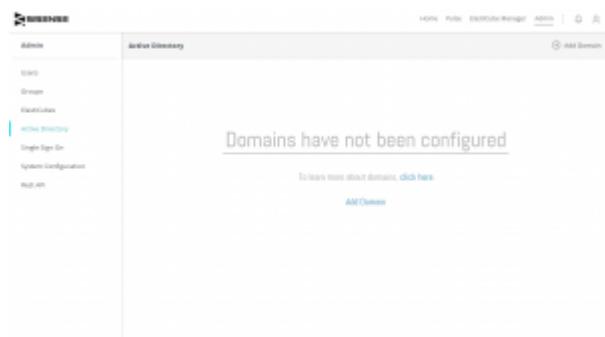
Sisense supports integrating Active Directory users and groups with your current Sisense users, so that you can share dashboards and email reports with any of your users. This works in parallel with Sisense's existing User Management, so you can continue to add users and user groups that are not part of the Active Directory domain.

To enable Sisense to identify your Active Directory users, you import your Active Directory users and groups into Sisense. When you import an Active Directory user, Sisense automatically creates a Sisense user. When you import an Active Directory group, Sisense does not automatically create a Sisense user for each member. The first time a member of an Active Directory group logs into Sisense, then Sisense automatically creates a Sisense user. Active Directory users that do not log in, are not assigned a Sisense user until their first login.

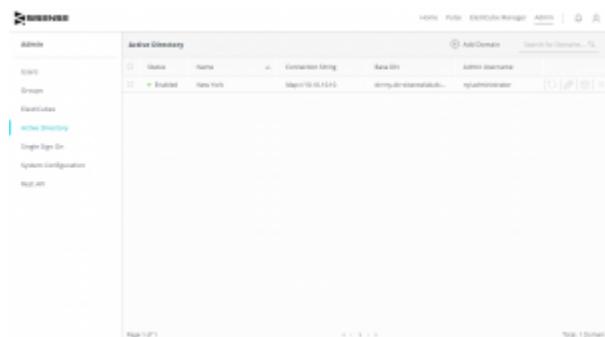
This section describes how to configure Active Directory in Sisense, and how to import users and user groups from Active Directory.

Adding an Active Directory Domain to Sisense

From the Active Directory page, you can add Active Directory domains to your Sisense configuration. Each Active Directory domain you add to your Sisense configuration is added to the Active Directory list displayed in the Active Directory page. The first time you open the Active Directory page, the Add Domain link is displayed, which you can click to begin adding Active Directory domains.



After you have added your first Active Directory domain, it is displayed in the Active Directory list.

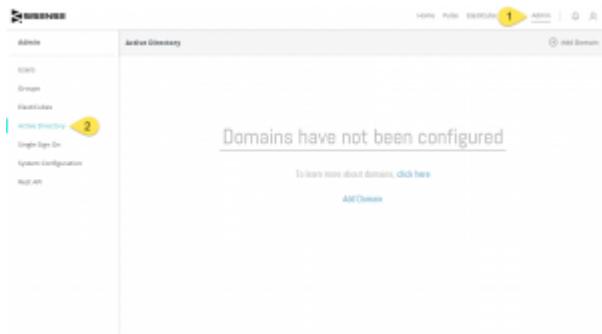


From the Active Directory list, you can add additional Active Directory domains. Each Active Directory domain you add to Sisense enables you to share your dashboards and reports with users from any of the domains as well as Sisense users not part of an Active Directory.

You can filter the list of the Active Directory domains by entering the name of the relevant domain in the Search field.

To add an Active Directory:

1. Click **ADMIN** in the upper right corner and select the **ACTIVE DIRECTORY** tab on the left.



2. In the Active Directory page, click **Add Domain**.
3. In the Add Domain window, fill in the Active Directory configuration fields to point to your Active Directory instance:
 - ▶ **Name:** The name of your Active Directory domain. This name is used to locate your Active Directory domain in Sisense when you have multiple domains.
 - ▶ **Connection String:** The full path to your Active Directory domain. Example: **ldap://dc.domain.com**
 - ▶ **Base DN:** The Active Directory location from where a server searches for users. Enter the domain by its distinguished name (DN) with its domain components (DC) connected by commas. For example: For the domain *domain.sisense.test*, use **DC=domain,DC=sisense,DC=test**.

Note: If you have a hierarchy of user groups in your organization, you can define the hierarchy in your string as follows:
If your domain is *domain.sisense.com/OU1/OU2*, then use **OU=OU2, OU=OU1,DC=domain,DC=sisense,DC=com**. Only users and user groups from OU3 will be available. Users/groups under OU1 will not be available.

- ▶ **Username:** Enter the username with domain.
Example: **domain\username**
 - ▶ **Password:** Enter the password for your Active Directory instance.
4. Click **TEST** to verify that your configuration successfully connects to your Active Directory domain.
 5. In the **Sync Data** field, select the frequency that Sisense automatically synchronizes with the users and groups in this Active Directory domain.
 6. Click **SAVE**.

Editing Active Directory Domain Settings

To edit an Active Directory domain:

1. From the Active Directory list, select  for the Active Directory settings you want to edit. The Edit Active Directory window is displayed.
2. In the Edit Active Directory window, edit the relevant fields.
See **Adding an Active Directory in Sisense** for more information about each field.
3. Click **Save**.

Disabling an Active Directory Domain

Supports allows you to disable Active Directory domains in Sisense.

An Active Directory domain that has been disabled is still displayed in the Active Directory list, however, users from this domain will not be able to log in to Sisense, and you will not be able to share dashboards or reports with them.

Disabled Active Directory domains have the status Disabled in the Status column in the Active Directory list.

To disable an Active Directory domain:

- ▶ From the Active Directory list, select  > **Disable** for the Active Directory domain you want to disable. The Active Directory domain is disabled.
You can enable the Active Directory domain by selecting  > **Enable**.

Deleting an Active Directory

You can delete an Active Directory domain from your configuration.

When you delete an Active Directory domain, the users included in that domain are no longer displayed when sharing dashboards or reports. Deleted Active Directory domains cannot be restored, however you can add the Active Directory domain again from the Active Directory page.

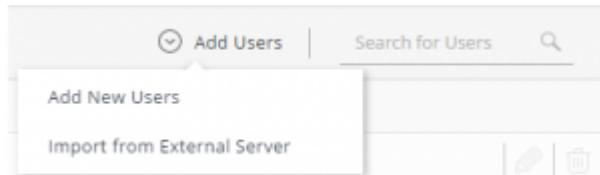
To delete an Active Directory domain:

1. From the Active Directory list, select  for the Active Directory settings you want to delete. The Delete Active Directory window is displayed.
2. Click **Delete**. The Active Directory domain settings are removed from Sisense.

Adding a User from Active Directory

Once a connection with Active Directory has been established, an additional **Import from External Server** option is added to the **ADD USERS** button in the **USER** tab.

The user's username and email will be added from Active Directory.

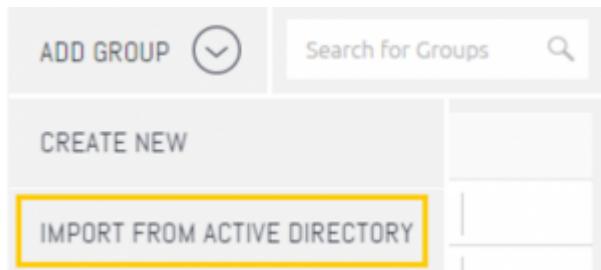


Note: You can add a user from Active Directory, even if you have not imported that user's user group from Active Directory.

See [Managing Dashboard Users](#) for general information about working with users/user groups in Sisense.

Adding a User Group from Active Directory

Once a connection with Active Directory has been established, an additional **IMPORT FROM ACTIVE DIRECTORY** option is added to the **ADD GROUP** button in the **GROUPS** tab.



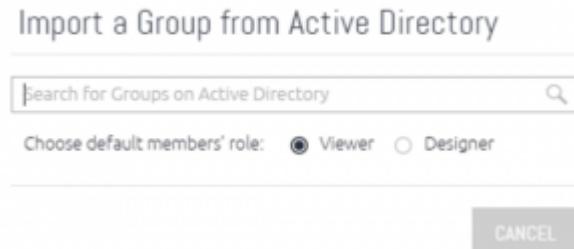
The following procedure describes how to import a list of users from a group in Active Directory to Sisense.

Users that are added in this way only appear in the Sisense users list after each user logs in to Sisense. In this way, your Sisense license slots are only consumed by users who actually log in to Sisense.

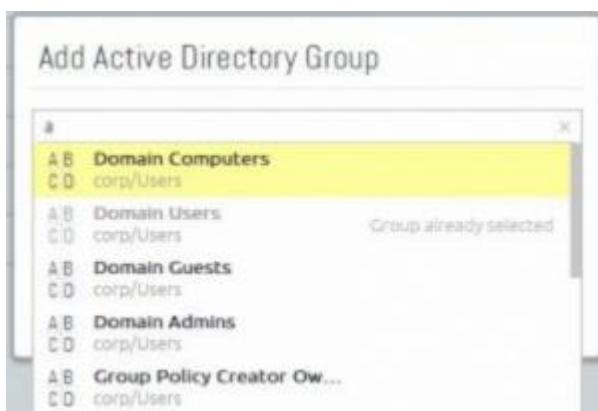
To add a user group from Active Directory:

1. Click **ADMIN** in the upper right corner and select the **GROUPS** tab on the left.

2. Click the **ADD GROUP** button and then select the **FROM ACTIVE DIRECTORY** option from the menu.



3. In the **Search Active Directory Group** field start typing in the name of the desired Active Directory Group. An auto-complete dropdown menu is displayed from which you can select a user group.



If an Active Directory user group has already been imported into Sisense, the words **Group already selected** appear next to it, as shown above. You can add multiple Active Directory groups, by selecting one group after another. The groups appear in this window as follows:

4. Select the default Role for the members of this user group: Viewer or Designer. It will be assigned to new users from this group, when the users are created. Subsequently, user's role can be changed in the Sisense Users list.



SISENSE

Changes to a group's default role will affect users created after that time, but will not affect existing users.

5. Click **ADD**.

Introduction to SSO

Single Sign-On (SSO) is a mechanism that allows a system to authenticate users and subsequently tell Sisense that the user has been authenticated. The user is then allowed to access Sisense without being prompted to enter separate login credentials.

The SSO security mechanism allows Sisense to trust the login requests it gets from your corporate authentication system, and grant access to the users that have been authenticated by it. An SSO session begins when the authenticated user requests a secured resource from Sisense while logged into your site or application. The user's browser sends an HTTP request to Sisense that includes a cookie which contains session and authentication information. This information is then used for session validation.

Users who already have Sisense accounts can continue to access Sisense through the Sisense Login page with their current accounts. To prevent users from directly logging in to Sisense instead of your login page, your Sisense administrator can change the passwords of your current users forcing them to log in with your company's credentials in your company's login page.

Sisense recommends that administrators always keep a Sisense password, so that the administrator can access Sisense in case the SSO server is not available.

Sisense SSO supports two SSO protocols for securing the exchange of user authentication data , JSON Web Token (JWT) and SSO via SAML 2.0 (Security Assertion Markup Language). For more information, see SSO via JWT or SSO via SAML 2.0.

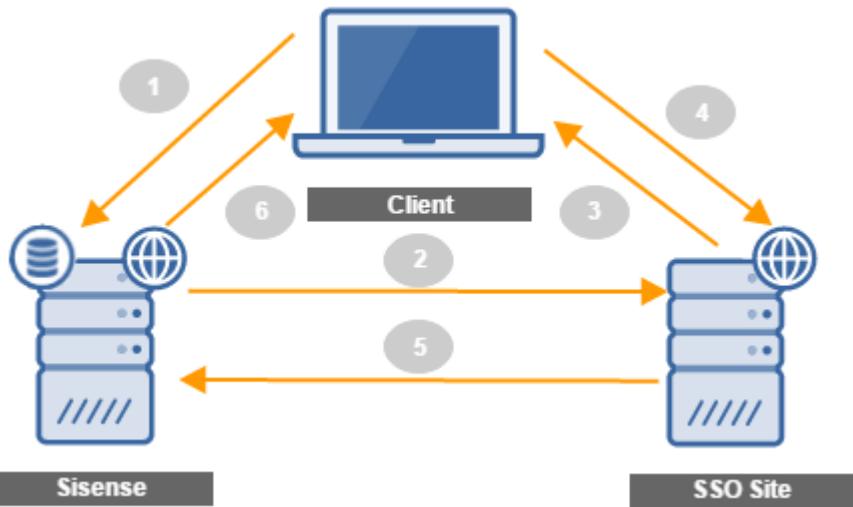
SSO via JWT

JWT is a technique that can be used for single sign-on (SSO) between your site and Sisense. JWT is a token that represents your users credentials wrapped in a single query string. In addition, Sisense uses the `jti` parameter (see below), which adds a unique ID to the token that prevents the token from being used more than once, thus preventing attacks on the system (also known as replay attacks).

The Sisense SSO via JWT authentication flow is explained in the following diagram:

SSO Authentication Flow

The following is a diagram of the SSO authentication flow from your site or application to Sisense.



1. Your user requests a resource from Sisense, typically a dashboard.
2. Sisense recognizes that no authenticated cookie is present. If you have enabled SSO in Sisense, the SSO handler redirects

the user to your Remote Login URL defined in the Sisense Web Application.

Single Sign On Configuration

Method	<input checked="" type="radio"/> JWT <input type="radio"/> SAML 2.0	i
Remote Login URL	<input type="text" value="http://dash.company.com/sisense/sso"/>	i
Remote Logout URL	<input type="text" value="http://dash.company.com/sisense/ssout"/>	i
Shared Secret	<input type="text" value="86345d077c5e87933582ccc0af769b9e5240df7ec9b6091acec364944bc09891"/>	i
Save		

3. Your user is challenged to authenticate their account.
4. Your Remote Login application authenticates your user and generates a JWT (JSON Web Token).
5. You redirect the user back to Sisense with the encoded JWT in a query string. Sisense sets a cookie that authenticates the user's session until they end it or you log them out via the Sisense REST API. For more information see [Logging Users Out](#).
6. Sisense provides the authenticated user with the request resource.

A common scenario that illustrates SSO is when an unauthenticated user navigates to your site in which Sisense is embedded via an iFrame. Sisense redirects this user to your SSO script. Your script authenticates the user through your login process and builds a JWT request with all the relevant credentials wrapped together. You then redirect the customer back to Sisense with the JWT payload. Sisense then decodes the user details from the JWT payload and then grants the user a session.

Configuring SSO in Sisense

While SSO is highly customizable, there are generally four steps you should complete when configuring SSO:

Note: Configuring SSO requires technical expertise and should be conducted by an administrator or developer with SSO experience.

- ▶ **Enabling SSO in Sisense:** Through the Sisense Web Application, an administrator can enable SSO in Sisense and define the relevant Login and Logout URLs.
- ▶ **Creating a JWT:** After you authenticate a user, you generate a JWT with the user's credentials to Sisense, so Sisense knows this user is allowed to access resources from Sisense through your site.
- ▶ **Configure Sisense as a sub-domain:** When authenticating users, you should configure SSO as a sub-domain.
- ▶ **Logging Users Out:** A user can access Sisense so long as a session is maintained. To end a session, the user's cookie that Sisense provides must be deleted. To delete this cookie, you can use the Sisense REST API.

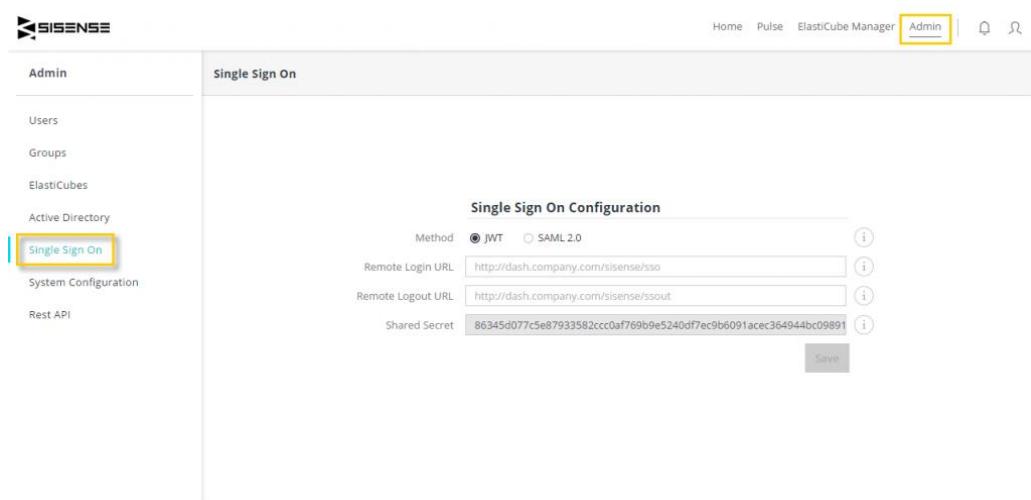
Enabling SSO in Sisense

For Sisense to recognize that your users should be authenticated through SSO, you must enable SSO in the Sisense Web Application. In the SSO menu of the Admin page of the Sisense Web Application, you define the URL where Sisense redirects users to authenticate on your side and where Sisense redirects users after they log out from Sisense.

When you access the SSO menu of the Admin page, Sisense displays the Shared Secret key. The Shared Key is a JWT encryption public key used to encrypt the JWT payload. It is generated once when the SSO configuration is saved. You include this key in the JWT payload when redirecting the user back to Sisense after authenticating them on your side.

To access and set up SSO:

1. Log into Sisense, and click **ADMIN** in the top-right corner of the screen. Click **SINGLE SIGN ON** in the left menu.



The screenshot shows the Sisense Admin interface. The top navigation bar includes Home, Pulse, ElastiCube Manager, Admin (which is highlighted with a yellow box), and search icons. The main menu on the left lists Admin, Users, Groups, ElastiCubes, Active Directory, Single Sign On (which is highlighted with a yellow box), System Configuration, and Rest API. The right panel is titled "Single Sign On Configuration". It contains fields for Method (radio buttons for JWT and SAML 2.0, with JWT selected), Remote Login URL (http://dash.company.com/sisense/sso), Remote Logout URL (http://dash.company.com/sisense/ssout), and Shared Secret (a long hex string: 86345d077c5e87933582ccc0af769b9e5240df7ec9b6091acec364944bc09891). A "Save" button is at the bottom right.

2. Fill in the following SSO configuration fields:
 - ▶ **Remote Login URL:** This is the URL that Sisense will invoke to attempt remote authentication. In that endpoint the participating application user authentication script is triggered and the JWT payload is generated.
 - ▶ **Remote Logout URL:** This is the URL that users will be redirected to after they log out from Sisense (i.e. the participating application's home page).
3. Click **SAVE**.

Creating a JWT

Your script builds a JWT request that contains the user data.

The table below provides a list and descriptions of the attributes your JWT should contain.

In addition, several samples are provided below in various languages.

Attribute	Mandatory	Description
iat	Yes	<p>Issued at the time the token was generated. This is used to help ensure that a given token gets used shortly after it is generated. The value must be the number of seconds since UNIX epoch. Sisense allows up to five minutes clock skew.</p> <p>Note: The date must be an integer and not a float.</p>
sub	Yes	Email of the user being signed in, used to uniquely identify the user in Sisense. If the user does not exist in Sisense, it will be created with default viewer privileges.
jti	Yes*	A unique string added to the token that is used to prevent replay attacks, by making sure the token is used only once.
exp	No	Expiration time of the token. After that time the token becomes invalid, and the user will be redirected again to the remote login URL for re-authentication. If not present, the token will expire within one week. The value must be the number of seconds since UNIX epoch.

* You can set this attribute as optional in the Sisense REST API v1.0 through the POST settings/SSO endpoint.

SSO Code Samples

[C#](#)

[Java](#)

[Javascript](#)

[PHP](#)

[Ruby](#)

[Python](#)

return_to URL

When Sisense redirects a user to your login script, Sisense passes a return_to parameter in the URL. This parameter contains the page that Sisense will return the user to after the authentication succeeds. For example:

1. A customer visits your site opens a dashboard embedded through an iFrame.
2. Sisense recognizes that the user is not authenticated.
3. Sisense redirects the user to:

https://yourcompany.com/sisense/sso?return_to=https://yourcompany.sisense.com/dashboards/

All your script needs to do, is take the return_to value from the invoked URL and pass it back to Sisense when submitting the JWT token. In other words, upon authentication on your side, your script redirects the user to:

https://yourcompany.com/access/jwt?jwt=payload&return_to=http://yourcompany.sisense.com/dashboards/

Configuring Sisense as a Sub-Domain with SSO

To authenticate your users locally and allow them to access Sisense, your first step should be to configure Sisense as a sub-domain of your web application and embed Sisense into your web application with SSO.

Note: Sisense also works when embedded in cross-domain iFrames.

To configure Sisense as a Sub-Domain:

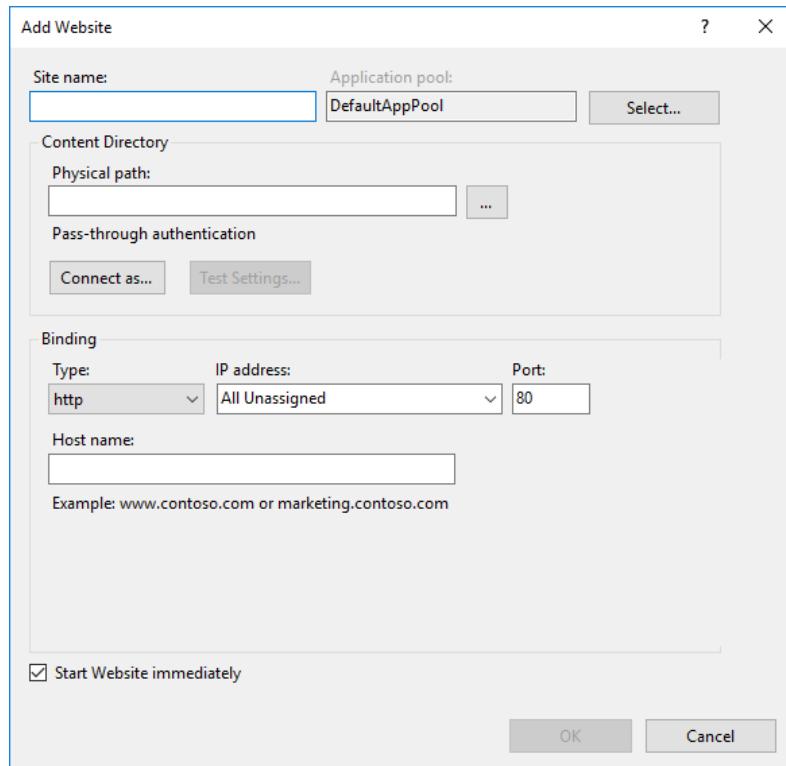
1. In IIS Management Console, add your website to IIS.



2. In the **Add Website** window, in the field **Site Name**, enter the site name.

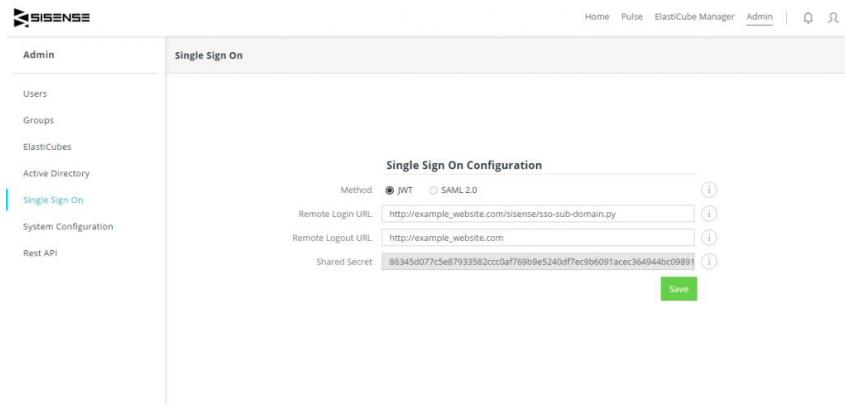
Note: You can not leverage SSO when you enter a DNS

address with an underscore “_” with Internet Explorer.



3. In **Physical path**, enter the subdomain directory.
4. Under the Binding area, in **Host name**, change the existing SisenseWeb site binding to use **sisense.example.website.com** as host name on **port 80**.
5. Open the file **C:\Windows\System32\drivers\etc\hosts** and add mapping for the sites:
192.168.5.148 sisense.examplewebsite.com
192.168.5.148 example.website.com
6. Sign in to the Sisense Web Application at **sisense.examplewebsite.com** and configure the SSO as

pictured below:



The screenshot shows the Sisense Admin interface with the 'Single Sign On' tab selected. The left sidebar includes options like 'Users', 'Groups', 'ElastiCubes', 'Active Directory', 'Single Sign On' (which is highlighted in blue), 'System Configuration', and 'Rest API'. The main content area is titled 'Single Sign On Configuration' and contains fields for 'Method' (set to 'JWT'), 'Remote Login URL' (set to 'http://example_website.com/sisense/sso-sub-domain.py'), 'Remote Logout URL' (set to 'http://example_website.com'), and 'Shared Secret' (set to '86345d077ce9e87933582cc0af769b9e5240d7ec9b6091ace364044bc0989'). A 'Save' button is at the bottom right.

7. Place the following SSO script in the server location corresponding to the **Remote Login URL** in the server's root directory. The SSO script can be implemented in any server-side language. This example uses Python. Example code has been attached for C# and Python.

[Python script example](#)

[C# script example](#)

In **index.html** from **examplewebsite.com**, the IFrame source is the Sisense dashboard URL.

index.html from **examplewebsite.com**:

```
<html>
<head>
<title>Example Website</title>
</head>
<body>
<p><b>examplewebsite.com</b> - <b>SSO</b> login with embedded dashboard from <b>sisense.examplewebsite.com</b></p>
<iframe width="100%" height="100%" src='http://sisense.examplewebsite.com/app/main#/dashboards/53b29843751b655443000018?embed=true' />
</body>
</html>
```

Navigate to **examplewebsite.com** and you should see the specific dashboard you embedded.

Logging Users Out

When a user is logged in, anyone using that browser can access the session, or users may encounter an issue where they remain logged in until the Sisense cookie is cleared.

Users are logged out when the session ends. A session ends when the user closes their browser or according to the value of the attribute `exp` you send in the JWT payload.

You can log the user out through the Sisense REST API.

To manually log a user out, access version .9 of the REST API. Through the Auth method, you can issue a get request to log out specific users.

auth

Show/Hide | List Operations | Expand Operations

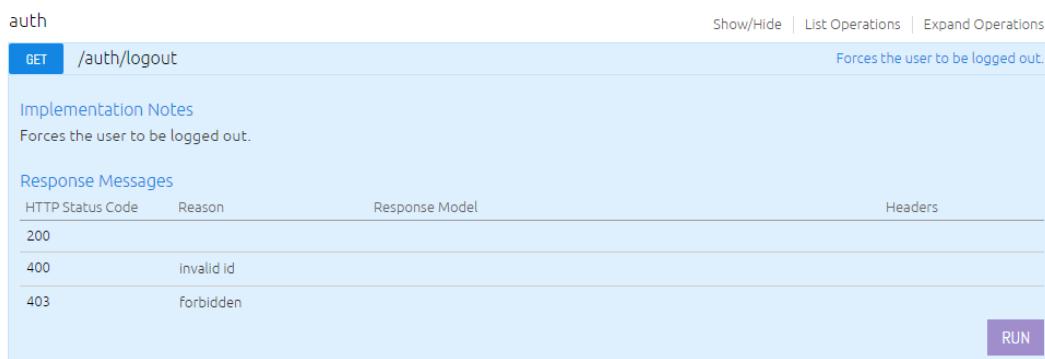
GET /auth/logout Forces the user to be logged out.

Implementation Notes
Forces the user to be logged out.

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200			
400	invalid id		
403	forbidden		

RUN



While the logout REST API can delete the SSO authentication cookie, it can only delete it when the call is made from within the Sisense domain. Scripts on different pages can access each other only if the pages that executed them are at locations with the same protocol.

If you have embedded Sisense in an iFrame and you want to log out the user from your application and Sisense, you can use the `window.postMessage` method to call the logout when the users asks to logout from your application. This method overcomes any cross-origin communication limitations. Sisense has created a plugin that



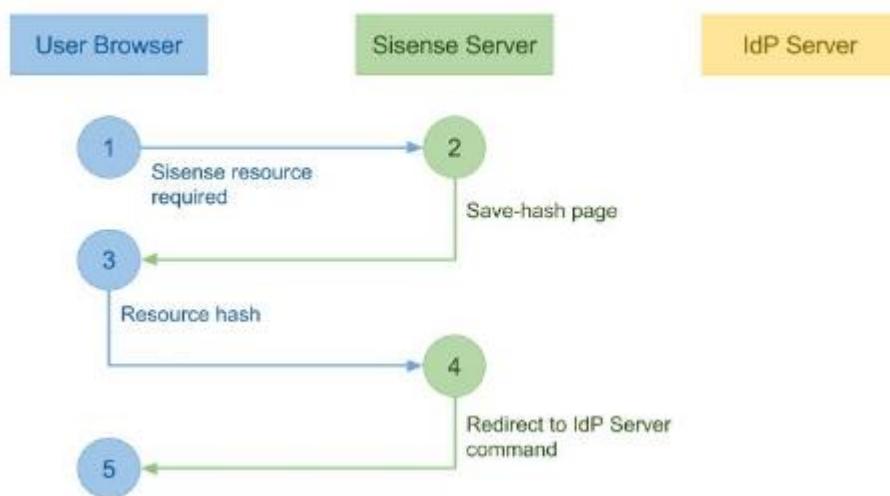
SISENSE

implements a listener, which calls the Logout API when the postMessage method() is called. For more information, click [here](#).

SSO via SAML 2.0

The Sisense SAML authentication process is based on the SAML 2.0 protocol, and is explained in the following diagrams:

Sisense SAML auth workflow (1st phase)



Phase 1

The first phase of this process begins when a user requests a resource from Sisense via their browser (1). The browser generates a resource request to the Sisense server. The server processes this request (2), and for unauthenticated users, returns a special save-hash page to save the requested hash data.

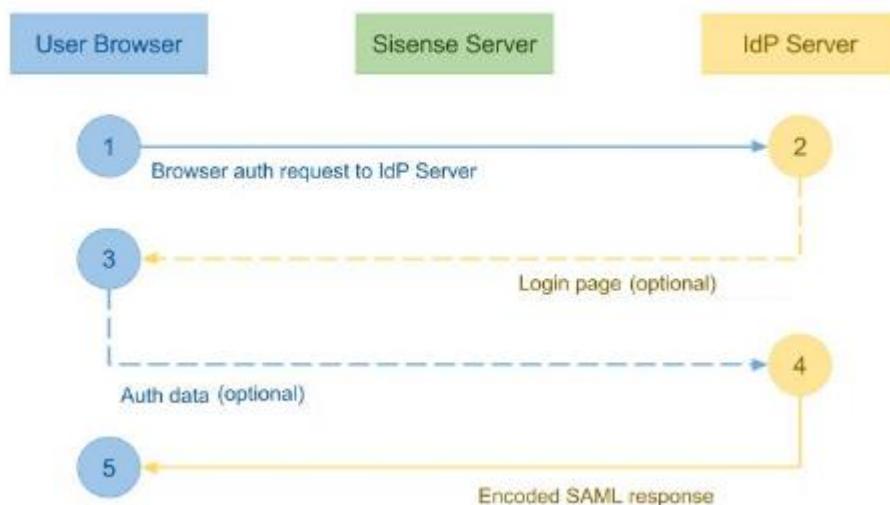
After the browser loads the save-hash page from the Sisense server, it runs a script, which loads the requested resource hash from the URL and sends it to the Sisense Server (3).

At the end of this phase, the Sisense Server converts the requested resource URL and sends it to the browser as a part of the redirect to IdP Server command (4).

Phase 2

The second phase of the authentication process starts after the browser retrieves the redirect command from the Sisense Server, and sends the authentication request to the IdP Server (1) as described in the diagram below.

Sisense SAML auth workflow (2nd phase)



The base URL for this request is taken from the Remote Login URL field of the Single Sign On Admin page in the Sisense Web Application.

The requested resource address is passed to this request as a RelayState parameter. All other data is provided as a SAMLRequest parameter.

The next step of this phase depends on the user authentication state and the IdP implementation.

If the current user isn't logged in as an IdP user, the IdP server redirects the browser to your Login page (2), where the user enters their IdP credentials. After logging in, the IdP Server sends the

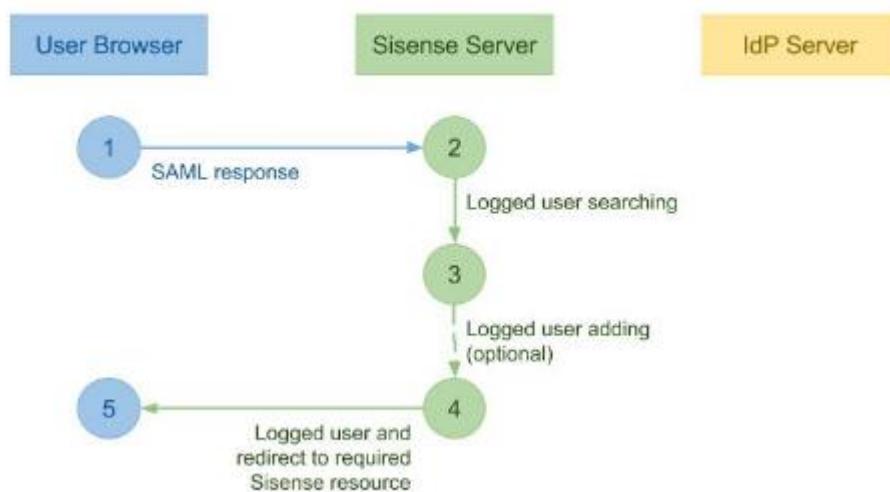
browser the Auto-Sign-In page with encoded data about the currently logged-in user (4).

After this document is loaded in the user's browser, it runs a script which creates the POST-query to the Sisense Server API, and passes the SAML response to this query (1). At this time, the third phase of the authentication process begins.

Phase 3

The Sisense Server handles the POST-query (2), decrypts it with the certificate specified in the Public X.509 Certificate field defined in the Single Sign On Admin page. The Sisense Server uses the decrypted User ID as a key to locate the Sisense user in the internal database.

Sisense SAML auth workflow (3rd phase)



If a user is not found in the system, Sisense creates a new user (3).

When Sisense creates a new user, Sisense analyzes the "memberOf" field to locate one or more groups related to the logged-in user.

If the "memberOf" field is empty, the user is assigned the role: "Viewer". If the "memberOf" field contains one or more groups, and

the groups were previously defined in Sisense, the newly created users' Role is taken from the groups' default role. When multiple Sisense Groups are found, the user is assigned the role with the maximum privileges.

After a user is created in the system, an administrator can modify the user role, if needed.

Below, is an example of an SAML XML where the "Test" group is specified:

```
<saml:Attribute  
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-  
format:basic"  
      Name="memberOf">  
      <saml:AttributeValue  
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-  
instance"  
            xsi:type="xs:string">  
          Test  
      </saml:AttributeValue>  
  </saml:Attribute>
```

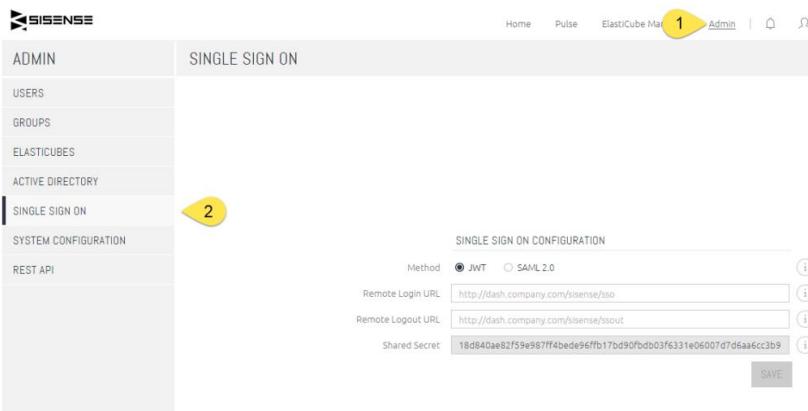
After the user is located or created, a new session is initialized, and the Sisense Server redirects the user browser to the requested resource (4).

Enabling SSO via SAML in Sisense

After you have configured your SAML server, sign in to Sisense as an Administrator and follow the instructions below.

To enable SAML in Sisense:

1. In the Sisense Web Application, click **Admin** and select **Single Sign On**.



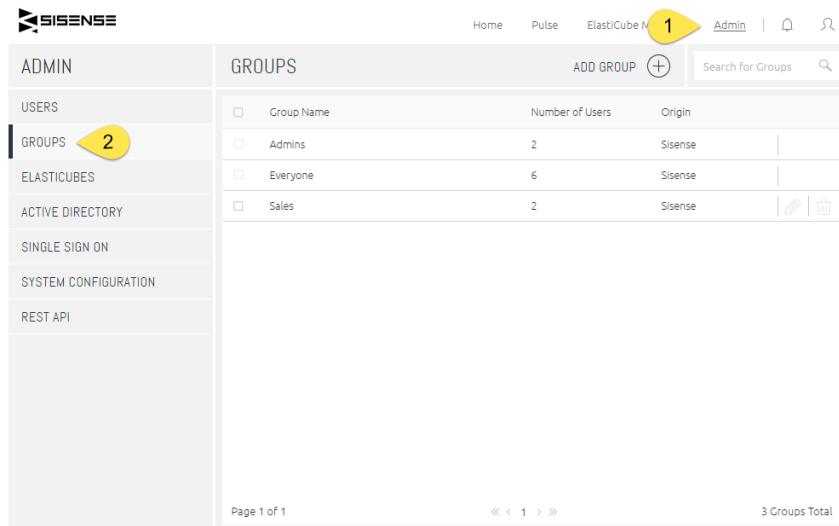
2. In the Single Sign On page, select **SAML 2.0**.
3. In the **Remote Login URL** field, enter the SAML Login endpoint. Sisense redirects the user to this field when they sign in. This value should be provided by the IdP Service.
4. In the **Remote Logout URL** field, enter the SAML Logout endpoint. Sisense redirects the user to this field when they sign out. This value should be provided by the IdP Service.
5. In the **Public X.509 Certificate** field, enter your public key for your SAML configuration. This value should be provided by the IdP Service.
6. Click **Save**. SSO via SAML 2.0 is configured.

Sisense Default Role Set-Up

When an authenticated user is not found in the Sisense database, a new account is created. The user role is specified based on the user group/groups default role.

To define a group's role:

1. In the Sisense Web Application, click **Admin** and select **Groups**.



The screenshot shows the Sisense Admin interface. The top navigation bar includes Home, Pulse, ElastiCube M, Admin (highlighted with a yellow arrow), and search functions. The left sidebar under the ADMIN heading has links for USERS, GROUPS (highlighted with a yellow arrow labeled '2'), ELASTICUBES, ACTIVE DIRECTORY, SINGLE SIGN ON, SYSTEM CONFIGURATION, and REST API. The main content area is titled 'GROUPS' and contains a table with three rows: 'Admins' (2 users, Sisense origin), 'Everyone' (6 users, Sisense origin), and 'Sales' (2 users, Sisense origin). An 'ADD GROUP' button with a plus sign is at the top right of the table. A search bar 'Search for Groups' is also present.

Group Name	Number of Users	Origin
Admins	2	Sisense
Everyone	6	Sisense
Sales	2	Sisense

2. Click **Add Group**. The Create a New Group window is displayed.

3. In the **Create a New Group** window, select the default role of the group.

Create a New Group

Group Name

Type in a descriptive name for the new group.

Group Users

+ Add Group Users



No members added yet

Choose default members role: Viewer Designer

CANCEL

4. Click **Save**.

Note: Changes to the group's default role are applied when new users are created, and do not affect existing users. After a user is created in the system, an administrator can modify the user role, if needed.

ElastiCube Management

Adding and Removing ElastiCube Servers

This section describes how to manage your ElastiCubes.

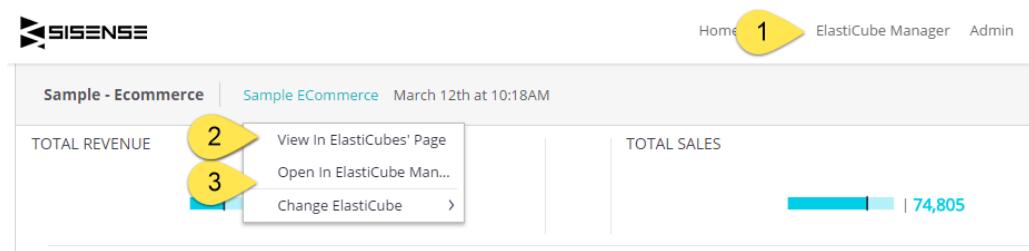
By default, the Sisense Web Application is connected to the ElastiCube server that is installed with it on the same machine.

By default, all ElastiCubes on that machine are accessible to all dashboard users, and are visible under LOCALHOST.

Opening an ElastiCube in the ElastiCube Manager

You have a number of ways to open your ElastiCube in the ElastiCube Manager:

- ▶ From Sisense web app:
 - Click **ElastiCube Manager** in the top menu (1).
 - If you have an open dashboard, click on the ElastiCube name next to the dashboard name. Select **View in ElastiCube's Page** (2) to open the ElastiCube management page in the web app. In the **ElastiCubes** page, select the ElastiCube and click **Open**. To open the ElastiCube of the open dashboard, click **Open in ElastiCube Manager** (3).

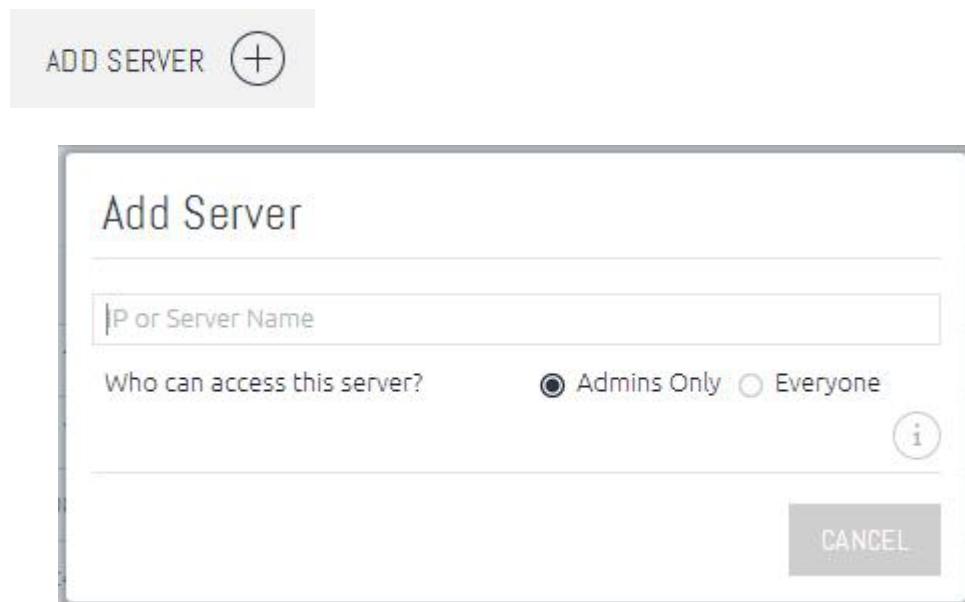


- ▶ From Windows: From the Start menu, open **Sisense ElastiCube Manager**.

Adding Another ElastiCube Server

To add another ElastiCube Server to Sisense:

1. Click **ADMIN** in the upper right corner and select the **ELASTICUBES** tab on the left.
2. Click **ADD SERVER**.



3. Select whether this ElastiCube Server is intended for Admins only. This selection also applies to all new ElastiCubes that will be added to this ElastiCube Server. A user that has access to an ElastiCube Server can access all the ElastiCubes on it.
4. Add the IP of the ElastiCube Server to which you want to connect and click **ADD SERVER**. The new ElastiCube Server is added to the list with all its ElastiCubes.
5. Click **SAVE**. Once a new ElastiCube Server has been added, its ElastiCubes immediately become available to all designers (by

default) when they create a new Dashboard.

To define which users are allowed to access the data of an ElastiCube Server, see [Assigning Rights to an ElastiCube Server](#).

To define which users are allowed to access each specific ElastiCube, see [Assigning Rights to an ElastiCube](#).

Removing an ElastiCube Server

To remove an ElastiCube server from the list:

► Hover over its title and click the Trash  button.



ElastiCube	Created On	Last Built	Size On Disk	Access Rights	Data Security
Sample EC...	Feb-10, 20...	Mar-12, 20...	60.57MB	Access Rights	Data Security
Sample He...	Jul-02, 201...	Mar-12, 20...	2.37MB	Access Rights	Data Security
Sample Le...	Mar-03, 20...	Mar-12, 20...	3.46MB	Access Rights	Data Security

ElastiCube	Created On	Last Built	Size On Disk	Access Rights	Data Security
Sample EC...	Feb-10, 20...	Mar-12, 20...	60.69MB	Access Rights	Data Security
Sample He...	Jul-02, 201...	Mar-12, 20...	2.37MB	Access Rights	Data Security
Sample Le...	Mar-03, 20...	Mar-12, 20...	3.46MB	Access Rights	Data Security

Note: The LOCAL MACHINE server cannot be removed.

Querying ElastiCubes during Builds

Sisense ElastiCube Sets are collections of ElastiCubes with identical schemas that allow you to query running ElastiCubes within the ElastiCube Set while other ElastiCubes are building.

Note: From Sisense V6.5 onwards, you can query ElastiCubes during accumulative builds without ElastiCube Sets. However, to ensure that your users can query ElastiCubes after a failed build, you can implement ElastiCube Sets to handle your queries while Sisense rebuilds the failed ElastiCube.

Grouping ElastiCubes into ElastiCube sets has several benefits:

Allowing viewers to query the most up-to-date cubes within an ElastiCube Set.

Reducing server load during builds by using multiple nodes in which only the non-building node is queried.

Achieving data redundancy by running builds interchangeably across multiple nodes.

To maximize availability and reduce build and dashboard response times, administrators can distribute resource allocation across multiple Sisense nodes. By adding ElastiCubes across multiple servers to a single ElastiCube Set, administrators can configure ElastiCube Sets to run builds interchangeably via the Sisense Orchestrator Service so the most up-to-date and complete ElastiCubes are queried.

The Sisense Orchestrator Service is an automated service that can be configured to synchronize builds across the ElastiCube Set. If you change your schemas within any of the ElastiCubes, you must

manually change the schemas in all of the ElastiCubes and rebuild the cube entirely.

This page provides the following information:

1. Instructions on Creating and Deleting ElastiCube Sets.
2. A walkthrough for how you can create a set and configure the Sisense Orchestrator Service. See Working with ElastiCube Sets.
3. (Optional) Instructions on how to implement and configure the Sisense Orchestrator Service. See Activating the Sisense Orchestrator Service.
4. How to update an ElastiCube within an ElastiCube Set.

Creating ElastiCube Sets

From the Admin page, Administrators can create ElastiCube Sets.

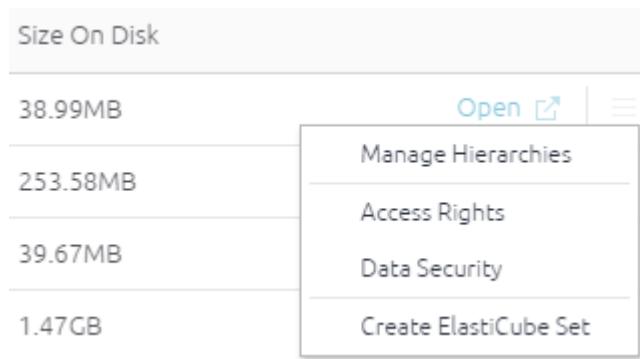
ElastiCube Sets require at least two ElastiCubes. The administrator can add multiple ElastiCubes to a set, however, ElastiCubes can only be part of a single ElastiCube Set at a time. After an administrator creates an ElastiCube Set or adds an ElastiCube to an existing set, the ElastiCube is disabled in the original server and no further actions can be performed outside of the set.

Sisense recommends that you develop ElastiCubes in a Dev environment or separate ElastiCube to ensure they work before adding them to an ElastiCube Set.

To create an ElastiCube Set:

1. Click **Admin** in the upper right corner and select the **ELASTICUBES** tab on the left.

2. Hover over the ElastiCube you want to add to the ElastiCube Set and click on the menu that appears. Click **Create ElastiCube Set**.



3. In the Create ElastiCube Set window, enter a name for the set and select the ElastiCubes you want to include in the set.
4. Click **Create**.

Create ElastiCube Set

ElastiCube Set Name

ElastiCubes

Sample		
	localhost	
<input checked="" type="checkbox"/>	Sample	59.32MB
<input type="checkbox"/>	Sample Healthcare	59.32MB
<input type="checkbox"/>	Elastic Cube A	59.32MB
1 ElastiCube Selected		
CREATE		CANCEL

5. In the Existing Dashboards popup message, indicate if you want to associate your ElastiCube's dashboards with the ElastiCube Set or maintain the association with the original ElastiCube.

Existing Dashboards in added ElastiCubes

- Connect existing dashboards to new ElastiCube Set
 Leave existing dashboards with their current ElastiCube

OK

CANCEL

6. Click **OK**. The ElastiCube Set is created and displayed in the Admin page.

ElastiCube Set	
Server	ElastiCube
LocalHost	Elastic Cube A
LocalHost	Sample
LocalHost	Sample Healthcare

After you have created an ElastiCube Set, you can configure the ElastiCube Set's Hierarchies, Access Rights, and Data Security.

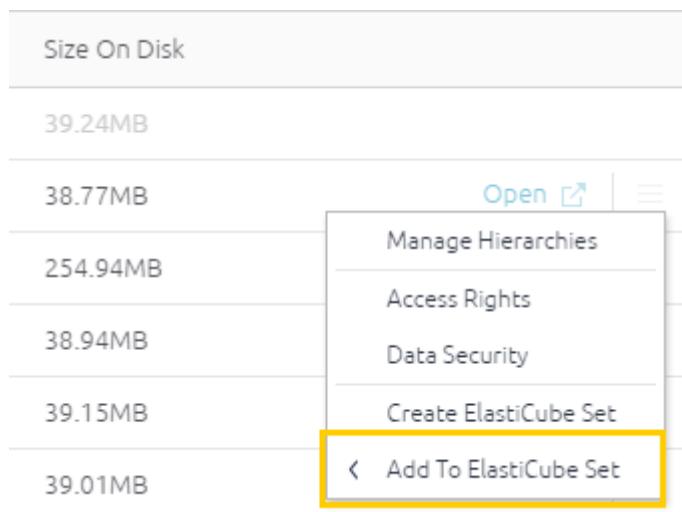
Adding ElastiCubes to ElastiCube Sets

After you have created an ElastiCube Set, you can add more ElastiCubes to the set.

When you add an ElastiCube to an existing ElastiCube Set, any dashboards associated with that ElastiCube are passed on to the ElastiCube Set.

To add an ElastiCube to the ElastiCube Set:

1. Click **Admin** in the upper right corner and select the **ELASTICUBES** tab on the left.
2. Hover over the ElastiCube you want to add to the ElastiCube Set and click on the menu that appears. Click **Add to ElastiCube Set** and select the set that you want to add the ElastiCube to.



OR

In the ElastiCube Set table, click  and select the ElastiCubes to be added to the set. Click **Save** after you have

selected the relevant ElastiCubes.

Create ElastiCube Set

ElastiCube Set Name

ElastiCube Set

ElastiCubes

Sample	59.32MB
▼ LocalHost	
<input checked="" type="checkbox"/> Sample	59.32MB
<input type="checkbox"/> Sample Healthcare	59.32MB
<input type="checkbox"/> Elastic Cube A	59.32MB
1 ElastiCube Selected	

CREATE

CANCEL

3. The ElastiCube is added to the ElastiCube Set and displayed in the ElastiCube Set table on the Admin page.

Deleting ElastiCube Sets

Administrators can delete ElastiCube Sets. Deleting an ElastiCube Set removes the set from the Admin page and any dashboards associated with that set.

To delete an ElastiCube Set:

1. Hover over its title and click the Trash icon.

2. Click **Delete** to confirm that you want to delete the ElastiCube Set.

Working with ElastiCube Sets

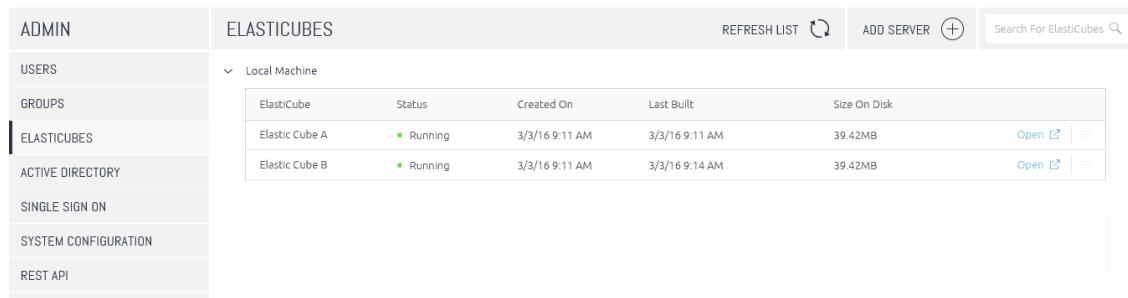
This section describes how to create your first ElastiCube Set and configure it for high availability.

In this example, let's assume that you have an ElastiCube, ElastiCube A, and you want to add it to a set to ensure that the data can always be queried, even during a build.

The first step is to create a copy of ElastiCube A. The ElastiCubes can have different names, but the schemas must be identical.

Note: If your ElastiCubes are located on a single node, the ElastiCubes cannot have the same name.

In the ElastiCube Manager, we will save a new version of ElastiCube A as ElastiCube B. In Sisense, you can see both ElastiCubes are identical (Select Admin > ELASTICUBES)



ELASTICUBES	REFRESH LIST	ADD SERVER	Search For ElastiCubes
Elastic Cube A	Running	3/3/16 9:11 AM	39.42MB
Elastic Cube B	Running	3/3/16 9:11 AM	39.42MB

Now, let's create an ElastiCube Set that contains both ElastiCubes.

Click  > **Create ElastiCube Set**. For more information, see Creating ElastiCube Sets.

Select the ElastiCubes to be added to the set. For this set, we will select ElastiCube A and B, which have different names, but identical schemas.

Create ElastiCube Set

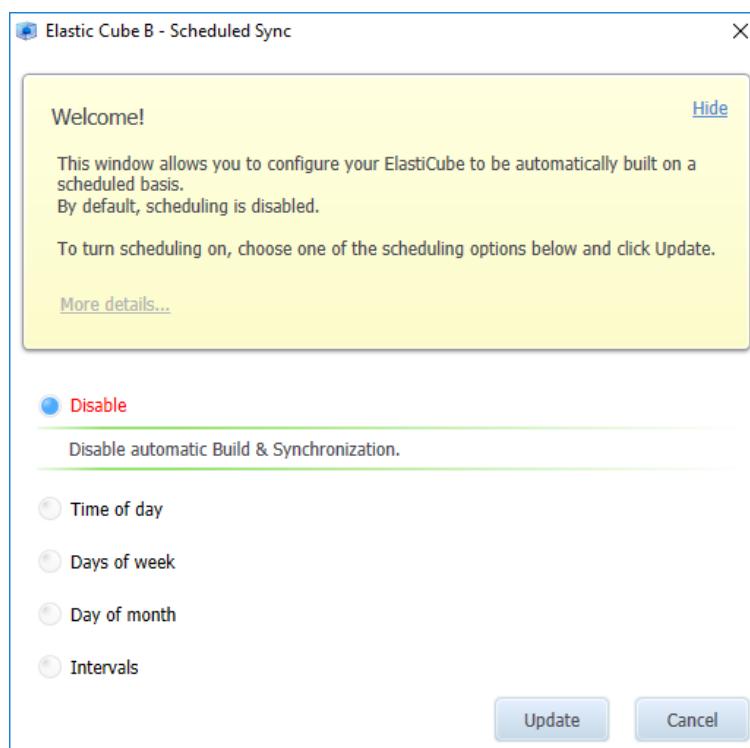
Click **Create** and confirm that you want to create a new set with these ElastiCubes. The set and its ElastiCubes are now displayed at the top of the Admin page.

ElastiCube Example					Manage Hierarchies	Access Rights	Data Security		
Server	ElastiCube	Status	Created On	Last Built	Size On Disk				
LocalHost	Elastic Cube A	● Running	3/3/16 9:11 AM	3/3/16 9:11 AM	39.42MB				Open
LocalHost	Elastic Cube B	● Running	3/3/16 9:11 AM	3/3/16 9:14 AM	39.42MB				Open

Now that the set is built, it's time to schedule automatic builds with the Sisense Orchestrator Service. This service automatically builds the ElastiCubes according to your preferences. For example, you can

schedule the builds to run every 10 minutes or at midnight when queries are at their fewest.

To schedule automatic builds, first you should disable any scheduled builds you have for the ElastiCubes. In the ElastiCube Manager in the top menu, select **ElastiCube > Schedule Build Settings** for each ElastiCube with a scheduled build, then select **Disable > Update**.



Now that all the ElastiCubes' scheduled builds are disabled, it's time to configure the Sisense Orchestrator Service to build the ElastiCubes within the set. To configure the Sisense Orchestrator Service, you must modify the Config.JSON file that was installed when you installed the latest version of Sisense. This file defines how the ElastiCube Set is being built, the order of the build, and how frequently a build occurs.

This file is located in the Sisense.Orchestration Config folder (/Sisense.Orchestration/Config/) of your Sisense Installation folder.

For this set, let's assume that you want to schedule an accumulative build for midnight. Within the **elastiCubes** object, we listed both ElastiCubes to be built by the Sisense Orchestrator Service. The order of the ElastiCube builds is in the order of oldest to newest. Next, in the **timer** object, we listed the hour when the build is to take place. The value is defined in Cron format, so midnight equals 0 minutes 0 hours "0 0 * * *".

Finally, we define the flow of the build. We want an accumulative build, but if for some reason this fails, we want to run an entire build as a back-up. Remember, when an accumulative build fails, the ElastiCube needs to be completely rebuilt to ensure that the data is intact. In the build **object**, we define the type of builds to be initiated and their order. The order of the values determines what build type is initiated first. In this example, the value is "FullUpdateExisting", "Full", which means run an accumulative build first and if that fails, run a full build. For a complete explanation on how to define the rest of the objects list in the Config.JSON file, see Activating the Sisense Orchestrator Service.

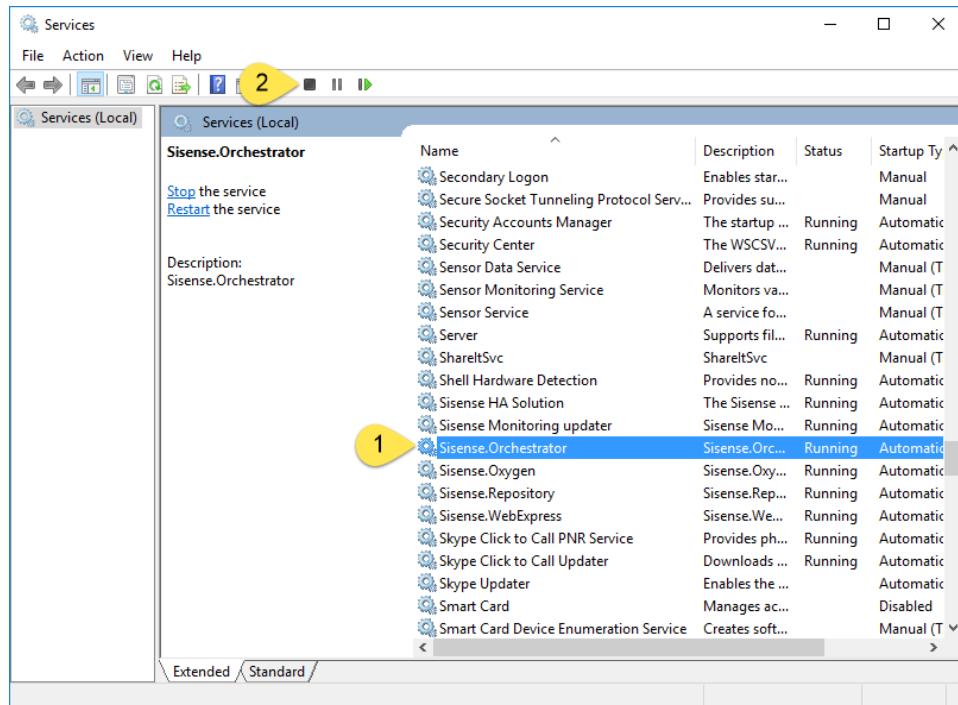
Now, save the file and Sisense automatically begins to run the builds according to the schedule defined in the timer object.

```
{  
  "servers": {  
    "localhost": {  
      "ip": "localhost",  
      "port": "8081",  
      "apiKey": "eyJ0eXAiHDHDHsXKSSZJdnUzI1NiJ9.eyJ1c2VyIjoiNTY1N332  
      RW2eygLOuVLufGYC-8vY"  
    }  
  },
```

```
"tasks": [ {  
    "name": "taskName",  
    "elasticubes": [ {  
        "server": "localhost",  
        "cube": "ElastiCube A"  
    }, {  
        "server": "localhost",  
        "cube": "ElastiCube B"  
    } ],  
    "timer": "0 0 * * *",  
    "build": {  
        "queue": [ "FullUpdateExisting", "Full" ],  
        "flow": "sync"  
    },  
    "email": {  
        "when": "Fail"  
    }  
},  
    "emails": [ {  
        "email": "JohnDoe@sisense.com",  
        "name": "John Doe"  
    } ],  
    "logLevel": "info"  
}
```

The Sisense Orchestrator Service automatically builds the ElastiCubes within the set, but if the schema changes, you must manually rebuild each ElastiCube within the set so the schemas match across all the ElastiCubes. While working with the schemas of an ElastiCube that is part of an ElastiCube Set, you should stop the Sisense Orchestrator Service.

To stop the Sisense Orchestrator Service, open Windows Services, select Sisense.Orchestrator, and click ■.



After the ElastiCubes are rebuilt, restart the Sisense Orchestrator Service by selecting Sisense.Orchestrator and click .



The ElastiCube Set is now configured and running ElastiCubes can be queried while the other ElastiCubes are being built.

Activating the Sisense Orchestrator Service

The Sisense Orchestrator Service automates the build process for ElastiCube Sets and determines how frequently the build process occurs.

To activate the Sisense Orchestrator Service, you must create and save a JSON file called config.json in the Sisense.Orchestration Config folder (/Sisense.Orchestration/Config/) of your Sisense Installation folder. This file defines how the ElastiCube Set is being built, the order of the build, and how frequently a build occurs.

After you save the file, the Sisense Orchestrator Service automatically begins to build ElastiCubes in your ElastiCube Set as defined by you in the config.json file. You can review the progress of the build in the /Sisense.Orchestration/Logs folder in a log file whose file name you define in the JSON file.

Note: During this phase it is recommended to stop the Sisense Orchestrator Service while importing data and reactivate it after the importing process is completed.

The objects that you must include in the config.json file and their descriptions are provided in the table below:

Key	Value
Servers	<p>The servers block contains a server object that defines each server that hosts an ElastiCube within your ElastiCube Set. Each server must include a server name, IP, port, and a unique API key.</p> <p>For example, if you have two different servers, each server must be represented as a separate object with its own name, IP, and port.</p>
ServerName	Your server's name. The value of this key is used when associating the ElastiCube Set with a server in the value of the Tasks key.
IP	The IP address of your server.
Port	The port of your web server.
API	<p>The access token used to identify your requests to the Sisense web server.</p> <p>For more information about the access token, see Retrieving a Token.</p>

Key	Value
Tasks	The Tasks object contains an array of ElastiCube objects. This object can contain multiple ElastiCube objects, but a minimum of two are required for a set.
Name	The name of the plan. The value of this parameter is the filename of the log file for the Sisense Orchestrator Service.
ElastiCubes	The ElastiCube object that holds an array of ElastiCubes. Each ElastiCube must include the name of its server and the name of the cube as defined in the Elastic Manager.
Server	The name of the server that hosts the ElastiCube.
Cube	The name of the cube to be included in the set.
Timer	<p>The frequency of builds in Cron format.</p> <p>Some examples:</p> <ul style="list-style-type: none"> -To run a build each night at midnight, enter the value "0 0 * * *" -To run a build every hour, enter the value "0 * * * *" <p>See the full Cron format reference here.</p>
Build	The Build object contains two keys, Queue and Flow, which determine the order in which the build is completed and the type of build.
Queue	<p>The value of this key is the type of build Sisense should attempt and in what order. There are two possible values delimited by a comma:</p> <p>FullUpdateExisting: Attempts an accumulative build.</p> <p>Full: Attempts an entire build.</p>

Key	Value
	<p>Sisense recommends the following value: "FullUpdateExisting","Full"</p> <p>In this example, Sisense attempts to do an accumulative build first, and if it fails, Sisense attempts an entire build. If the build fails for all builds, Sisense retries again according to the next scheduled build.</p>
Flow	<p>Determines the order of the build. A build occurs according to the value of Timer. For example, if your builds are scheduled for midnight, then the Orchestrator begins to build the ElastiCubes at midnight according to the order you define below.</p> <p>Single: Builds a single cube, then waits for the next time a build is scheduled according to the value of Timer. The cube to be built is a cube that previously failed or the cube with the oldest build.</p> <p>Sync: Builds the cubes with the oldest build first and then the remaining cubes in the order of oldest to newest until all ElastiCubes are updated.</p> <p>All: Builds the cube with the oldest build first and then the remaining cubes asynchronously.</p>
Email	This object determines what triggers activate Sisense to send an email to users you define in the Emails block
When	<p>Indicates when an email alert is triggered. Email alerts can be triggered for the following scenarios:</p> <p>None: No email is ever sent.</p> <p>Build: An email is sent when a build is complete.</p> <p>Fail: An email is sent when a build fails.</p>
Emails	This object defines who receives an email and their email address when an email is triggered according to the events you define in the Email object.

Key	Value
Email	The email address where emails are to be sent depending on the value of the When key.
Name	The name of the recipient of the email.
LogLevel	The type of info returned in the log. The possible values you can enter are Info and Error.

Example:

The following is an example of a config.json file. In this example, the ElastiCube Set contains two ElastiCubes, ElastiCube1 and ElastiCube2, hosted on the two different servers, Server1 and Server2. The ElastiCubes, even though they are located on separate servers, are connected through the Tasks object. The Sisense Orchestrator Service attempts to build the ElastiCube Set every 10 minutes. If the build fails, Sisense sends an email to JohnDoe@sisense.com.

```
{
  "servers": {
    "Server1": {
      "ip": "localhost",
      "port": "80",
      "apiKey": "eyJ0eXAiHDHDHsXKSSZJdnUzI1NiJ9.eyJ1c2VyIjoiNTY1N332RW2eygL0uVLufGYC-8vY"
    }
    "Server2": {
      "ip": "192.168.5.134",
      "port": "8081",
      "apiKey": "dgcfdgSSdSSZJdnUzI1NiJ9.eyJ1c2SDFSDFSDFDSFDGSSDFVfasddgdeygL0uVLufDFeY"
    }
  },
  "tasks": [ {
    "name": "ElastiCube1"
  },
  {
    "name": "ElastiCube2"
  }
]
}
```

```

"name": "taskName",
"elasticubes": [
  {
    "server": "Server1",
    "cube": "ElastiCube1"
  },
  {
    "server": "Server2",
    "cube": "ElastiCube2"
  }
],
"timer": "* / 10 * * * *",
"build": {
  "queue": ["FullUpdateExisting", "Full"],
  "flow": "sync"
},
"email": {
  "when": "Fail"
},
"emails": [
  {
    "email": "JohnDoe@sisense.com",
    "name": "John Doe"
  }
],
"logLevel": "info"
}

```

Overcoming Build Failures in ElastiCube Sets

In Sisense, if an accumulative build fails, you must rebuild the ElastiCube that failed entirely to ensure the validity of the ElastiCube.

If you use the Sisense Orchestrator Service, Sisense recommends that you define the value of the **Queue** key as “FullUpdateExisting, Full”

```
"queue": ["FullUpdateExisting", "Full"]
```

In this configuration, the Sisense Orchestrator Service attempts to build the set using accumulative builds and in the event of a failure, a full build is initiated.

Retrieving a Token

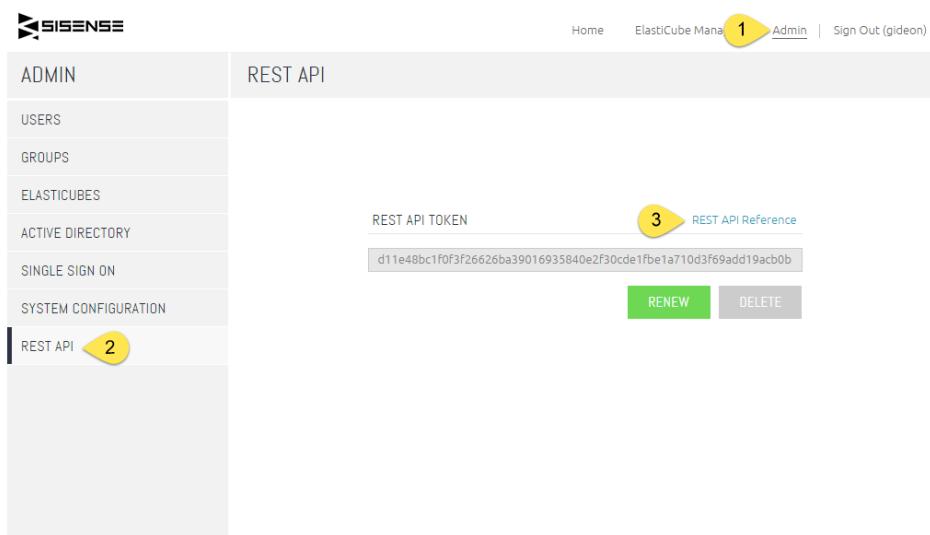
As part of the config.json file, you must provide a token that Sisense uses for authenticating your ElastiCube Sets.

You can retrieve this token through Sisense's API documentation.

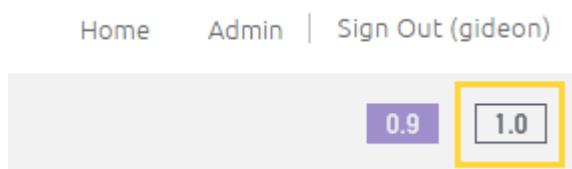
After you retrieve the token, save it as the value of apiKey in the config.json file.

To retrieve a token:

1. In Sisense, access the API documentation, select **Admin> REST API > REST API Reference.**

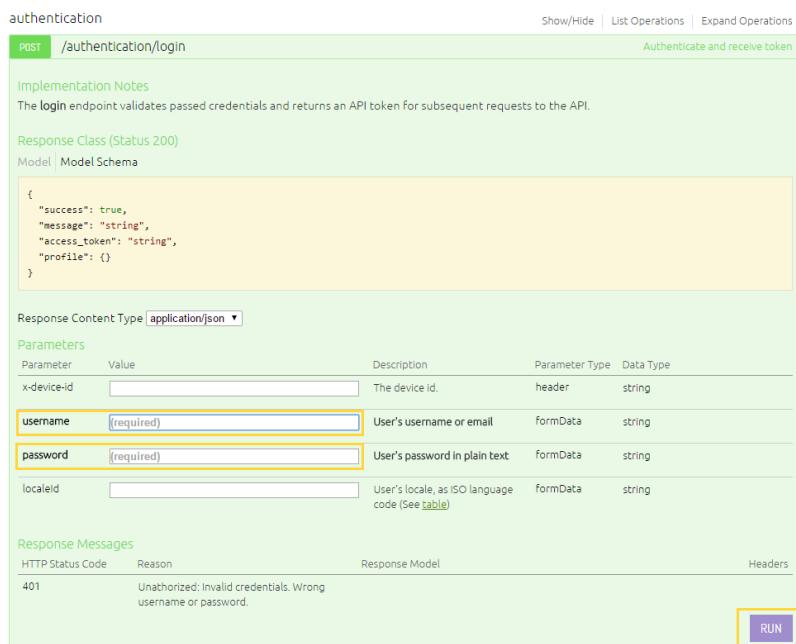


2. In the REST API Reference site, select Version 1.0 in the top-right corner of the page.



3. Open the authentication method.

4. Enter your login credentials and click **Run**. The access_token is displayed in the Response Body below.



The screenshot shows the Sisense API documentation for the `POST /authentication/login` endpoint. The `Response Class (Status 200)` section displays a JSON schema:

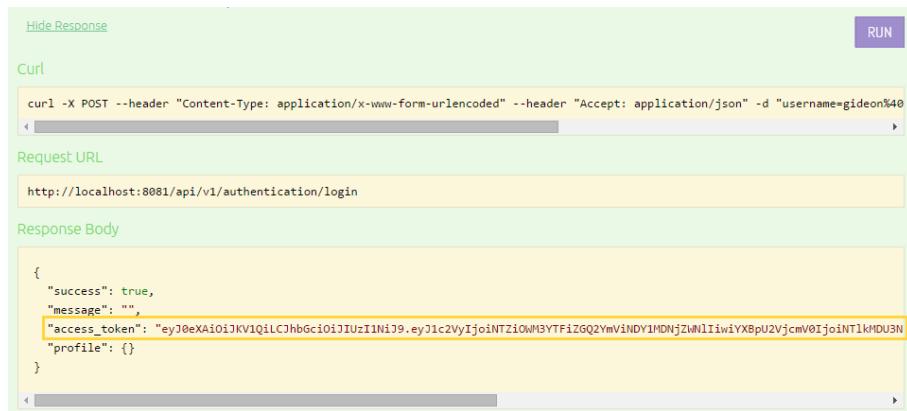
```
{
  "success": true,
  "message": "",
  "access_token": "string",
  "profile": {}
}
```

The `Parameters` section lists the required fields:

Parameter	Value	Description	Parameter Type	Data Type
x-device-id	<input type="text"/>	The device id.	header	string
username	<input type="text" value="gideon"/>	User's username or email	formData	string
password	<input type="text" value="password123"/>	User's password in plain text	formData	string
localeId	<input type="text"/>	User's locale, as ISO language code (See table)	formData	string

The `Response Messages` section shows a 401 status code with the reason: "Unauthorized: Invalid credentials. Wrong username or password." A **RUN** button is located at the bottom right.

5. Copy the token and paste it into the value of apiKey in the config.json file.



The screenshot shows the curl command output for the `curl -X POST --header "Content-Type: application/x-www-form-urlencoded" --header "Accept: application/json" -d "username=gideon&password=password123" http://localhost:8081/api/v1/authentication/login` request. The `access_token` field is highlighted with a yellow box.

```

curl -X POST --header "Content-Type: application/x-www-form-urlencoded" --header "Accept: application/json" -d "username=gideon&password=password123" http://localhost:8081/api/v1/authentication/login
{
  "success": true,
  "message": "",
  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJcI2VybjoNTzIwM3YTF1ZQ2YmV1NDY1MDNjZWNIiwiYXBpu2VjcmV0IjoiNTlkMDU3N
  "profile": {}
}

```

Updating ElastiCubes within an ElastiCube Set

The Sisense Orchestrator Service automatically updates your data across ElastiCubes within an ElastiCube Set. If you want to modify the schema, for example, by adding new tables, you must manually

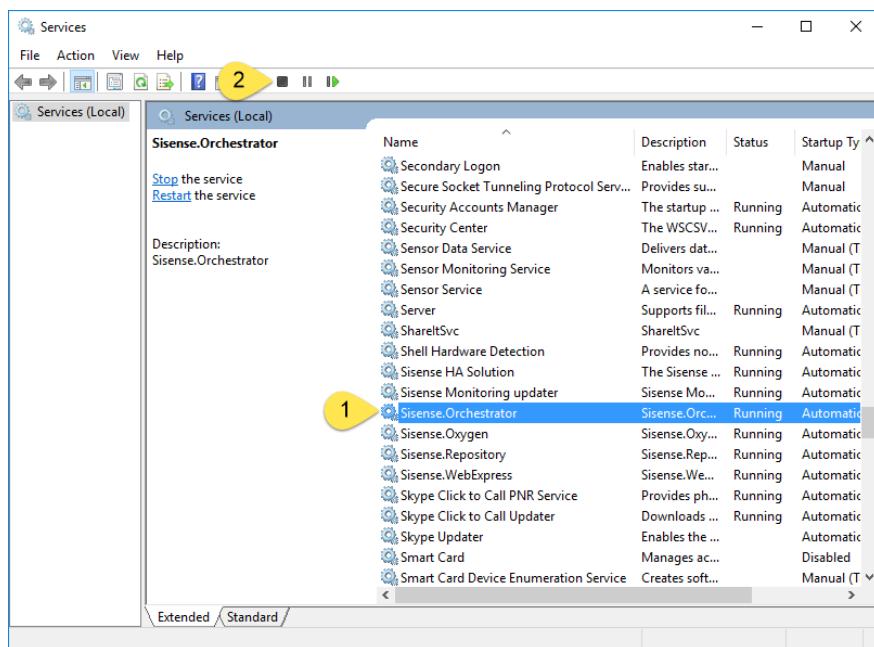
rebuild the ElastiCubes and then import that ElastiCube to the other machines hosting your ElastiCube Set.

While rebuilding your ElastiCube, you should deactivate the Sisense Orchestrator Service to prevent it from updating the ElastiCubes with different schemas.

After the schemas are identical for all the ElastiCubes across the ElastiCube Set, you can reactivate the Sisense Orchestrator Service to automatically update your ElastiCubes.

To update an ElastiCube within ElastiCube Set:

1. Stop the Sisense Orchestrator Service. Open Windows Services, select Sisense.Orchestrator, and click ■.



2. Apply your schema changes to the ElastiCube and run a full build.
 3. Export the ElastiCube and import it on your other machines.
- For more information, see Importing and Exporting ElastiCube Data.



SISENSE

4. In Windows Services, restart the Sisense Orchestrator Service by selecting Sisense.Orchestrator and clicking ▶.

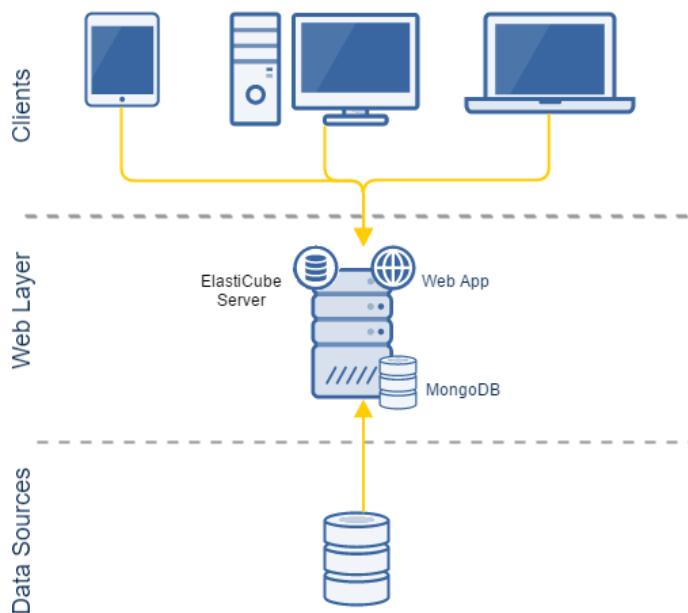
High Availability in Sisense

Overview

Sisense provides flexible design choices for supporting high availability (HA) and scalability for your Sisense configuration. To configure Sisense for high availability or optimized performance by scaling out Sisense servers, you must build in redundancy, thus reducing potential downtime or bottlenecks.

Note: High availability must be enabled in your license. For more information, please contact your Sisense representative or submit a request to Sisense Support through our [community website](#).

In a simple configuration, you interface with the Sisense Web Application that interacts with the ElastiCube.



This configuration represents the Sisense full stack solution. At least one instance of each of the following components must be active to

enable you update your data and allow your users to query that data from the dashboard:

Sisense Web Server

The Sisense Web Server hosts the Sisense Web Application that provides the user interface and hosts the API endpoints. If the Sisense Web Server fails, your users cannot access the Sisense Web Application to view dashboards or use the Sisense APIs.

MongoDB

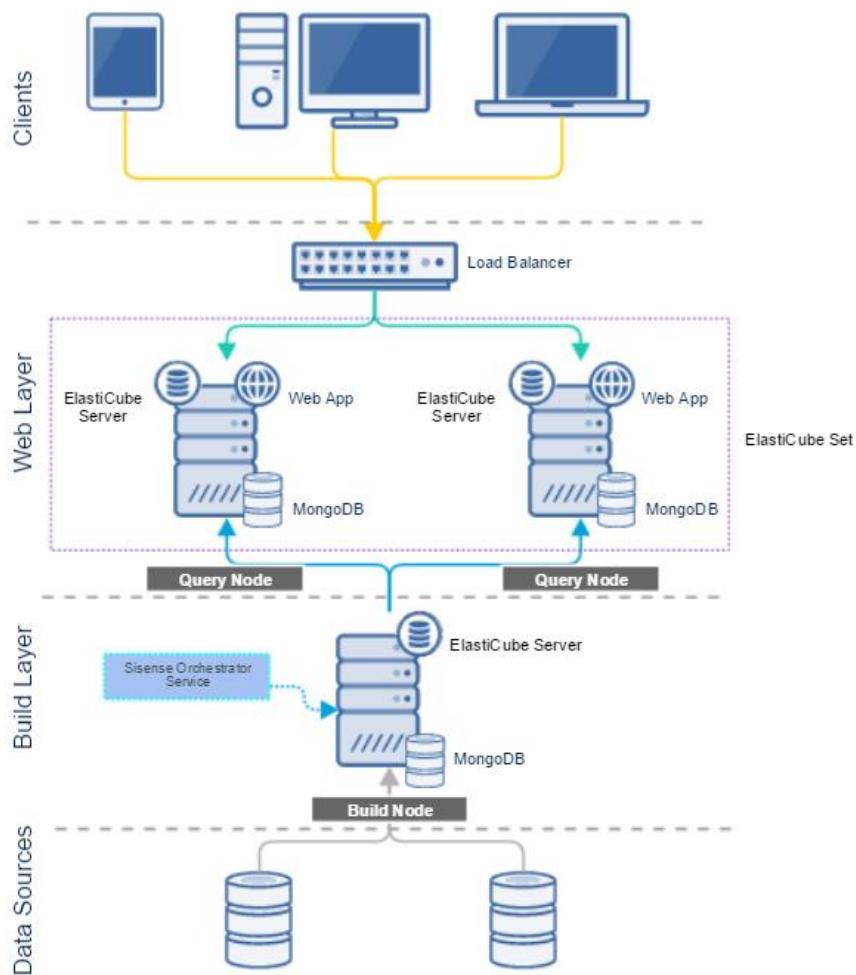
MongoDB is installed with Sisense and supports the Sisense Web Application. The MongoDB contains dashboard, filters, and user information necessary for ensuring data consistency across all web servers. In addition, MongoDB is used for authentication when you make calls to Sisense's APIs. If the MongoDB fails the Sisense Web Application will fail.

ElastiCube Server

The ElastiCube Server is installed locally on your computer and provides access to ElastiCubes. If access to the ElastiCube Server fails, queries from the Sisense Web Application will fail.

Understanding High Availability in Sisense

While Sisense is fully-functional in an environment without high availability, a multi-node configuration is necessary for scalability to support large amounts of concurrent users and redundancy in case of a failure of one of the Sisense components.



In a multi-node configuration, Sisense components are replicated.

Replicating each of these components provides redundancy and fault tolerance against the failure of any single component. The replicated components are combined into nodes. There are two types of nodes, a build node and query nodes. The query nodes, which handle user queries from the Sisense Web Application, are replicated to support high availability. The build node is typically not replicated as its failure only prevents building new ElastiCubes not issuing queries from the Sisense Web Application.

Build Node

The build node is responsible for building ElastiCubes and distributing the build to query nodes via the Sisense Orchestrator Service. The Sisense Orchestrator Service is an automated service that you configure on the build node to synchronize and distribute builds to the query nodes. For more information, see [Distributing ElastiCube Builds to Query Nodes](#).

Query Nodes

Query nodes are responsible for supporting queries from Sisense users. These nodes contain a web server, MongoDB, and an ElastiCube server. ElastiCubes are distributed by the build node to the query node. The query nodes' ElastiCubes are combined into ElastiCube Sets to support high availability by separating the web and ElastiCube servers across multiple query nodes. If a build node is distributing a build to one ElastiCube server, Sisense automatically directs any queries to the other ElastiCubes in the ElastiCube Set.

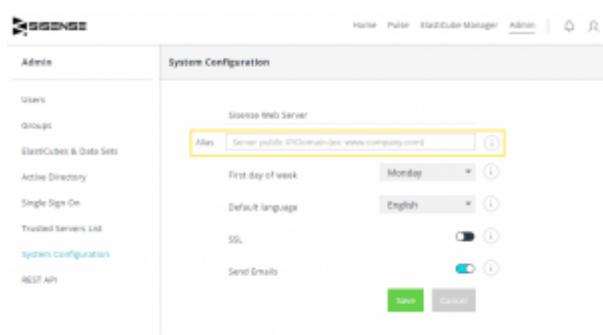
The query node can be configured as a single application stack where each node hosts a Sisense Web Application, ElastiCube Manager, and a MongoDB. In this configuration, if the machine hosting the components fails, the whole query node will fail. Queries will then be redirected to the next available query node. Another option is to host each component of the query node separately in a distributed application stack. In this configuration, if a component of the query node fails, the rest of the query node is not affected. For an example of a single application stack, see [Scenario 1](#) and for an example of a distributed application stack, see [Scenario 2](#).

Load Balancer

In addition to query and build nodes, to support a high availability configuration, you must handle load balancing on your side prior to directing traffic to one of your Sisense nodes. Load balancing spreads requests across multiple query nodes according to an algorithm you define and the current status of the query node.

When implementing ElastiCube Sets, Sisense's query nodes operate in active-active mode. This means that each of the query nodes is active and can handle requests when the node is not building and its components are available. For example, traffic could be spread 50-50 across two web servers and if a component fails, a load balancer should redirect traffic to the other available web server.

The URL of your load balancer should be provided as an Alias in the Admin section of the Sisense Web Application. This directs Sisense to send traffic to your load balancer, which then sends the traffic to the relevant server.



Within an ElastiCube set, you can define how queries are routed to the available ElastiCubes. For more information, see [Routing Queries in ElastiCube Sets](#).

Creating a High Availability Environment

There are many possible configurations you can implement to support high availability. What determines the type of configuration you choose based on three parameters:

1. **Redundancy:** Which components require redundancy?
2. **Concurrency:** How many users do you need to support?
3. **Resources:** How many systems can you add and maintain to your configuration?

Regardless of which configuration you choose, there are three steps to achieving high availability in Sisense:

1. Install Sisense on each machine in the build and query nodes. The Sisense installation includes the Sisense Web Application, ElastiCube server, and the MongoDB.
2. Replicate the MongoDB instance on the build node and distribute it to each of the query nodes. For more information, see [Replicating MongoDB](#).
3. Configure the Sisense Orchestrator Service on the build node to distribute the latest ElastiCube builds to each of the query nodes. For more information, see [Distributing ElastiCube Builds to Query Nodes](#) below.

Distributing ElastiCube Builds to Query Nodes

The Sisense Orchestrator Service is an automated service that distributes builds and balances queries across an ElastiCube Set. The Sisense Orchestrator Service is installed during the default installation of Sisense. To distribute ElastiCube builds across all query nodes, you must configure the Sisense Orchestrator Service. When an ElastiCube

is building, or receiving a build from the build node, the Sisense Orchestrator Service redirects requests to another available ElastiCube in the set.

Note: After configuring the Sisense Orchestrator Service, the directory C:\ProgramData\SiSense\PrismServer\ElastiCubeData\ and the relevant ElastiCube folders within it must be shared with permissions for 'Everyone' so Sisense can build or update the ElastiCubes in those folders.

The Sisense Orchestrator Service is defined in a JSON file called config.json in the Sisense.Orchestration.Config folder (Sisense/Sisense.Orchestration/Config/) of your Sisense Installation folder of your build node.

The config.json file has three main objects that you must define. The first object is the ElastiCube object that defines which ElastiCube is to be built on the build node and which ElastiCubes are located on query nodes. The second object is the Tasks object that determines how and when the build ElastiCube is to be distributed to the query nodes. The final object that you must define is the Schedule object that defines when an ElastiCube is to be built and distributed.

To distribute your ElastiCube builds through the Sisense Orchestrator Service, you define and save the config.json file. After saving the file, the Sisense Orchestrator Service automatically begins to build ElastiCubes in your ElastiCube Set. You can review the progress of the build in the /Sisense.Orchestration/Logs folder in a log file whose file name you define in the JSON file.

Note: Sisense recommends stopping the Sisense Orchestrator Service while importing data and reactivating it after the importing process is completed.

The objects that you must include in the config.json file and their descriptions are provided in the table below:

Key	Value
General	The General object contains two objects, LogLevel and emails that define what types of logs Sisense generates and where to send them to.
LogLevel	The type of info returned in the log. The possible values you can enter are Info, debug, and Error.
Emails	This object defines who receives an email and their email address when an email is triggered according to the events you define in the mail object.
Email	The email address where emails are to be sent depending on the value of the When key.
Name	The name of the recipient of the email.
Cubes	The cubes object contains all your ElastiCubes in your configuration and their location. As the Sisense Orchestrator Service is installed on the build node, the build cubes are local, so you must define the name of the ElastiCube. For ElastiCubes on a query node, you must define the ElastiCube name, URL of the remote server, and its directory.
Build ElastiCubes	The Build ElastiCube object defines the ElastiCubes that will be used as Build ElastiCubes nodes.
ecube	The name of the ElastiCube.

Key	Value
Query ElastiCubes	The Query ElastiCube object defines the name of the remote ElastiCube, its URL address, and directory.
ecube	The name of the ElastiCube.
URL	The URL address of the ElastiCube server. If the URL is to a secure address, the value should include the username and password credentials. For example, "ssh://username:password@10.50.1.128:/C/ecubes". See Scenario 1 for an example.
localPath	Directory of the ElastiCubes.
Tasks	The Tasks object contains a task array that defines which ElastiCube should be built, the type of build, and to which ElastiCubes the build should be distributed to. Currently, Sisense only supports one Task object.
Task	The name of the task array. Currently, you can define only one task. Defining multiple tasks in the config.json file may cause builds to fail.
Build	The build object defines the ElastiCube to be built and distributed.
Cube	The name of the cube to be built.
Queue	<p>The value of this key is the type of build Sisense should attempt and in what order. There are two possible values delimited by a comma:</p> <p>accumulate: Attempts an accumulative build.</p> <p>entire: Attempts an entire build.</p> <p>schemaChanges: Attempts to update the build only if changes were made to the schema since the previous build.</p> <p>Sisense recommends the following value: "accumulate", "entire"</p> <p>In this example, Sisense attempts to do an accumulative build first, and if it</p>

Key	Value
	fails, Sisense attempts an entire build. If the build fails for all builds, Sisense retries again according to the next scheduled build.
Distribute	<p>Determines which cubes the latest build should be distributed to. The value should be the ElastiCube name for your ElastiCubes on your query nodes. For example, ["cube1, cube2, cube3"].</p> <p>You can add multiple ElastiCubes delimited by a comma separator.</p>
Reattach	Reattach is an object that contains two objects, DeleteOldDbfarm and Cube.
DeleteOldDbfarm	A boolean value that determines if an old ElastiCube is to be deleted after it has been updated.
Cube	The name of the ElastiCube to be reattached or not.
Mail	<p>Indicates when an email alert is triggered. Email alerts can be triggered for the following scenarios:</p> <p>None: No email is ever sent.</p> <p>Build: An email is sent when a build is complete regardless if the build was successful or not.</p> <p>Fail: An email is sent when a build fails.</p>
Scheduler	The Scheduler object contains an array of tasks that define when a task is to be initiated.
Task	Defines which task to perform and the order. Currently, Sisense only supports one task.
Schedule	<p>The time that a task is to be initiated in Cron format. Some examples:</p> <ul style="list-style-type: none"> -To run a build each night at midnight, enter the value "0 0 * * *" -To run a build every hour, enter the value "0 * * * *" <p>See the full Cron format reference here.</p>

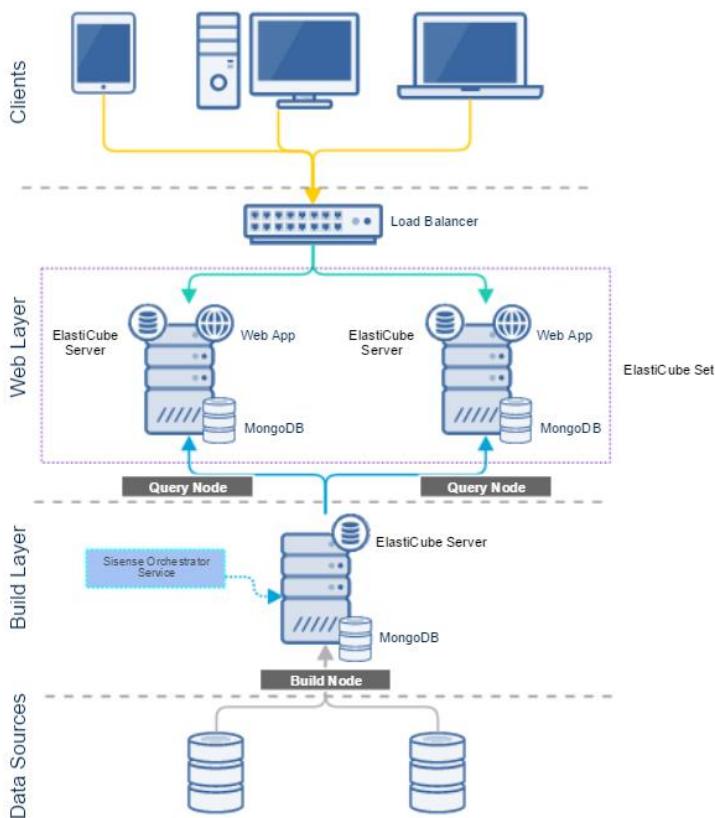
Key	Value
Enabled	A boolean value that indicates if the task is to be executed or not.

Use Cases

The following section presents two use cases and an example config.json file that demonstrates how to support these use cases. For assistance with additional HA configurations, contact your Customer Service Manager.

Scenario 1 – Single Stack

In this example, there is one build node and two query nodes. Each query node is hosted on a separate machine while the components that make up the query node are hosted on the same machine as a single application stack.



In this example, the build cube is called "BuildTest1". This ElastiCube is distributed to two ElastiCube servers defined in the cube1 and cube2 objects. The build cube is distributed first to cube1 and then to cube2 as defined in the tasks object. The first time the Sisense Orchestrator Service tries to build the BuildTest1 ElastiCube, it attempts an accumulative build and if that fails, an entire build. After each build is replicated, the previous version of the ElastiCube is deleted as specified in the reattach object. This task takes place hourly. If the distribution fails, Sisense sends an email to johndoe@sisense.com with a debug log attached.

```
{  
  "general": {  
    "logLevel": "INFO",  
    "emails": [ {  
      "email": "johndoe@sisense.com",  
      "name": "John Doe"  
    } ] },  
  "cubes": {  
    "localCubeBuildTest1": {  
      "ecube": "BuildTest1"  
    },  
    "cube1": {  
      "ecube": "BuildTest1",  
      "url": "\\\\10.49.1.68\\\\ecubes",  
      "localPath": "d:\\\\ecubes"  
    },  
    "cube2": {  
      "ecube": "BuildTest1",  
      "url": "\\\\10.70.0.32\\\\ecubes",  
      "localPath": "e:\\\\ecubes"  
    },  
    "cube3": {  
      "ecube": "BuildTest1",  
      "url": "ssh://administrator:Password@10.70.1.133:/C/ecubes",  
      "localPath": "e:\\\\ecubes"  
    }  
  }  
}
```

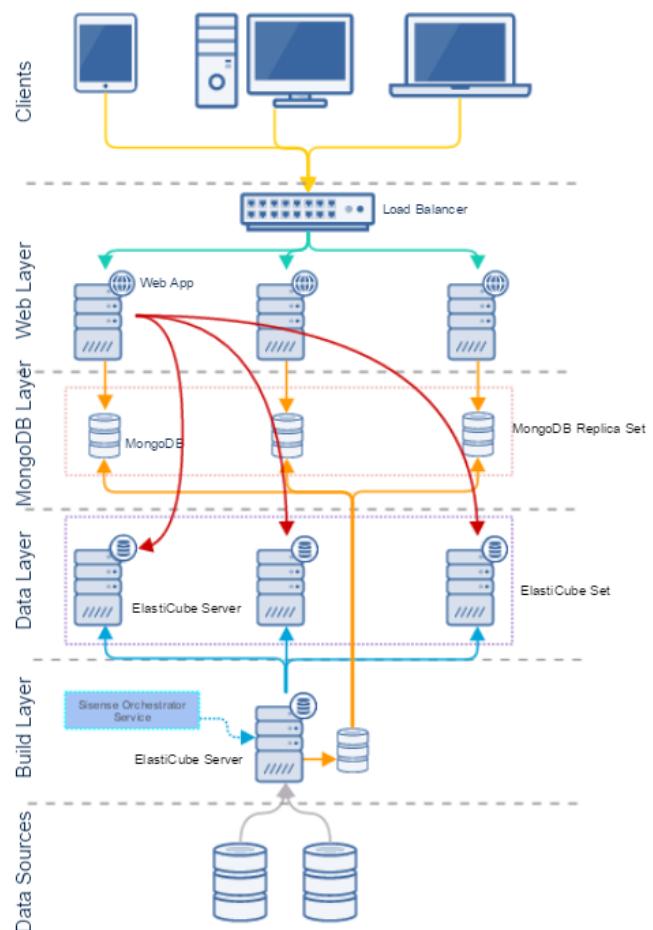
```

    }
},
"tasks": {
"task1": [
{
"build": {
"cube": ["localCubeBuildTest1"],
"queue": ["accumulate","entire"] }
},
{
"distribute": ["cube1","cube2","cube3"] },
{
"reattach": {
"deleteOldDbfarm":true,
"cube": ["cube1","cube2","cube3"] }
},
{
"mail":"build"
}] },
"scheduler":[{
"task": "task1",
"schedule": "0 * * * *",
"enabled": true
}] }

```

Scenario 2 – Distributed Stack

In this example, there are three query nodes in which each component is located in a different remote location. This scenario, while costly to implement, demonstrates a distributed application stack configuration where each component is replicated and located on a separate server independent of the other components. If any component fails, the rest of the architecture remains intact. For example, the web server is on one server, the ElastiCubes are stored on another server, and the MongoDB is located on another server.



The build cube in this example is called "BuildTest1". This ElastiCube is distributed to three remote ElastiCube servers defined in the cube1, cube2, and cube3 objects. The build cube is distributed first to cube1, then cube2, and then cube3 as defined in the tasks object. The URLs are standard URLs for cube1 and cube2 while the URL for cube3 is secured through SSH. When the URL is secured, you must provide the URL and the required credentials to access it.

The first time the Sisense Orchestrator Service tries to build the BuildTest1 ElastiCube, it attempts an accumulative build and if that fails, an entire build. If both builds fail, Sisense attempts to update the schema only. After each build is replicated, the previous version of the ElastiCube is deleted as specified in the reattach object. This task

takes place At 14:15 on the 1st of every month. After a build is successful, Sisense sends an email to johndoe@sisense.com.

```
{
  "general": {
    "logLevel": "DEBUG",
    "emails": [
      {
        "email": "johndoe@sisense.com",
        "name": "John Doe"
      }
    ],
    "cubes": {
      "localCubeBuildTest1": {
        "ecube": "BuildTest1"
      },
      "cube1": {
        "ecube": "BuildTest1",
        "url": "\\\\10.49.1.68\\ecubes",
        "localPath": "d:\\ecubes"
      },
      "cube2": {
        "ecube": "BuildTest1",
        "url": "\\\\10.70.0.32\\ecubes",
        "localPath": "e:\\ecubes"
      },
      "cube3": {
        "ecube": "BuildTest1",
        "url": "ssh://administrator:Password@10.70.1.133:/C/ecubes",
        "localPath": "e:\\ecubes"
      }
    },
    "tasks": {
      "task1": [
        {
          "build": {
            "cube": ["localCubeBuildTest1"],
            "queue": ["accumulate", "entire", "schemachanges"]
          }
        }
      ]
    }
}
```

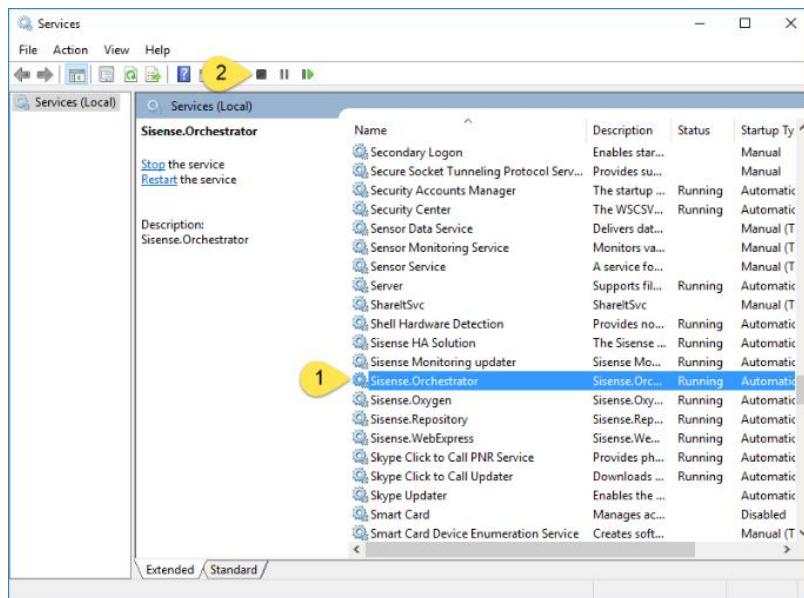
```
},
{
  "distribute": ["cube1", "cube2", "cube3"] },
{
  "reattach": {
    "deleteOldDbfarm": true,
    "cube": ["cube1", "cube2", "cube3"] }
},
{
  "mail": "fail"
}] },
"scheduler": [
  "task": "task1",
  "schedule": "15 14 1 * *",
  "enabled": true
}] }
```

Add Query Nodes

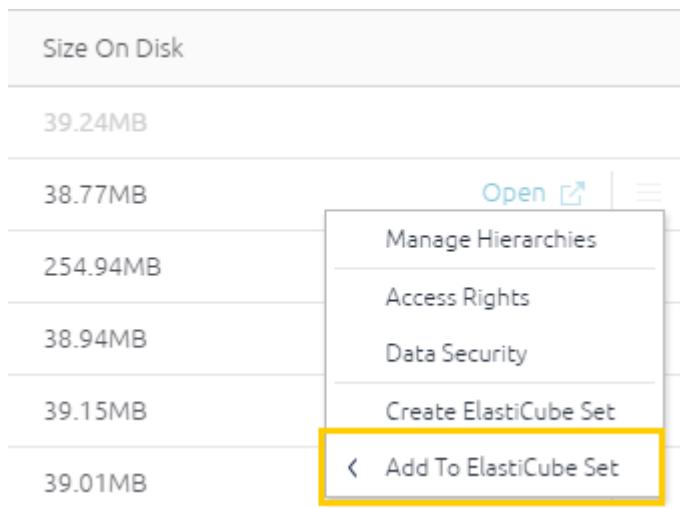
In some configurations, replicating components can improve performance by scaling out and preventing any potential bottlenecks. Each component you replicate must be added to the ElastiCube Set and the config.json file. You can continue to add query nodes according to your requirement so long as the nodes are included in the config.json file.

To add a query node:

1. Stop the Sisense Orchestrator Service. Open Windows Services, select Sisense.Orchestrator, and click .



2. In the Sisense Web Application, click **Admin** in the upper-right corner and select the **ELASTICUBES** tab on the left.
3. Hover over the ElastiCube you want to add to the ElastiCube Set and click on the menu that appears. Click **Add to ElastiCube Set** and select the set that you want to add the ElastiCube to.



OR



In the ElastiCube Set table, click and select the ElastiCubes to be added to the set. Click **Save** after you have selected the relevant ElastiCubes.

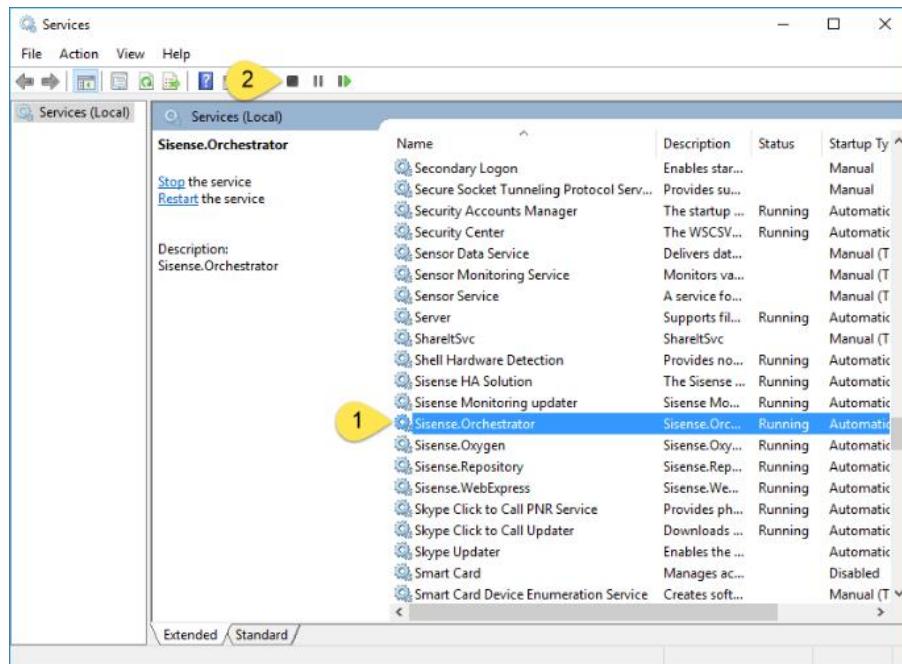
4. Edit the config.json located at Sisense/Sisense.Orchestration/Config/ and add the new ElastiCube to the Cubes object.
5. Save the config.json file.
6. Restart the Sisense Orchestrator Service.

Remove Query Nodes

You can remove query nodes by removing them from the ElastiCube Set and the config.json file.

To remove a query node:

1. Stop the Sisense Orchestrator Service. Open Windows Services, select Sisense.Orchestrator, and click .



2. In the Sisense Web Application, click **Admin** in the upper-right corner and select the **ELASTICUBES** tab on the left.
3. In the ElastiCube Set table, click  and clear the checkboxes of the ElastiCubes to be removed. Click **Save** after you have selected the relevant ElastiCubes.
4. Edit the config.json located at `Sisense/Sisense.Orchestration/Config/` and delete the relevant ElastiCubes from the Cubes object.
5. Save the config.json file.
6. Restart the Sisense Orchestrator Service.

Limitations

If your user queries a query node (Query Node 1), then the build node begins to replace a build on that node (Query Node 1), there are two scenarios in which the data may be inconsistent.

1. When generating PDFs or Images, the widgets may not match the dashboard depending.
2. In a Pivot widget, the subtotal value may not match the expected value.

This occurs because the widgets exported to PDF or subtotal values inside Pivot widgets run new queries against the node. If the data was originally taken from query node 1, then subsequently queries were run against another query node (Query Node 2), the data may have changed leading to inconsistent results.

Replicating MongoDB

When you install Sisense, a MongoDB instance is installed locally on your machine that contains metadata for ElastiCubes, dashboards, and users. The goal of the replica set is to ensure that each MongoDB has the same information about ElastiCubes, dashboards, and filters. If a MongoDB fails on one query server, the next available MongoDB can replace it. To maintain consistency throughout your configuration and support high availability, each MongoDB instance must be identical with the other MongoDB instances in your configuration.

Each MongoDB instance is part of a replica set. A replica set is a group of MongoDB instances that have the same Sisense data. There are two types of nodes within a replica set. The primary node, which supports write operations and the secondary nodes, which replicate the primary MongoDB's operations log and syncs the data to reflect that of the primary MongoDB.

The primary and secondary MongoDB nodes can be hosted on the build or query nodes in a high availability configuration or on their own servers.

There are four steps to support high availability for the MongoDB in Sisense:

1. Preparing to set up replica set.
2. Setting up replica nodes.
3. Creating a replica set.
4. Connecting Sisense to the replica set.

Prerequisites

1. Servers in which port 27018 is open.

2. RoboMongo installed on a machine that has access to those servers on port 27018 (any node can be on one of the servers).
3. MongoDB installed on the machine. By default, this is installed when Sisense is installed. If Sisense is not installed, download MongoDB version 2.4.5 zip on the secondary servers from [here](#).

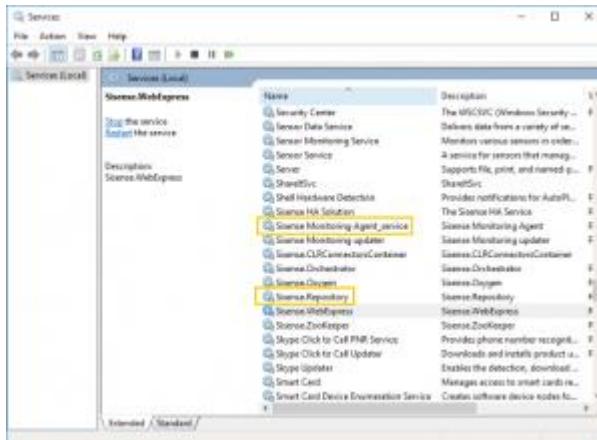
Preparing to Set up a Replica Set

For each machine that you have installed Sisense, you must stop the Sisense-related services from running to prepare the main MongoDB instance that will be replicated to the other secondary nodes.

This procedure assumes your primary node has Sisense installed.

To prepare each server node in your replica set:

1. Log in to the machine that hosts Sisense.
2. Depending on your Sisense installation:
 - ▶ **Full IIS:** Open the IIS Manager and stop the website **SisenseWeb**.
 - ▶ **IIS Express:** Open Windows Services and stop the service **Sisense.WebExpress**.
3. In Windows Services, stop the service named **Monitoring Agent_service** and the service **Sisense.Repository**.



4. For the server that will be the primary node, backup the Sisense MongoDB instance by copying this directory and saving to another location:

C:\ProgramData\Sisense\PrismWeb\DB\Prod

You can delete the backup copy after you have verified that everything is running as expected.

5. For each secondary server where Sisense is installed, backup the contents of the directory below including the MongoDB instance.

C:\ProgramData\Sisense\PrismWeb\DB\Prod

Setting Up the Replica Nodes

For each node in your replica set, you must have a MongoDB configuration file. The configuration file contains information about the MongoDB such as its open port and the name of the replica set the MongoDB instance belongs to.

When Sisense is installed on the node, this configuration file is created for you. You must add the line replSet=replica_set to the configuration file. This line is used to identify the node as part of the replica set.

For nodes in which Sisense is not installed, you must create this configuration file as described in the procedure below.

To set up a node:

When Sisense is installed:

1. In the directory, 'C:\Program Files\Sisense\Infra\MongoDB\MongoDB.conf', backup the MongoDB configuration.

2. In the original MongoDB.conf file, add the line **rep1Set = replica_set**.

```
dbpath=C:\ProgramData\Sisense\PrismWeb\DB\Prod  
logpath=C:\ProgramData\Sisense\PrismWeb\DB\Logs\sisense  
Repository.log  
port=27018  
noauth=true  
smallfiles=true  
rep1Set=replica_set
```

3. In Windows Services, restart the **Sisense.Repository** service.

When Sisense is not installed:

1. If Sisense is not installed, MongoDB must be downloaded as described in the Prerequisites. Extract the MongoDB zip to a directory called c:\Sisense.

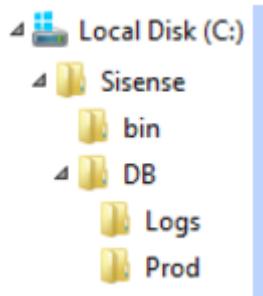
After extracting the file you should see an executable mongod file at c:\Sisense\bin\mongod.exe.

2. In the directory c:\Sisense\bin\, create a file named 'MongoDB.conf' as shown below:

```
dbpath=C:\Sisense\DB\Prod  
logpath=C:\Sisense\DB\Logs\PrismWeb.log  
port=27018  
noauth=true  
smallfiles=true  
rep1Set = replica_set
```

3. In the directory c:\Sisense\, create the following directories as listed below: **DB** with two subdirectories

named **Log** and **Prod** within it.



4. In the Windows Command Prompt, run the following command to install MongoDB as a Windows service:

```
C:\Sisense\bin\mongod.exe --config  
C:\Sisense\MongoDB.conf --install
```

5. In Windows Services, start the MongoDB service.
6. Verify that the MongoDB service is running and that its startup type is automatic.

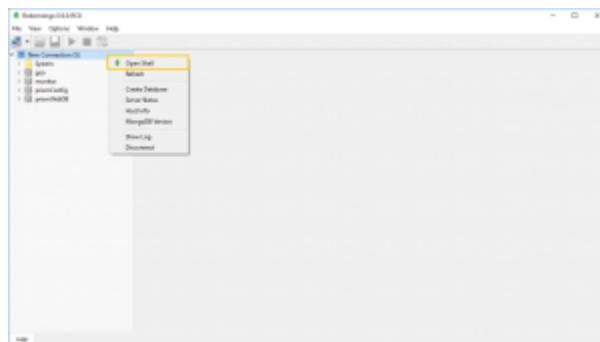
Name	Description	Status	Startup Type
Mongo DB	Mongo DB ...	Running	Automatic

Creating a Replica Set

After you have prepared the MongoDB configuration file for each of your nodes, you must create a replica set with these nodes. With a replica set, each member has the same Sisense data but is otherwise independent. If the primary becomes unavailable, the replica set holds an election to select a new primary from one of the secondary nodes.

To create replica set:

1. Open Robomongo and select **Open Shell** to connect to the primary node.



2. In the new shell, run the command **rs.initiate()** to create a replica set.
3. Run the following commands:

```
cfg = rs.conf()
cfg.members[0].host = "XX.XX.XX.XX:27018" (Enter your
IP address of the primary node and the MongoDB port, by
default 27018).
rs.reconfig(cfg)
```

4. Run the command **rs.add("XX.XX.XX.XX:27018")**, where the XX is the IP address of your secondary node. Run this command to add the unique IP address for each secondary node. For example:

```
rs.initiate()
cfg = rs.conf()
cfg.members[0].host = "10.50.1.199:27018"
rs.reconfig(cfg)

rs.add("10.20.1.196:27018")
rs.add("10.20.1.197:27018")
```

5. Run the command **rs.status()** to verify that the replica set was configured properly. Run this command every couple of minutes to check the progress until stateStr of the secondaries

service displays **SECONDARY**.

 [1]	{ 12 fields }
 _id	1
 name	10.50.1.23:27018
 health	1
 state	2
 stateStr	SECONDARY
 uptime	166202
 optime	2016-05-17T08:51:25.000+0000
 optimeDate	2016-05-17 08:51:25.000Z
 lastHeartbeat	2016-05-17 08:51:40.000Z
 lastHeartbeatRecv	2016-05-17 08:51:42.000Z
 pingMs	1
 syncingTo	10.50.1.82:27018

Connecting Sisense to the Replica Set

After you have configured the replica set, you must connect it to Sisense to support your high availability configuration.

Repeat the procedure below for each machine with Sisense installed.

To connect Sisense to a replica set:

6. In the directory "C:\Program

Files\Sisense\PrismWeb\App_Data\Configurations\" backup the file db.config.

7. In the original db.config file, modify the line:

```
<add name="MongoProvider"
      connectionString="MongoDB://localhost:27018/?safe=true"
      />
```

To read:

```
<add name="MongoProvider"
      connectionString="MongoDB://XX.XX.X.XX:27018,XX.XX.X.XX
      :27018,XX.XX.X.XX:27018/?replicaset=replica_set&readPreference=primaryPreferred&safe=true"/></connectionStrings>
```

Where XX.XX.XX.XX:27018 is the IP address and port for each MongoDB instance in your replica set separated by a comma.

8. In this directory "C:\Program

Files\Sisense\PrismWeb\vnnext\config", backup the file default.yaml

9. In the original default.yaml file, at the bottom add the following:

```
replica_set:
  name: 'replica_set'
  servers:
    - host: XX.XX.X.XX
      port: 27018
      dbpath: 'C:\ProgramData\Sisense\PrismWeb\DB\Prod'
    - host: XX.XX.X.XX
      port: 27018
      dbpath: 'C:\Sisense\DB\Prod'
    - host: XX.XX.X.XX
      port: 27018
      dbpath: 'C:\Sisense\DB\Prod'
```

The dbpath line should be

'C:\ProgramData\Sisense\PrismWeb\DB\Prod' on nodes in which Sisense is installed and 'C:/Sisense/DB/Prod' when Sisense is not installed.

10. In the directory "C:\Program Files\Sisense\Monitoring

Agent\logstash\bin\", backup the file InfraAgent.exe.config.

11. In the original InfraAgent.exe.config file, change the value of the run_shipper_key to **false**.

```
<add key="queue_name" value="MonitorQueue" />
<add key="run_shipper_outer" value="false" />
<add key="sisenseWebsiteName" value="SisenseWeb" />
```

12. In the directory "C:\Program Files\Sisense\Monitoring

Agent\logstash\bin\", backup the file InfraAgent.exe.config.

13. In the original InfraAgent.exe.config file, change the value of the mongo_url key to the IP address of your primary node.

```
<add key="queryServerPort" value="41112" />
<add key="mongo_url" value="mongodb://localhost" />
<add key="mongo_port" value="27018" />
```

14. In the directory "C:\Program Files\Sisense\Monitoring

Agent\logstash\bin\configs\", backup the file
shipperInner.conf.

15. In the original shipperInner.conf file, change the value of the "**monitor**" uri to the IP Address of the primary node.

```
output {
    mongo{ collection => monitoring_logstash database => "Sisense" uri => "mongodb://localhost:27018/testdb" }
}
```

16. In Windows Services, start the service

named **Sisense.Monitoring Agent_service** and the
service **Sisense.Repository**.

17. Depending on your Sisense installation,

- ▶ **Full IIS:** Open the IIS Manager and start the website **SisenseWeb**.
- ▶ **IIS Express:** Open Windows Services and start the service **Sisense.WebExpress**.

Routing Queries in ElastiCube Sets

When a user queries an ElastiCube that is part of an ElastiCube Set, Sisense routes the query to a single ElastiCube with the latest ElastiCube build. By default, ElastiCube Sets operate in active-passive mode. In active-passive mode, one ElastiCube, by default the ElastiCube with the latest build, handles incoming queries. This ElastiCube is active while the remaining ElastiCubes are on standby in passive mode.

Sisense also supports active-active mode where queries are spread across available ElastiCubes within an ElastiCube Set. In active-active mode, each ElastiCube that is not currently building can handle queries.

You can modify how queries are routed to ElastiCubes within ElastiCube Sets by selecting one of the following routing strategies:

Single: (Default). ElastiCubes operate in active-passive mode where the ElastiCube with the freshest data receives all the queries while the remaining ElastiCubes are on standby.

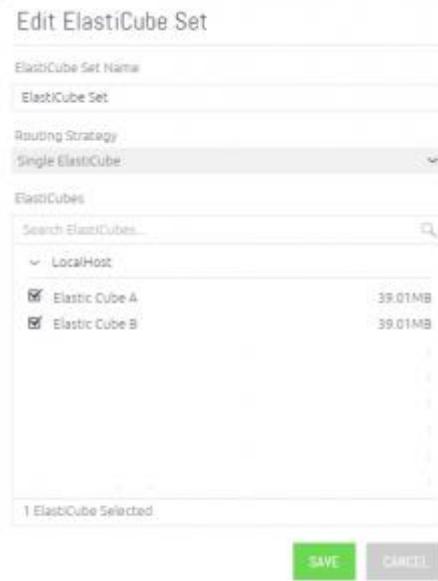
Multiple ElastiCubes – Latest Data: ElastiCubes operate in active-active mode where queries are spread across all ElastiCubes with the latest data.

Multiple ElastiCubes – Optimized Performance: ElastiCubes operate in active-active mode where queries are spread across the largest group of ElastiCubes with identical data.

To set a query routing strategy:

1. Click **Admin** in the upper-right corner and select the **ELASTICUBES** tab on the left.

2. In the ElastiCube Set table, click  for the relevant ElastiCube Set.



The screenshot shows the 'Edit ElastiCube Set' dialog box. It has fields for 'ElastiCube Set Name' (set to 'ElastiCube Set') and 'Routing Strategy' (set to 'Single ElastiCube'). Below these, there's a list titled 'ElastiCubes' with a search bar. Under 'LocalHost', two cubes are listed: 'Elastic Cube A' and 'Elastic Cube B', both with a size of '39.01MB'. At the bottom, a message says '1 ElastiCube Selected.' followed by 'SAVE' and 'CANCEL' buttons.

3. From the Routing Strategy list, select the relevant strategy.
4. Click **Save**.

Security

Sisense Security Architecture

Sisense is built around a robust and flexible security architecture that is both comprehensive and intuitive. This architecture has been designed to ensure security processes are enforced while scaling to enterprise deployments of Sisense. This includes the ability to secure dashboards and data as well as implement custom security requirements that suit your organization. This section provides a general overview of the main security features.

Security is based around three levels associated with sets of security features. The diagram below maps this security architecture on a system, data and object level.



System Level Security

System-level security encompasses security features for role-based settings and integration options. This includes user and server management, connection to an active directory, Single Sign-On (SSO) implementation, and use of the security REST API.

User Management

You can assign three primary roles to Sisense users:

- ▶ **Viewers:** Can view dashboards
- ▶ **Designers:** Can create and edit dashboards

- ▶ **Administrators:** Can create users and user groups, set up Active Directory, and more.

These roles can be defined on either a user or group level to determine sharing, access and security.

ElastiCube Server Access Rights

You can assign access rights to different ElastiCube servers for individual users, groups or to everyone. These settings allow the management of different environments such as a testing and production server, or servers for specific projects or departments. See also ElastiCube Server and ElastiCube Security.

Active Directory

Connect existing users and groups from your organization's Active Directory to define security and sharing properties and reduce deployment time. This removes 'password fatigue' as users can rely on existing credentials while organizational policies around security credentials such as updates can be enforced. See also Integrating Active Directory.

Single Sign-On (SSO)

SSO facilitates seamless integration between Sisense and other systems in your organization while offering standardization of authentication policies across your organization. This can improve user productivity by avoiding password fatigue and reduce support overhead. See also Configuring SSO.

REST API

The Security REST API provide access to parameters to integrate and automate restrictions and access control based on existing settings and standards. Specify access rights and security to dashboards, ElastiCubes and data. Manage users via the API to create, edit and assign new users or groups. [Click here](#) to visit the API documentation site.

Object Level Security

Object security defines access rights for different users and groups to various components within Sisense. The two main objects are dashboards and ElastiCubes.

Dashboards

You can share dashboards on either a user or group level. The sharing options include the configuration of access rights for all users as well as whether users defined as designers may edit the dashboard. The sharing options also include subscription settings that define which users and groups will receive email reports. See also [Sharing Dashboards](#).

ElastiCubes

You can define access rights to different ElastiCubes on a user or group level. This enables flexibility to create ElastiCubes for specific user or group needs while offering strict access control. See also [Assigning Access Rights to an ElastiCube](#).

Data Level Security

Data access must provide data to people only to the extent that they need to complete their jobs. Data Level Security provides the

necessary control to enforce varying degrees of data visibility and access to support the separation of duties. A single dashboard can be shared with many users, but each viewer sees only data relevant to their needs. This reduces both development time and provides for security.

Security on the Row Level

You can grant user and group permissions to specific rows in the data. For each ElastiCube, you can apply multiple rules to enforce granular access control.

Row Level Defaults

Control which data is accessible for users or groups that do not have explicit security rules. For example, enable new employees to access a restricted data set until they are added to relevant groups. You can set defaults to include everything, nothing or view based on a security rule.

See more in Data Security below.

Security Levels

Sisense provides three levels of security:

- ▶ ElastiCube Server Security
- ▶ ElastiCube Security
- ▶ Data Security

ElastiCube Server and ElastiCube Security

What is ElastiCube Server and ElastiCube Security?

Sisense enables you to define access rights to control which users can access which ElastiCubes, whether they are creating new dashboards or trying to access shared dashboards.

ElastiCube Server Security – Use Case Example

You may already have access to an ElastiCube Server named *Production*. As an administrator, you want to create a *Staging* environment with its own ElastiCube Server; and you do not want other users to be exposed to it. In this case, you can set the ElastiCube Server rights to allow only you to access it.

ElastiCube Security – Use Case Example

You may have an ElastiCube named **Marketing** and only want the CEO and Marketing team to have access to it. You can grant rights only to them using ElastiCube Security, thus denying anyone else access.

How Does ElastiCube Security Work?

- ▶ You can define that users/user groups have access to either:
 - ▶ An entire ElastiCube Server, including all the ElastiCubes on the ElastiCube Server.
and/or
 - ▶ Only to specific ElastiCubes.
- ▶ By default, **all** users are allowed to access the data of all ElastiCube Servers and ElastiCubes. Once you start assigning users/user groups access rights to an ElastiCube Server/ElastiCube, then **only** the users/user groups that you assign can access that ElastiCube Server/ElastiCube.

- ▶ Access to ElastiCube Servers/ElastiCubes is by *Grant* rules.
There are no *Deny* rule options.
- ▶ When no access rights have been defined for a specific ElastiCube, then that ElastiCube inherits its access rights from the ElastiCube Server settings.
- ▶ When a user attempts to access a Dashboard using a direct link and that dashboard is based on an ElastiCube to which that user does not have access rights, a security message is displayed.
- ▶ When access rights are defined for a specific ElastiCube, then those access rights override the ElastiCube Server access rights. This means that a user **can** access an ElastiCube to which he/she is assigned access, even if that user has not been assigned access rights to the ElastiCube server on which that ElastiCube is stored.

Assigning Access Rights to an ElastiCube Server

By default, when you add an ElastiCube server (as described in Manage ElastiCubes), **all** users are allowed to access it.

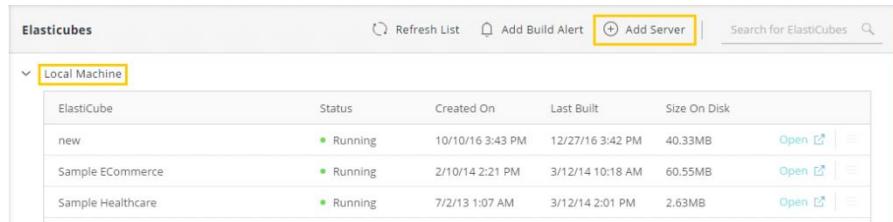
A user that has access to an ElastiCube server can access all the ElastiCubes on it.

Once you start assigning users/user groups access rights to the ElastiCube server, then **only** the users/user groups that you assign can access the data of the ElastiCube Server.

To assign access rights to access an ElastiCube Server:

1. Click **ADMIN** in the upper right corner and select the **ELASTICUBES** tab in the menu.

2. To define that an ElastiCube Server can only be accessed by specific users/user groups, hover over the list of ElastiCubes under the server name, and click **Access Rights** next to the server name.



Elasticubes				
↻ Refresh List 🔔 Add Build Alert ➕ Add Server Search for Elasticubes				
▼ Local Machine				
ElastiCube	Status	Created On	Last Built	Size On Disk
new	Running	10/10/16 3:43 PM	12/27/16 3:42 PM	40.33MB
Sample ECommerce	Running	2/10/14 2:21 PM	3/12/14 10:18 AM	60.55MB
Sample Healthcare	Running	7/2/13 1:07 AM	3/12/14 2:01 PM	2.63MB

3. Enter the users/user groups that you want to give access to this ElastiCube Server.
4. Click **SAVE**.

Assigning Access Rights to an ElastiCube

By default, **all** users are allowed to access each ElastiCube.

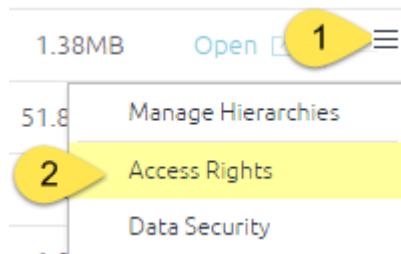
You can define which users are allowed to access each specific ElastiCube. You can assign a user access to an ElastiCube even when the user does not have access to the entire ElastiCube server.

Once you start assigning specific users/user groups access rights to this ElastiCube, then **only** the users/user groups that you assign can access the ElastiCube.

To assign access rights to an ElastiCube:

1. Click **ADMIN** in the upper right corner and select the **ELASTICUBES** tab in the menu.
2. In the ElastiCube's menu on the right, click **Access Rights**. By default this list is empty, meaning that all users have access

rights to this ElastiCube.



Access Rights: 54.75.231.25

Search for users and groups

A	B	Admins
C	D	2 admin users currently
Demo demo@sisense.com		

SAVE **CANCEL**

3. Enter the users/user groups allowed to access this ElastiCube.
If for example you add your own name, then only you will have access to the ElastiCube.
4. Click **SAVE**.

Data Security

Data security in Sisense can be divided into two types, data encryption and data access. Data communication is related to how data is secured by Sisense while being imported into Sisense and written on your server's disk.

When data is imported into Sisense, the protocol used depends on the protocols supported by the data source. Sisense supports

importing data over SSL, if the source supports it. Sisense supports SSL for data Retrieval, for example, when viewing data in dashboards.

Configuration data, such as account credentials and authorization profiles, are encrypted prior to being written to the disk. The encryption technology used by Sisense includes:

1. SHA-256
2. TripleDES
3. AES-256

For data at rest, Sisense supports OS based disk encryption, Windows file system encryption Transparent Data Encryption (TDE). When using Windows transparent encryption, the key pair (private/public) is bound to the user identity..

The second type of data security is data access. This type of data security refers to who can access your data after its imported into Sisense and displayed in a dashboard.

What is Data Access Security?

Sisense enables you to define Data Security Rules that control which users can access which portions of the raw data in an ElastiCube, down to row granularity. For example:

Each widget only shows the data permitted by the Data Security Rules that apply, including totals, averages and so on.

The data browsers used while building dashboards and widgets only show the data permitted by the Data Security Rules that apply.

You can define a single dashboard that automatically displays different results for each user (or user group), based on the rows that user is permitted to see.

Use Case Example

A Sales Order table has a column representing the salesperson that closed a deal.

You created a quarterly performance dashboard for your salespeople, but want each of them to see only their own data.

You do not want any of them to be exposed to data that represents the performance of others.

How Does Data Access Security Work?

Each ElastiCube contains tables and each table contains fields. A Data Security Rule defines that a specific user can only see any data of an entire row of a table, if a specific field in that row has a specific value(s).

For example, in a Sales widget a salesperson (for example, Dan) will only see the sales amounts from the rows of a Sales ElastiCube whose **Salesperson** field contains the value **Dan** (rows 1 and 4).

Sales Table

#	Salesperson	Product	Amount
1	Dan	HD-TV	\$100
2	Matthew	TV	\$300
3	Amber	Media Center	\$700
4	Dan	Player	\$200
5	Matthew	Air Conditioner	\$600

Dan will not see any part of a row in the ElastiCube that does not contain the value **Dan** in the **Salesperson** field, nor will any amounts from this row be included in totals.

Note: The entire row of data is not seen by the relevant user even when the field to which the rule applies does not appear in the widget.

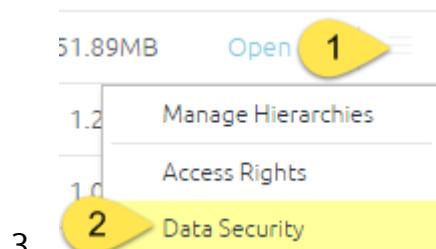
If a widget that shows the amount spent per product is shared with Dan , then he will only see **HD-TV** and **Player** and the sales total will be **300\$**.

Defining Data Access Security for an ElastiCube

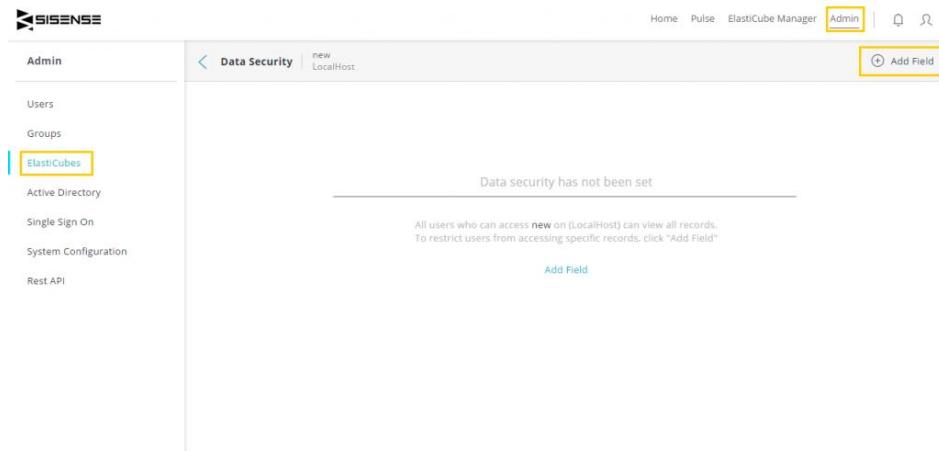
Each Data Security Rule applies to a specific field in an ElastiCube and to specific user(s)/user group(s). It enables you to define the values that must be contained in a specific field to enable that entire row of data to be available to a user.

To access Data Security:

1. Click **ADMIN** in the upper right corner and select the **ELASTICUBES** tab in the menu.
2. In the ElastiCube's menu on the right, click **Data Security**.

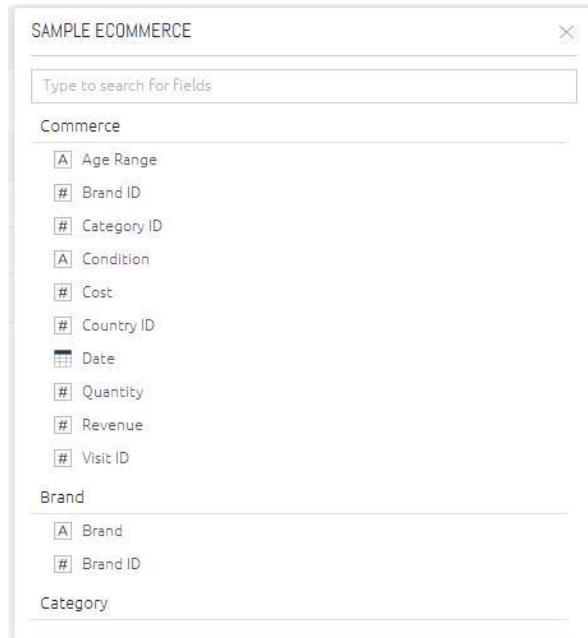


3. If no data security rules have yet been defined for this ElastiCube, then the following message is displayed:



The screenshot shows the Sisense Admin interface. On the left, there's a sidebar with 'Admin' at the top, followed by 'Users', 'Groups', and 'ElastiCubes' (which is highlighted with a yellow box). Below these are 'Active Directory', 'Single Sign On', 'System Configuration', and 'Rest API'. The main content area is titled 'Data Security' and shows a message: 'Data security has not been set. All users who can access new on (localhost) can view all records. To restrict users from accessing specific records, click "Add Field".' At the bottom right of this area, the 'Add Field' button is also highlighted with a yellow box.

5. Click the **ADD FIELD** button to display a list of the fields in this ElastiCube.



The screenshot shows a modal window titled 'SAMPLE ECOMMERCE'. At the top is a search bar with the placeholder 'Type to search for fields'. Below the search bar, there are three sections: 'Commerce' containing 'Age Range', 'Brand ID', 'Category ID', 'Condition', 'Cost', 'Country ID', 'Date', 'Quantity', 'Revenue', and 'Visit ID'; 'Brand' containing 'Brand' and 'Brand ID'; and 'Category' which is currently empty.

6. Select a field. For example, **Brand**. The following window is then displayed in which you can define rules.

Note: You cannot select date type fields.

[**< DATA SECURITY**](#) | MOVIES
LOCALHOST

▼ Series Name (Table: 5d02ccc4-d761-464d-878b-b0bc59007588_Data.csv)

Restricted Users/ Groups	Accessible Values
+ Add Restriction	
Everyone	Nothing
	evel Data from database: World Dev... Everything Nothing

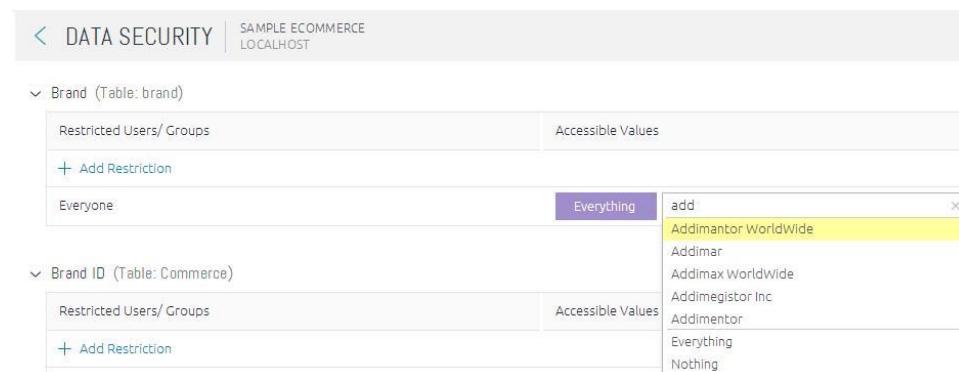
The left side of this table enables you to define which users/user groups can access this data. Click **+ Add Restriction** and start typing into the **Restricted User/Groups** field to get a drop-down list. Add as many users/user groups, as necessary.

[**< DATA SECURITY**](#) | SAMPLE ECOMMERCE
54.75.231.25

▼ Brand (Table: brand)

Restricted Users/ Groups	Accessible Values
a	x
A B Admins C D 2 admin users currently	Everything
B Bo@Sisense.Com boazf@sisense.com	
A B Marketing C D 1 members	

The right side of this window enables you to define which values the specified users/user groups are permitted to see. Start typing into the **Values** field to get a drop-down list.



The screenshot shows the Sisense Data Security interface. It displays two tables: 'Brand' (Table: brand) and 'Brand ID' (Table: Commerce). In the 'Brand' table, under the 'Accessible Values' column for the 'Everyone' row, a dropdown menu is open, showing options like 'add', 'Addimantor WorldWide' (which is highlighted in yellow), 'Addimair', 'Addimax Worldwide', 'Addimegistor Inc', 'Addimentor', 'Everything', and 'Nothing'. A similar dropdown menu is visible for the 'Brand ID' table.

In this example, **Everyone** can only view rows that have **Addimantor Worldwide** in the **Brand** field.

Multiple values can be selected.

The value of numeric type fields must be typed into this field, as no auto-complete option appears for numeric type fields.

Alternatively, you can select:

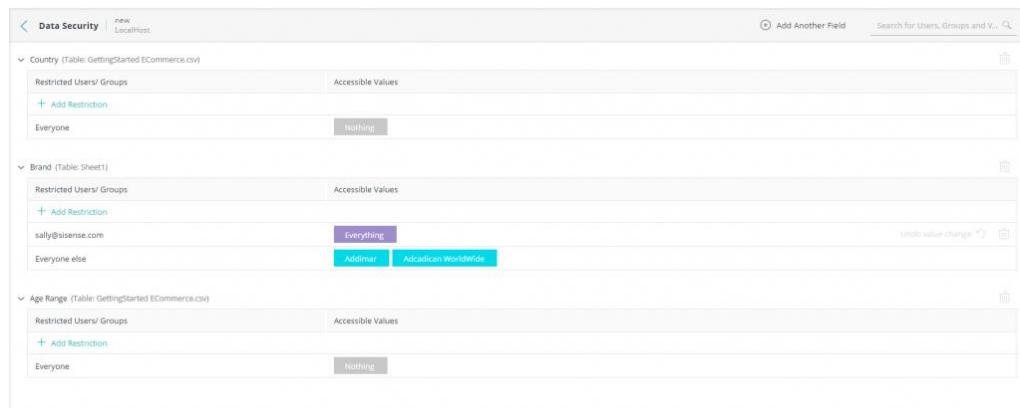
- ▶ **Everything:** To specify that the selected users/user groups can see this data no matter what the value is in this field.
- ▶ **Nothing:** To specify that the selected users/user groups cannot see this data no matter what the value is in this field.

For example, you can define that the following Users/User Groups must have the following values in the Product Category column to enable them to see their data row in a widget.

#	User/User Group	Product Category
1	Management	Everything
2	Bob	Apple Mac Desktops
3	Don	Calculators, Camera Flashes
4	Everyone else	Nothing

This means that management can see the data of all Product Categories, Don can only see the data of Calculators and Camera

Flashes, Bob can only see the data of Apple Mac Desktops, and Everyone else won't see anything.



How Does Data Level Security Work for Tables with Relationships?

Tables in an ElastiCube may have a relationship between them.

As described above, each widget only shows any data of an entire row of a table, if a specific field in that row has a specific value.

In addition, a widget may further restrict the data shown to a specific user when a rule is defined for a table that has a relationship to a table that has a field in the widget.

This means that a widget only shows the data permitted by the **combined** Data Security Rules assigned to all the tables that have **any field in the widget**.

As described above, the entire data row is restricted even when the field to which the rule applies does not appear in the widget. The entire row of data is also restricted even when the field of the relationship between the two tables does not appear in the widget.

Use Case Example –Expanding Upon the Example Above

- ▶ The Sales table has a column that has a relationship with a Deal Contacts table that holds the contacts that were involved in each deal.
- ▶ You created a Deal Contacts widget for your salespersons.
- ▶ As described in the example above, the Sales table has a Data Security Rule that maps each user to his/her matching field value, so that each sales person only sees their own data.
- ▶ Even though the Deal Contacts table does not have any Data Security Rules defined for it, the Deal Contacts widget only enables each sales person to see the contacts associated with their own sales, because of the Data Security Rule assigned to the Sales table.

White Labeling Sisense

Sisense enables you to rebrand (white label) the Sisense Web Application into your native application or site by using the Sisense REST API to replace existing icons and images with your own.

In addition, you can rebrand the automated emails Sisense sends to match your company's branding.

White labeling must be enabled in your license. To enable this feature, please contact your Sisense representative or open a request through our [Help Center](#).

Rebranding Sisense

You can rebrand the Sisense Web Application and the automated emails sent by Sisense through Sisense's REST API.

There are two steps to rebranding the Sisense Web Application. The first step is to place your branded content into the Sisense *branding* directory where Sisense is installed. For a list of the files you can rebrand and their specific dimensions, see White Labeling Configuration Options.

The second step is to access the the Sisense REST API, where you can embed your content directly into Sisense and modify text displayed to customers.

To rebrand the Sisense Web Application:

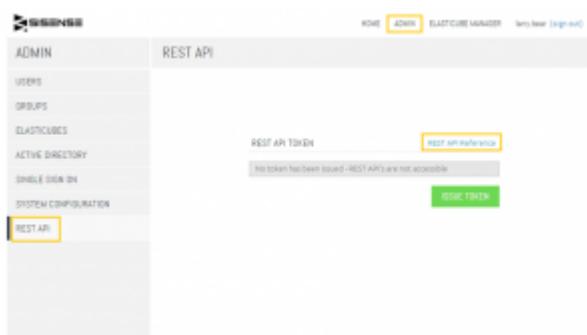
1. Create a subfolder with your brand name in this directory (on the machine where Sisense is installed): "C:\Program Files\Sisense\PrismWeb\Resources\<YourBrand>".

Put your branded files in this directory. The following files are supported:

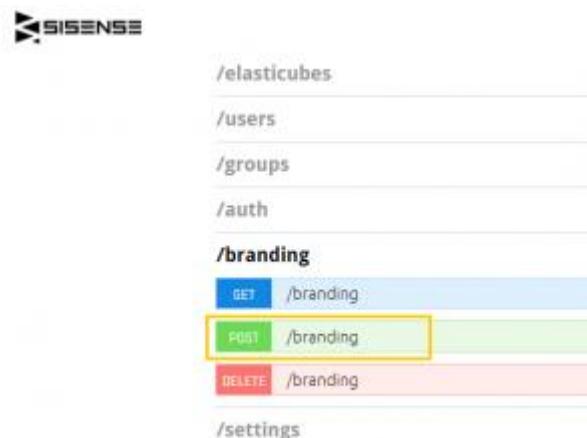
- ▶ Desktop Logo (for example: Logo-Desktop.png) –This is the main icon that is visible at the top left of the Sisense Web Application.
- ▶ Tablet Logo (for example: Logo-Tablet.png)
- ▶ Mobile Logo (for example: Logo-Mobile.png)
- ▶ Favicon (for example: favicon.ico)

Note: Optimal dimensions for logo images: 92×26

2. In the Sisense Web Application, click **ADMIN** at the top right, and then **REST API**, followed by **REST API Reference**.

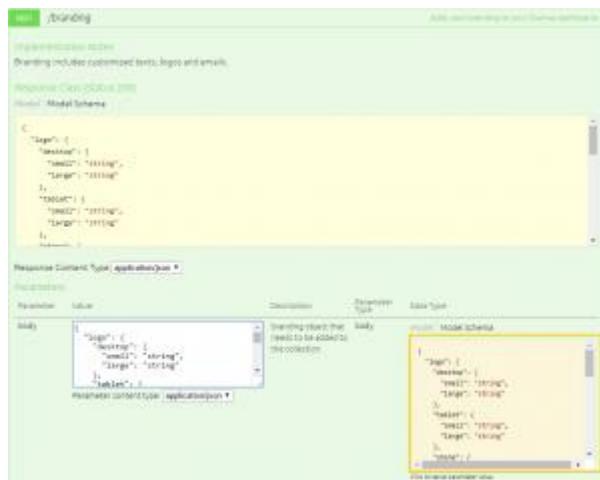


3. Click **/branding** and then click the POST operation **/branding**.



To get the required JSON format for the request, click the sample model schema to place the sample text in the body

parameter.



4. Replace the sample values with your required values. For an explanation of each parameter and how it modifies the Sisense Web Application, see White Labeling Configuration Options.
5. Click **RUN**. Refresh the Sisense Web Application page to view your changes.

Note: You can change just some of the parameters by leaving out those you do not want to update.

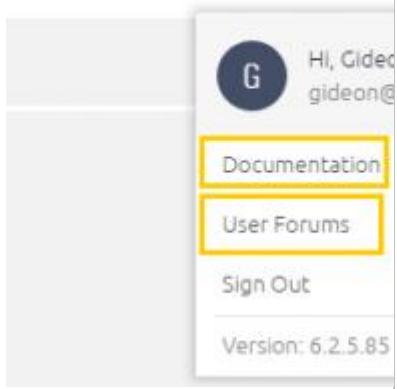
White Labeling Configuration Options

The following table provides a list of the features you can rebrand within the Sisense Web Application and the JAQL code that must be modified to rebrand it. Click the thumbnail image of the feature to see the default branding provided by Sisense and where your branding will appear.

Name	Description						
Page Title and Favicon	<p>The page title that appears in the browser tab/page and the Sisense favicon.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. Image must be in URI format (web address) 2. .ico format is preferred <p>You can modify these images by updating the values of the following JAQL parameters in the REST API:</p> <pre>"favicon": "/resources/branding/fav.ico"</pre> <pre>"pageTitle": "Sisense",</pre>						
Sisense Logo	<p>The Sisense logo can be replaced with your own. You need to supply two sizes of the logo, for each device (Desktop / Tablet / Mobile). These logos affect the following screens:</p> <ul style="list-style-type: none"> • Login Screen • Account Activation Screen • ‘Forgot Password’ Screen <p>Below is a table with the ideal logo dimensions (pixels):</p> <table> <tr> <td style="text-align: center;">Small</td> <td style="text-align: center;">Large</td> </tr> <tr> <td style="text-align: center;">Desktop</td> <td style="text-align: center;">92x26 204x60</td> </tr> <tr> <td style="text-align: center;">Tablet</td> <td style="text-align: center;">92x24 163x74</td> </tr> </table>	Small	Large	Desktop	92x26 204x60	Tablet	92x24 163x74
Small	Large						
Desktop	92x26 204x60						
Tablet	92x24 163x74						

	<p>Phone 92x24 198x168</p> <p>Important Notes:</p> <ol style="list-style-type: none"> 1. Images must be in URI format (web address) 2. No spaces are allowed in the address of the images 3. Transparent .png format is preferred <pre>"logo":{ "desktop":{ "small":"/resources/branding/logoDesktop Small.png", "large":"/resources/branding/logoDesktopL arge.png" }, "phone":{ "small":"/resources/branding/logoPhoneS mall.png", "large":"/resources/branding/logoPhoneLar ge.png" }, "tablet":{ "small":"/resources/branding/logoTabletS mall.png", "large":"/resources/branding/logoTabletLar ge.png" } }</pre>
--	--

	},
Homepage	<p>The home screen when your users log into Sisense.</p> <p>Note: The prefix http:// or https:// must be included.</p> <p>"homePage" : "http://www.mysite.com",</p>
Login Page	<p>The Login page title and subtitles.</p> <p>"loginTitle" : "WELCOME TO SISENSE", "loginSubtitle" : "Sign in & start exploring"</p>
Copyright Text	<p>The copyright text that appears in Sisense.</p> <p>"copyrightText": "copyright © 2014 sisense inc. All rights reserved",</p>
Activation Titles	<p>The titles on the activation page for new users signing in to Sisense.</p> <p>"activationTitle": "Welcome", "activationSubtitle": "Welcome",</p>

Contact Us 	<p>The Contact Us text is displayed when you request a password and enter your email on the Recover Password page.</p> <p>This text displays a message and provides a link to your Support services.</p> <p><code>"contactUsText": "For further assistance, please contact our success management team!: <a href='mailto:support@sisense.com'support@sisense.com","</code></p>
User Support 	<p>The User Support text is displayed in the top right corner of the Sisense Web Application.</p> <p>Be default, the links redirect customers to the Sisense user forums and documentation site. You can modify these URLs to redirect users to your user forums or documentation. This is useful for example when providing OEM services or if you have special forums or instructions to display to your users.</p> <p><code>"documentationUrl": "string", "forumUrl": "string"</code></p> <p>Replace the string with the URL to the relevant locations to redirect your users to your support sites.</p>
Emails	<p>The sender information and subject line text displayed in automated emails.</p>

	<p>For further customization, see <i>Rebranding Sisense Automated Emails</i>.</p> <p>Note: Branded emails (for example, using the senderEmail and senderName parameters) will be used for new user/group invitation emails and password recovery emails. When sharing a dashboard report, or for subscription emails, the email will be sent without the original name and email of the dashboard's creator.</p> <pre> "emails" : {"senderEmail":"sys- admin@company.com","senderName":"sys- admin","templates_directory" : "/resources/branding/emails","passwordRe coverySubject" : "Sisense password recovery", "newUserInviteSubject" : "Sisense account activation", "shareWithNewUserSubject" : "A Dashboard has been shared with you:", "shareWithExistingUserSubject" : "A Dashboard has been shared with you:" "transferOwnership": "string" } </pre>
--	---

Rebranding Sisense Automated Emails

Sisense automated emails are sent to users according to predefined scenarios. When a predefined scenario such as a build alert is triggered, an automated email is sent to the relevant Sisense users.

You can customize these emails to match your company's branding.

This page describes the Sisense automated emails and how you can replace them with your own so each time a predefined scenario is triggered, Sisense sends your customized emails to your Sisense users.

Sisense Automated Emails

You can rebrand the following automatic emails:

- ▶ **Build Alert:** Email sent each time a build alert is triggered.
- ▶ **Dashboard Errors Reports:** Email sent when a dashboard report fails.
- ▶ **Dashboard Errors Reports Bulk Fail:** Email sent when a large amount of email reports fail.
- ▶ **Dashboard Reports:** Email sent to the recipient of a shared dashboard when you share a dashboard.
- ▶ **Empty Dashboard Report:** Email sent to an administrator when an empty dashboard report is sent.
- ▶ **KPI Alert:** Email sent each time a KPI alert is triggered.
- ▶ **New User Invitation:** Email sent to a new user invited to Sisense.
- ▶ **Password Recovery:** Email sent to a user who requested a new password.
- ▶ **Share With:** Email sent to the recipient of a shared dashboard.

- ▶ **Transfer Ownership:** Email sent to a user who received ownership of a dashboard from another owner.
- ▶ **User Created:** Email sent to a user when a user is created.
- ▶ **User Created AD:** Email sent when a user is created in Sisense after being added from Active Directory.

For each automated email, Sisense provides an email template in the format of [EJS](#) and LESS files. These files are located within template folders inside the Template Parent folder located at:

```
...Program  
Files\Sisense\PrismWeb\vnext\src\features\emails\templates
```

LESS Files

All templates contain the style.less file that refers to a stylesheet that defines the branding used in Sisense's automated emails. The content of every style.less is as follows:

```
@import '../styles.less';
```

The stylesheet referred to by all the email templates is called styles.less located at:

```
...\Program  
Files\Sisense\PrismWeb\vnext\src\features\emails\templates
```

By replacing the content of the styles.less file with your own CSS, you can redefine the branding used in all of the email templates to your custom branding.

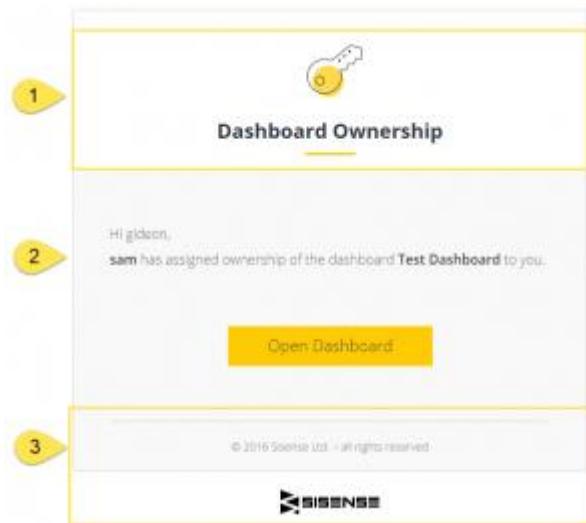
EJS Files

Each template folder contains an EJS file, which contains HTML and JavaScript that describe the content of the email.

You can use any text/HTML editor to modify the HTML.ejs files according to your needs.

The following is an example of an HTML.ejs file.

```
<% include ../header %>
<tr>
    <td>
        <div class="contentText">
            <p class="userText" style="margin: 0;"><%= i18nContent.hi %> <%= newOwnerUserName %>, </p>
            <p class="generalInfo" style="margin: 0;"><%- i18nContent.generalInfo %></p>
        </div>
    </td>
</tr>
<% include ../footer %>
```



Each EJS file has three sections, a header, body, and footer; each of which describes various areas in the email your users receive.

Section 1 is the Header section defined in the EJS file as `<%include ../header %>`. This is a reference to the header.ejs file located at:

`...\\Program Files\\Sisense\\PrismWeb\\Resources\\branding\\emails`

The header.ejs file defines the title and image used in the header section of each email template. You can modify the header by

replacing the header.ejs file with or replace the `<%include ..//header %>` tag from each template with your own HTML and JavaScript.

Section 2 is the Body section, which contains the content displayed in the email your users receive. There are two types of content displayed in automated emails, tokens and strings. Tokens refer to Sisense variables such as the Sisense username or dashboard owner's name. For example, the `<% newOwnerUserName %>` token displays the new owner of a dashboard's Sisense username in the automated email.

```
<tr>
    <td>
        <div class="contentText">
            <p class="userText" style="margin: 0;"><%= i18nContent.hi %> <%
            newOwnerUserName %>, </p>
            <p class="generalInfo" style="margin: 0;"><%- i18nContent.general
            %></p>
        </div>
    </td>
</tr>
```

When you replace the HTML.ejs file with your own files, you can still use the Sisense tokens. For more information about tokens you can add to your automated emails, see [Applying Sisense Tokens](#).

In addition, each template contains an object such as `i18nContent.hi` that displays strings predefined by Sisense. These strings cannot be modified, however, you can remove and replace the object with your own content.

The final section, Section 3 contains the footer. Like the header, the footer, `<%include ..//footer %>`, is a reference to a footer.ejs file located at:

```
...\\Program Files\\Sisense\\PrismWeb\\Resources\\branding\\emails
```

The footer.ejs file defines the copyright text and image used in the footer section of each email template. You can modify the footer by replacing the footer.ejs file with or replace the `<%include ..\footer %>` tag from each template with your own HTML and JavaScript.

Images that are displayed in the Header and Footer are stored in the following location:

```
...\\Program  
Files\\Sisense\\PrismWeb\\vnext\\src\\features\\emails\\templates\\im  
ages
```

You can replace these with your own images by maintaining the same file names used, however, it is recommended to replace the content of the Header and Footer sections altogether with your own content.

Replacing Sisense Automated Emails

If you prefer to implement your own email templates for Sisense's predefined scenarios, you can replace Sisense's emails with your own through the Sisense REST API.

Note: Rebranding Sisense emails must be enabled for your license. Contact your Account Manager for more information.

For each automated email, Sisense provides an email template in the format of [EJS](#) and LESS files. These files are located within template folders inside the Template Parent folder located at:

```
...\\Program  
Files\\Sisense\\PrismWeb\\vnext\\src\\features\\emails\\templates
```

Some template folders may contain an html-org.ejs file. These files are legacy templates and are maintained for backwards compatibility. Newer developments should only modify or replace the html.ejs files.

After you have prepared your new automated emails, through the /branding endpoint of the REST API, you can configure Sisense to send the customized automated emails when the relevant event is triggered.

To define the location of your branded emails:

1. Copy the contents of the directory:

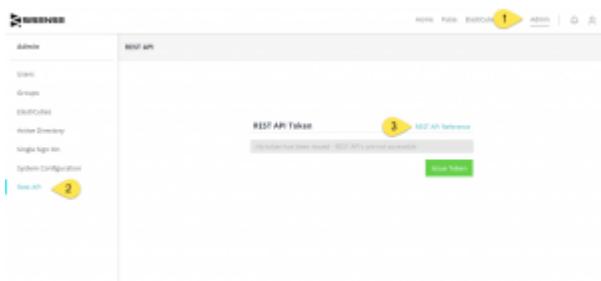
```
...Program  
Files\Sisense\PrismWeb\vnext\src\features\emails\templates
```

2. Paste all the folders in the following directory:

```
...Program  
Files\Sisense\PrismWeb\Resources\branding\emails
```

You will modify the templates in this folder and configure Sisense to send them instead of the original Sisense automated emails.

3. For each template you want to modify, change the contents of html.ejs files and styles.less as described in **Sisense Automated Emails**.
4. Access the Sisense REST API.



5. In version .9 of the REST API, select the POST /branding.



6. In the email object define the following information:

Note: If you have already rebranded your Sisense Web Application, copy the current settings through the GET /branding endpoint and paste them into the POST /branding endpoint modifying or adding the emails object. If you only modify the emails object, the remaining objects and keys are set to their default Sisense and will overwrite any existing settings.

Key	Description
senderEmail	<p>The email address from which the automated email is to be sent.</p> <p>When you define the senderEmail value, you must verify your domain in Sisense's Mandrill email service. Click here for more information.</p>
senderName	<p>The directory location of your template folders. Sisense accepts /resources/branding/emails as a custom location for automated email templates.</p> <p>This directory is located on your Sisense server at ...Program Files\Sisense\PrismWeb\Resources\branding\emails</p>

Key	Description
templates_directory	Create, update, delete, and query YouTube Analytics Groups.
passwordRecoverySubject	The subject line of the password recovery email.
newUserInviteSubject	The subject line of the new user email.
createdUserSubject	The subject line of the new user invite email.
shareWithNewUserSubject	The subject line of the shared dashboard with a new user email.
shareWithExistingUserSubject	Query time-based info on views, subscribers, etc. For example, how many new subscribers were gained at a specific time.
transferOwnership	The subject line of the dashboard transfer ownership email.

For example:

```
"emails": {
  "senderEmail": "sys-admin@company.com",
  "senderName": "sys-admin",
  "templates_directory": "/resources/branding/emails",
  "passwordRecoverySubject": "Recover Password",
  "newUserInviteSubject": "new User Account Activation",
  "shareWithNewUserSubject": "A Dashboard has been shared with you:",
  "shareWithExistingUserSubject": "A Dashboard has been shared with you:"
},
```

- Click **Run**. The next time an automated email is sent, your templates are sent instead of Sisense's predefined emails.

Applying Sisense Tokens

Most Sisense automated emails include tokens that you can apply to your automated emails. When the email is sent, Sisense populates the token with the relevant value. For example, the <% user %> token represents the Sisense's username.

Each template has tokens defined by Sisense that you can apply listed below. Tokens from one automated email cannot be applied to another. Applying tokens to other automated emails, for example adding <% user %> to the KPI alert email results

build_alert: Sent when a build alert is triggered.

No Template Tokens Available

Dashboard_errors_report: Sent when an Dashboard email report fails.

Template Tokens Available

<% user %>

<% errors[i].user %>

<% errors[i].error %>

<% errors[i].widgetFailed %>

<% errors[i].widgetCount %>

Dashboard_errors_report_bulk_fail: Sent when a Dashboard email report sent to a large amount of users fails.

Template Tokens Available

<% user %>

<% errors[i].user %>

<% errors.groups[i] %>

Dashboard_report: An automated email that contains a dashboard report sent to users.

Template Tokens Available

<% url %>

<% images[i] %>

Empty_dashboard_report: An automated email that contains an empty dashboard report sent to users.

Template Tokens Available

<% user %>

Kpi_alert: Sent to shared users when a KPI alert is triggered.

Template Tokens Available

<% measure %>

<% value %>

<% message %>

New_user_invitation: Sent to a new user that you have invited to join.

Template Tokens Available

<% owner %>

* The folder for this template contains a file called HTML-ORG.ejs.

This file is for older versions of Sisense and is maintained for backward compatibility.

Password_recovery: Sent when a user requests their password from the Forgot Password page.

Template Tokens Available

<% user %>

* The folder for this template contains a file called HTML-ORG.ejs. This file is for older versions of Sisense and is maintained for backward compatibility.

Share_with: Sent to a new recipient of a dashboard when the dashboard is shared with them.

Template Tokens Available

<% user %>

* The folder for this template contains a file called HTML-ORG.ejs. This file is for older versions of Sisense and is maintained for backward compatibility.

transfer_ownership: Sent to the new owner of a dashboard when ownership is transferred.

Template Tokens Available

<% newOwnerUserName %>

<% oldOwnerUserName %>

<% sharingDashboardName %>

User_created: Sent to the recipients of new Sisense accounts.

Template Tokens Available

<% user %>

User_created_ad: Sent when a user is created in Sisense after being added from Active Directory

Template Tokens Available

<% user %>

Displaying Custom URLs

You can display your company's URL in the address of your Sisense dashboards by defining an alias URL from the Sisense Admin page.

This is useful when you are providing OEM services and you want to mask Sisense's URLs and branding.

To implement an alias URL in Sisense:

1. Click **ADMIN** in the top menu, and then **SYSTEM CONFIGURATION** in the left menu.
2. Click **EDIT** to enable the **Alias** field.
3. In the **Alias** field, enter the web server's alias or IP address. This is especially important when you have configured your IIS with a domain name (alias), but when you share a dashboard, the IP address and/or the incorrect port is sent instead. Reports will be sent from the alias entered in this field.
4. Click **SAVE**.

Internationalization and Localization

Sisense supports localization and internationalization by displaying your users' native language in the Sisense Web Application. You can select the language displayed in the Sisense Web Application from a list of languages provided by Sisense or display your own translation. Each time you select a language, the Sisense Web Application interface is displayed in that language.

When you change the default language displayed in the Sisense Web Application, words that are part of the Sisense framework such as

menu items, buttons, and messages are displayed in the selected language. Words specific to your dashboard, such as table names, field names, and filters, are not automatically translated when you select a new language to display. These words are called metadata and you can provide your own translation of these terms and display those translations to your users through a Dashboard script.

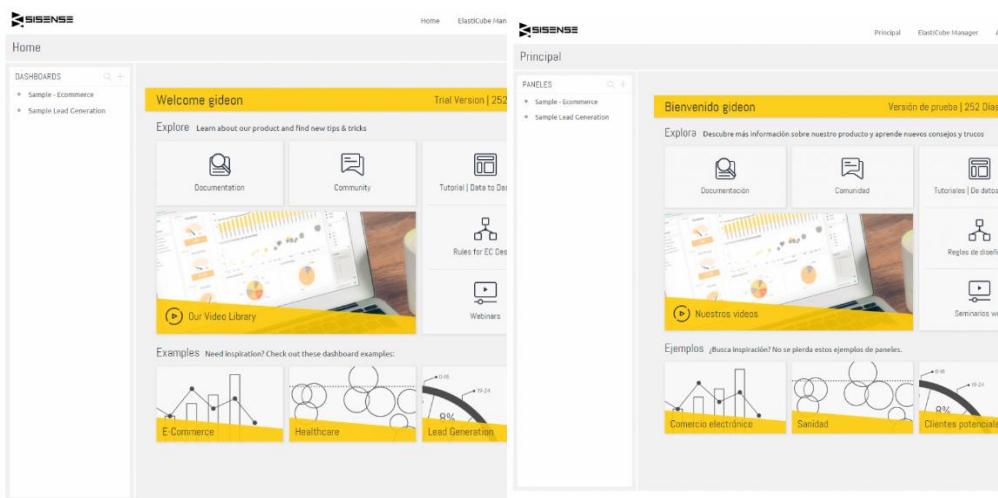
For instructions on how to change the language displayed in the Sisense Web Application, see [Changing the Sisense Web Application's Language](#).

For instructions on how to translate your metadata, see [Translating Sisense Metadata](#).

Changing the Sisense Web Application's Language

Sisense supports multiple languages in the Sisense Web Application making it easier to extract meaningful insights from your data in your users' local language.

The following images display two interfaces, one in English and the other in Spanish.



English

Spanish

What has been Translated?

Sisense has translated text that your users might see including menu names, buttons, messages, and other elements in the Sisense Web Application. Translating the Sisense Web Application does not affect your data or how it is displayed.

The Sisense Web Application has been localized to the following languages:

- ▶ Chinese (Simple)
- ▶ French (France)

- ▶ German
- ▶ Italian
- ▶ Japanese
- ▶ Portuguese (BR)
- ▶ Russian
- ▶ Spanish (ES)
- ▶ Spanish (LA)

Note, the following content is not being localized:

- ▶ Built-in Javascript editor
- ▶ ElastiCube Manager and Server Console
- ▶ Sisense documentation and online help
- ▶ REST API and documentation
- ▶ Sisense automated emails

Note: You can use your own translation of Sisense automated emails by defining your own.

Changing your Default Language

From the System Configuration page, you can select the default language for the Sisense Web Application. After selecting a new language, each string with a translation is displayed in that language. Some words and concepts, such as ElastiCube, are still displayed in English.

If you have defined a custom translation, the name of the language as defined in the settings.json is displayed in this list. See Embedding Custom Languages below for more information.

Sisense's default language is English. After changing the system language, you can revert back to the English or any other language by selecting it from the list.

To modify your default language:

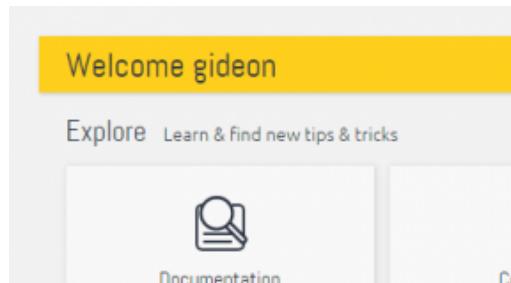
1. In the Sisense Web Application, click **ADMIN** in the top menu, and then **SYSTEM CONFIGURATION** in the left menu.
2. Click **EDIT**.
3. Select the relevant language. After selecting the language, the UI is immediately translated.
4. Click **Save**. The Sisense Web Application is translated in the selected language.

Embedding Customized Languages

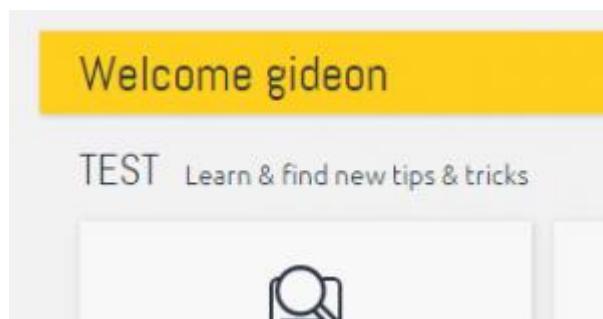
Sisense enables you to display your own translations in the Sisense Web Application by replacing the values of strings in any language you choose.

You can embed customized languages by modifying configuration and resource files replacing the string values according to your requirements.

For example, in the homepage of the Sisense Web Application, the word "Explore" is displayed. In the resource file home.js, you can modify the value of this string to display your own text as shown below.



```
"welcome_user": "Welcome",
"explore": {
  "explore": "Explore",
  "learn": "Learn & find new tips &
tricks",
  "links": {
```



```
"welcome_user": "Welcome",
"explore": {
  "explore": "TEST",
  "learn": "Learn & find new tips &
tricks",
  "links": {
```

In addition to modifying the resource files, you must modify the settings.json file and specify your language's name and update the version. In this example, the language has been changed to "test" and the version increased from 1.0.0 to 1.0.1.

```
{ "displayName": "test", "version": "1.0.1" }
```

The value of "displayName" is displayed as your language's name in Default Language list on the System Configuration page. In the example below, the displayed was set to Test.



Upgrading Sisense may replace the bundled translation files. If you wish to modify a bundled language, copy all of the translation files to a new directory, and change the copied versions to prevent your translation from being overwritten.

To customize the Sisense Web Application's language:

1. Navigate to the directory "..\Sisense\PrismWeb\translations\"
2. In the \translations directory, copy the **en-us(default)** directory, paste it in the \translation directory and rename it.
3. In the new directory, open each resource file and translate the strings into the desired language saving each file after your changes are made.
4. Open the settings.json file and replace the following key values:
displayName: Enter the name of the language.
version: Enter the name of the new version. Currently, the version is 1.0.0. For each version, increase this value, for example, 1.0.1.
5. Save the changes in a file called settings.json in your new language folder.
6. In the Sisense Web Application, click **ADMIN** in the top menu, and then **SYSTEM CONFIGURATION** in the left menu.
7. Click **Edit**.
8. Select your customized translation from the list. The Sisense Web Application is displayed in your customized translation.
9. Click **Save**.

Translating Sisense Metadata

Sisense enables you to translate metadata in your users' native language. Sisense metadata is displayed in your users' dashboards and includes the following:

- ▶ ElastiCube fields
- ▶ ElastiCube tables
- ▶ Widget Titles
- ▶ Dashboard Names
- ▶ Folder Names
- ▶ Saved formulas
- ▶ Default and custom hierarchies
- ▶ Customized renamed field titles

The following images display two interfaces with metadata, one in English and the other in Russian.

Brand	Бренд
Thrubanower Holdings Corp.	Thrubanower Holdings Corp.
Undimedar WorldWide	Undimedar WorldWide
Reseller	Reseller
APPLES	APPLES
WHITEBERRY	WHITEBERRY
MICRON	MICRON
Unerommazz Direct	Unerommazz Direct
LONY	LONY
POKIA	POKIA
NG	NG
Qwhupistor Holdings Group	Qwhupistor Holdings Group

<< < 1 2 3 4 5 6 7 > >>

English
Russian

As fields such as ElastiCube names and tables are defined by you, these fields are not translated when changing the default language of the Sisense Web Application.

You can translate your metadata by modifying the default values using a JavaScript plugin. Through the plugin, you define the new values to be displayed. You may also include additional logic to determine which users or user groups the translated values should be displayed to.

When you create a JavaScript plugin, you should create a folder for the plugin and place it inside the plugins folder in the following location:

```
...\\Program Files\\Sisense\\PrismWeb\\plugins\\
```

If the \\plugins\\ folder does not exist, create it and then create a new folder within the \\plugins\\ folder with the relevant files.

You can download an example of a Metadata Translation plugin [here](#).

Extract the enclosed folder into the plugins folder, ..\\Program Files\\Sisense\\PrismWeb\\plugins\\.

This file contains a folder called Metadata and two files:

plugin.json: The plugin.json file contains the name of the plugin and a link to the file that describes the plugin's functionality, in this example, run.js.

run.js: The run.js file contains the plugin's functionality and describes which metadata is to be translated and how it is to be translated.

This example, and any metadata translation JavaScript file you write should include three parts:

1. **Provider Function:** These functions provide the mapping context contained in the Mapping object.
2. **Event Subscription:** This event is triggered when a datasource changes, and calls the Provider function.
3. **Mapping Object:** Contains key/value pairs with your default and translated metadata.

Provider Functions

The code sample below is an example of two Provider functions, the first is a global provider function for folder and dashboard aliasing. The second is the datasource provider function for translating ElastiCube and formula names. You can replace the parameters in this example to provide the mapping context you define in the Mapping object.

Global Provider

```
function (resolve, reject) {
    //an ajax request or any other logic.
    setTimeout(function () {
        resolve(globalAliasing);
    }, 0);
};
```

The following table describe the Provider function and its arguments:

Parameter	Description
resolve	A callback function which should be called with the provided aliasing context object or without any parameter if aliases shouldn't be provided for the datasource.
reject	A callback function to be called in case of error, so the system can properly handle it.

Datasource Provider

```

function (datasourceId, resolve, reject) {
    //an ajax request or any other logic.
    setTimeout(function () {
        resolve(datasourceAliasing);
    }, 0);
}
;

```

The following table describe the Provider function and its arguments:

Parameter	Description
datasourceId	An object that represents the ID of the datasource with the following properties: 1. address 2. database 3. fullname 4. ID 5. title
resolve	A callback function which should be called with the provided aliasing context object or without any parameter if aliases shouldn't be provided for the datasource.
reject	A callback function to be called in case of error, so the system can properly handle it.

Event Subscription

To trigger the Provider functions, your plugin should include a handler for a global event 'beforealiascontextinit'.

In the example below, datasourceProviderFunction and globalAliasingProvider are the Provider functions. The timeout parameter is optional and configures the amount of time in milliseconds which the system waits for the aliasing context to be resolved by the provider function. By default the system uses a timeout of 2000 ms

```
prism.on("beforealiascontextinit", function (ev, args) {  
    var timeout = 2000;  
    args.register(datasourceProviderFunction,  
        globalAliasingProvider, timeout);  
});
```

Mapping Object

The translation context is a simple JavaScript object which contains key value pairs for tables, columns, hierarchies, formulas and custom item titles, all case-insensitive.

The sample below is a skeleton of a Mapping object that defines what metadata is to be translated and its new value.

```
/**  
  
 * tables - aliases for table names  
 * "tables": {  
 *     "table_name_1": "alias_1",  
 *     "table_name_2": "alias_2",  
 * }  
 * columns - aliases for column names in scope of table  
 * "columns": {  
 *     "table_name_1": {  
 *         "column_name_1": "alias_for_column_name_1"  
 *         "column_name_2": "alias_for_column_name_2"  
 *         "column_name_3": "alias_for_column_name_3"  
 *     },  
 *     "table_name_2": {  
 *         "column_name_1": "alias_for_column_name_1"  
 *         "column_name_2": "alias_for_column_name_2"  
 *         "column_name_3": "alias_for_column_name_3"  
 *     }  
 * }  
 * titles - aliases for custom dimension titles, etc  
 * "titles": {  
 *     "Custom title": "Alias",  
 * }
```

```

* "Sample Revenue": "Sample Profit"
* },
* formulas - aliases for custom formula titles
* "formulas": {
* "Custom title": "Alias"
* },
* hierarchies - aliases for hierarchy titles
* "hierarchies": {
* "Custom title": "Alias"
* }
*/
var datasourceAliasing = {
"tables": {
"brand": "бренд",
"category": "категория",
"country": "страна",
"commerce": "коммерция"
},
"formulas": {
"Revenue": "Чистый доход"
},
"hierarchies": {
"Category by Brand and Age Range": "Категория через Бренд и
Год"
},
"titles": {
"CUSTOM REVENUE": "доходы"
},
"columns": {
"Brand": {
"Brand": "Бренд",
"Brand ID": "Бренд Id"
},
"Category": {
"Category": "Категория",
"Category ID": "Id Категории"
},
"Commerce": {
}
}
}

```

```

"Age Range": "Возрастные диапазоны",
"Brand ID": "Id Бренда",
"Category ID": "Id Категории",
"Condition": "Состояние",
"Cost": "Затраты",
"Country ID": "Id Страны",
"Date": "Дата",
"Gender": "Пол",
"Quantity": "Количество",
"Revenue": "Доходы",
"Visit ID": "Id Посетителя"
},
"Country": {
"Country ID": "Id Страны",
"Country": "Страна"
}
}
};

/***
* Global object example
* folder - aliases for folder names
* "folder": {
* "folder_name_1": "alias_1",
* "folder_name_2": "alias_2",
* }
* dashboards- aliases for dashboard names
* "dashboards- ":
* "dashboard_name_1": "alias_1",
* "dashboard_name_2": "alias_2",
* }*
*/
var globalAliasing = {
"folders": {
"New Folder": "Folder renamed",
"my folder": "my folder renamed",
"Any name": "любое название"
},
"dashboards": {

```

```
"Sample - Ecommerce (1)": "Коммерческий Dashboard",
"Sample - Healthcare": "Sample Healthcare renamed",
"Sample - Healthcare (1)": "Healthcare renamed",
"my dashboard": "my dashboard renamed"
},
};
```

System Configuration

The System Configuration settings let you define the web server's alias, the first day of the week, SSL security, and email settings.

To update one or more System Configuration settings:

1. Click **ADMIN** in the top menu, and then **SYSTEM CONFIGURATION** in the left menu.
2. Click **EDIT** and update one or more of the following settings:
 - ▶ **First Day of the Week:** Select the first day of the week so all of your Sisense users' queries will be calculated according to the day you define. By default, the first day of the week is Monday when querying an ElastiCube. Sisense defines Monday as the first day according to ISO 8601. This means that when selecting filters such as Week 1, Week 2, etc., the first day of the week is not that defined by first day of that year, but rather the first Monday of the year. Each week starts on a Monday and ends on a Sunday for 52-53 full weeks in a year.
Note: Changing the first day of week after using a dashboard or widget filter/drill with Weeks filter in use might lead to bad results. To resolve the issue, the Weeks filter should be removed and added again.
 - ▶ **Default Language:** Select the language to be displayed in the Sisense Web Application. If you have defined a custom

translation, the name of the language as defined in the settings.json is displayed in this list. Select it to display your custom translation.

- ▶ **Alias:** Define the web server's alias or IP address. This is especially important when you have configured your IIS with a domain name (alias), but when you share a dashboard, the IP address and/or the incorrect port is sent instead. Reports will be sent from the alias entered in this field.
- ▶ **SSL:** Enable SSL if you have configured your web environment with SSL security (https) and certificates. This will enable https in the URLs of shared dashboards.
- ▶ **SEND EMAILS:** You can disable sending emails. This will disable emails sent when sharing dashboards as well as any scheduled email reports.

3. When done, click **SAVE**.

Setting Up a Custom Email Server

Sisense uses Mandrill by MailChimp to send automated emails such as dashboard reports or password reset emails. Custom email servers allow all our email transactions to be completely managed on your side.

Through the Sisense REST API, you can send an HTTP POST request to issue Sisense emails through your own custom email server instead of the default Sisense email server. The REST API exposes properties that you can configure for the custom email servers such as security, timeout, and logging properties.

To set up a custom email server:

1. In the Sisense Web Application, click **ADMIN** at the top right, and then **REST API**, followed by **REST API Reference**.
2. In the top-right corner of the page, select version 1.0 of the REST API.
3. Select **Settings**, then **POST /settings/email_server**.
4. In the body of your call, define the values of the following keys:

Key	Value
enable	If True, allows access to the custom email server instead of the default Sisense email server. Note: This property is different from the enable property found in version .9 of the REST API POST /settings/system request. In .9, enable activates Sisense emails. In version 1.0, enable sends Sisense emails through your custom server.
host	The hostname or IP address to connect to. By default, this is 'localhost'.
port	The port to connect to (defaults to 25 or 465).
secure	When you send an email, some email servers check if secure transport (TLS) is available for that address or domain. This value determines if the connection should use TLS (if True) or not (if False). The default value is false. The connection may be upgraded to TLS if the email server requires it.
ignoreTLS	If this is True and secure is False, TLS will not be used even if the email server attempts to upgrade the connection to TLS-secure. This field should be set to True when connecting to a non-secure SMTP server.

Key	Value
requireTLS	If this is True and secure is False, it forces the connection to be upgraded to TLS-secure even if the email server does not advertise support for TLS.
connectionTimeout	How many milliseconds to wait for the connection to establish.
greetingTimeout	How many milliseconds to wait for the greeting after connection is established.
socketTimeout	How many milliseconds of inactivity to allow.
user	The user name of the user. As non-secure or restricted SMTP servers do not require authentication, do not include this field when connecting to a non-secure SMTP server.
pass	The user's password for the email server. As non-secure or restricted SMTP servers do not require authentication, do not include this field when connecting to a non-secure SMTP server.
logger	If set to True, Sisense logs events such as which server you connected to. If the value is not defined or is False, then nothing is logged.
debug	If set to True, then Sisense logs SMTP traffic. When False, Sisense logs only transaction events. Debug logs record everything and may increase in size quickly and impact performance. You should only set activate debug logging when troubleshooting a specific issue and then reset the logging level back to info.

5. Click **Run**.

Limitations

- ▶ When authenticating an HTTP request, Sisense only supports the basic authentication through the Authorization method using plain text.
- ▶ No Errors are displayed in the UI.

Troubleshooting a Custom Email Server

Sisense does not provide a UI for troubleshooting an issues with your custom email server. To troubleshoot any issue with your custom email server, activate logging through the **logger** and **debug** keys in the REST API. When **logger** and **debug** are set to True, Sisense generates logs that are saved in the following directory:

c:\Program Files\Sisense\PrismWeb\vnext\iisnode
After troubleshooting the issue, the debug logger should be disabled as the logs can grow in size rapidly and this may impact performance.

Managing Drill Hierarchies

Dashboard viewers can drill down in most widgets to get an in-depth view of a selected value. To make it easier for viewers to select drill-down paths, dashboard designers can add predefined drill hierarchies to their widgets.

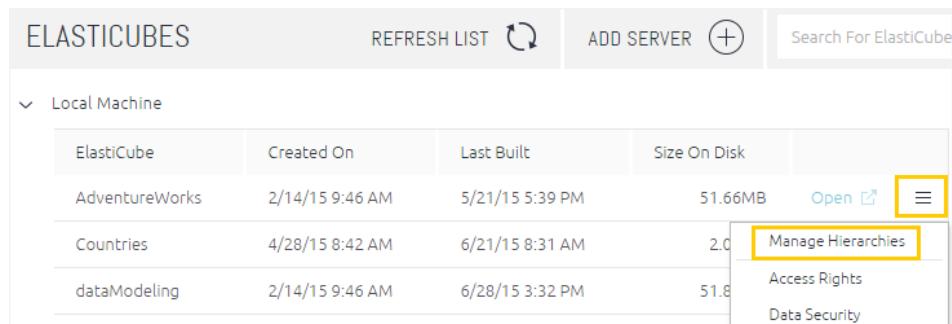
As an administrator, you can create a pool of drill hierarchies, from which dashboard designers can select the drill hierarchies they want to include in the widgets they build.



This section describes how to create and manage the drill hierarchies that you want to make available for designers.

To create a drill-down hierarchy:

1. Click **ADMIN** in the top-right corner of the Sisense web app. In the left menu, select **ELASTICUBES**.
2. Hover over the relevant ElastiCube, and from the menu on the right, select **Manage Hierarchies**.



The screenshot shows the 'ELASTICUBES' page with a header containing 'REFRESH LIST', 'ADD SERVER', and a search bar. Below the header is a table with columns: ElastiCube, Created On, Last Built, and Size On Disk. There are three rows: 'AdventureWorks', 'Countries', and 'dataModeling'. For the 'AdventureWorks' row, there is a 'Manage Hierarchies' button highlighted with a yellow box. A vertical ellipsis menu is also visible next to the 'Manage Hierarchies' button, with options: 'Manage Hierarchies', 'Access Rights', and 'Data Security'.

ElastiCube	Created On	Last Built	Size On Disk
AdventureWorks	2/14/15 9:46 AM	5/21/15 5:39 PM	51.66MB
Countries	4/28/15 8:42 AM	6/21/15 8:31 AM	2.0
dataModeling	2/14/15 9:46 AM	6/28/15 3:32 PM	51.8

3. If this is the first hierarchy you are adding, select **Add Hierarchy**. If you already have hierarchies, click **Add Hierarchy** in the top-right corner.
4. Select the field. This will be the first field in the drill hierarchy, and the same field used in the widget.
5. Click **Add Field** to add the next level in the drill hierarchy. Repeat this step for as many levels as you need.

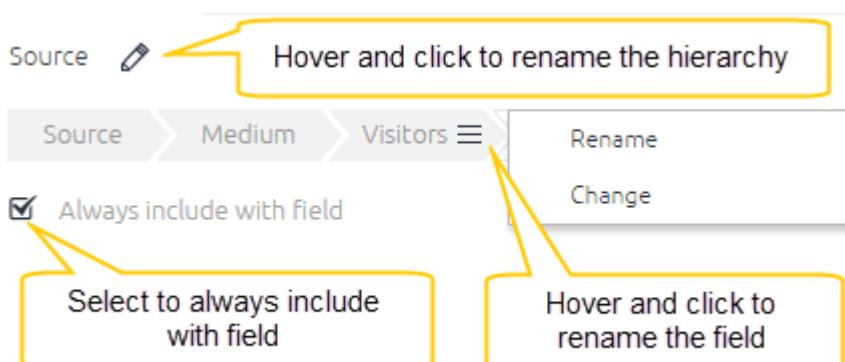


Additional options include:

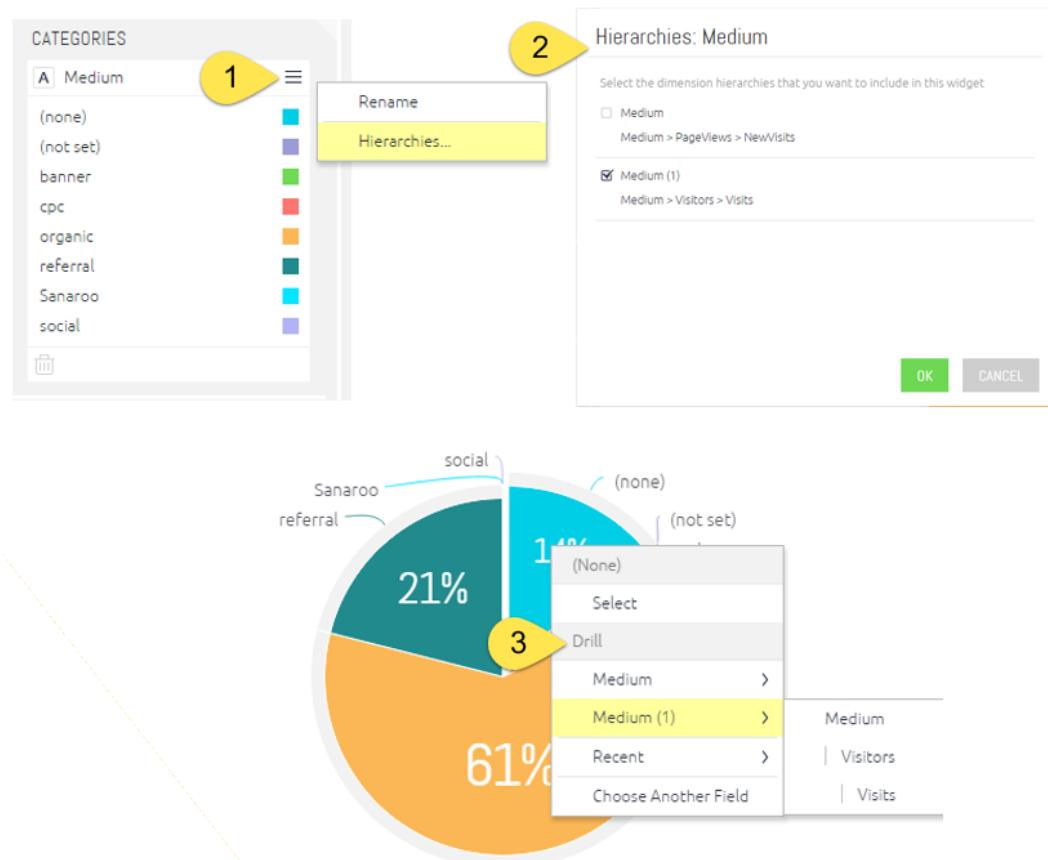
Always include with field: Select this checkbox to always include the drill hierarchy in the widget for this field.

Editing the Hierarchy name: To edit the name of the drill hierarchy, click on the pencil icon, and type in the name. Click on the tick icon to confirm the change.

Editing the field name: To edit the name of a field in the hierarchy, hover over the field name and click on the menu > **Rename**. Type in the name, and click on the tick icon to confirm the change.



6. Add as many hierarchies as you need for the ElastiCube. When done, dashboard designers will have access to the hierarchies in the widget designer, and will be able to select and add them to the widget.



To change a field in a hierarchy:

- ▶ Hover over the field in the hierarchy, and click on the menu that appears. Click **Change** and select a different field.



To delete a hierarchy:

- ▶ Click on the bin to the right of the hierarchy.

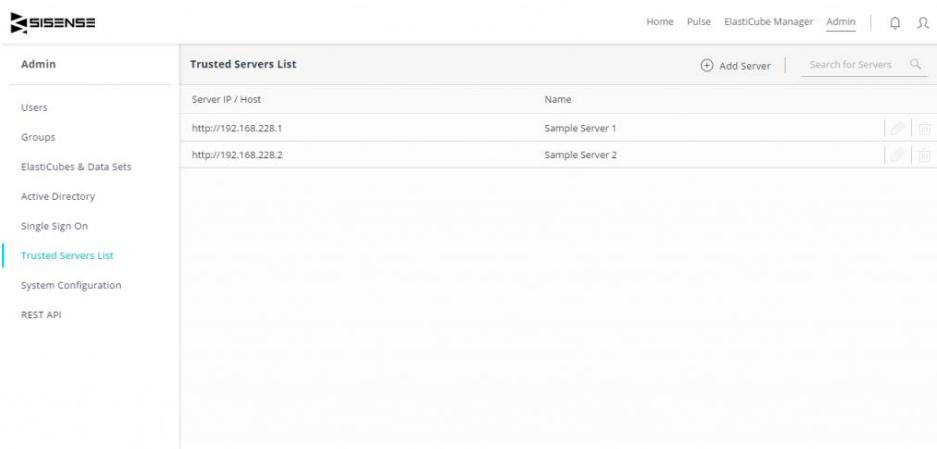
Migrating Sisense across Environments

Many users that implement Sisense maintain copies of their dashboards locally or across multiple environments for a variety of reasons. With multiple instances of Sisense, you can test and verify your data's accuracy before publishing dashboards to a production environment where accuracy is critical. You can save external copies of dashboards for disaster recovery. OEMs can support a master server and separate servers for each of their customers, copying the relevant dashboards from the master server to the appropriate customer server. Sisense enables Administrators and Designers to maintain multiple copies of their dashboards and migrate them quickly and easily across multiple environments.

There are two options for migrating dashboards from the Sisense Web Application. The first option is to export your dashboards into a .dash file locally, and then import the dashboards into the relevant environments. This option is useful if you do not have access to the server, for example, if you are an OEM providing dashboards to your customers. For more information about exporting dashboards, see [Exporting Sisense Dashboards](#).

The second option is to copy the dashboard to another server from the Sisense Web Application. In this scenario, you can copy dashboards from one environment directly to another environment after the Administrator of the target server provides access to the source server. The target server can only accept files from servers available on the Trusted Server List. For example, if you have two machines with Sisense installed, on the server where you want to

copy dashboards to, you can provide access to the source server by adding the source server's IP address to my list of allowed servers.



Server IP / Host	Name
http://192.168.228.1	Sample Server 1
http://192.168.228.2	Sample Server 2

After providing access, an Administrator or Designer can copy the dashboard directly to the target server. If the dashboard already exists on the target server, the Administrator or Designer will be notified and can overwrite the existing dashboard or create a duplicate dashboard on the target server.

The option you choose depends on which methodology is convenient for you and what you need to achieve. The end result is the same.

There are several things you should remember when copying a dashboard to another server. First, copied dashboards and their widgets keep their IDs. This ensures that any users with bookmarks of the dashboard or its widgets can continue using the same bookmarks.

Second, when copying a dashboard, the dashboard's filters and scripts on the target server are overwritten. The filters and scripts from the source are used instead. If your data source is not the same in each environment, you will need to manually adjust the data source after copying the dashboards. If plug-ins are installed on the original

server, these plug-ins will not work on the dashboard unless they are also installed on the target server.

Third, changes to your dashboards are not reflected on the same dashboard in another environment. To modify a remote dashboard you can either modify it on the remote server, or modify it in your source server, and repeat the migration process. Once you have completed your changes, you will need to republish the dashboard.

Copying Dashboards between Sisense Servers

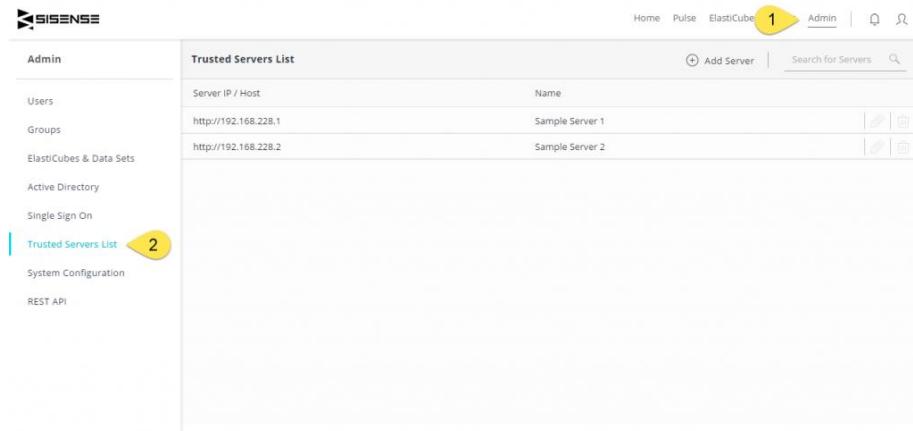
Administrators and Designers with access to a remote Sisense server can copy dashboards from one Sisense server to another from the Sisense Web Application. When you copy a dashboard or a folder to a remote server, all the dashboards, including those in the selected folders are duplicated into the main Dashboard list in the target server. Folders are not duplicated on the target server.

Before copying a dashboard to a target server, the Administrator of the target server must provide access to target server. The procedure below explains how to provide access to another server and then how to copy a dashboard from the source server to the target server through the Sisense Web Application.

To copy your dashboard to another Sisense server:

Note: Some of the dialog boxes may be considered as pop-ups by your ad blockers. Make sure that Sisense is whitelisted or that your ad blocker is disabled when performing this procedure.

1. On the target server, the Administrator needs to provider access for the source server to the target server. On the target Sisense server, select **Admin > Trusted Server List**.



The Trusted Servers List page is displayed.

Server IP / Host	Name
http://192.168.228.1	Sample Server 1
http://192.168.228.2	Sample Server 2

2. In the Trusted Server List page, click **Add Server**. The Create a New Server Access dialog box is displayed.

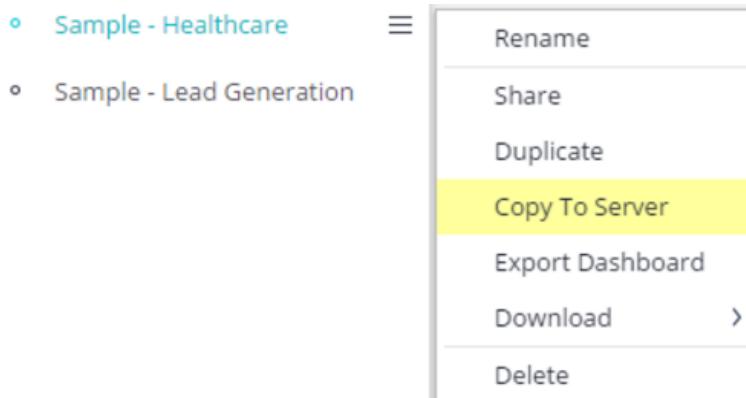
Add Server

Server	<input type="text" value="Type URL or Hostname (Including Port)"/>
Name	<input type="text" value="Optional"/>

Cancel

3. In the **Server** field, enter the IP address or hostname of the source server that contains the original dashboard.
4. In the **Name** field, enter a name for the source server. This is an optional identifier to help you keep track of your servers.
5. Click **OK**. The server is added to the list and now has copy access to the target server.

On the source server, from the Dashboard list, click the dashboard's menu and click **Copy to Server**.



OR

To copy multiple dashboards, in the Dashboards list, click the Select Dashboards button  to display checkboxes next to the dashboard and folders, which allows you to select multiple dashboard to be exported. Then, select the relevant dashboards and folders to be exported and click . The Copy Dashboards to Server dialog box is displayed.

Copy Dashboards to Server

Server

Type URL (including port number)

Republish dashboards after copying

Cancel

6. In the Copy Dashboards to Server dialog box, enter the following information:

Server: The URL or Hostname of the remote server. Sisense supports SSL connections so dashboards can be transferred

securely when SSL is activated.

Republish dashboards after copying: Select to automatically republish the dashboard on the target server after the selected dashboards are copied. Republishing a dashboard refreshes the dashboard definitions (widget types, titles, scripts etc.) for each of the users who share the dashboard.

7. Click **OK**.

The selected dashboards are copied to the remote server and displayed in the Dashboard's list.

Viewing Dashboards on Mobile Devices

You can easily view your Sisense dashboards on the go.

On a mobile phone you can use Sisense's native mobile app or a browser to view your dashboards. Widgets appear one below the other in a view-only scrollable list.

On tablets, you can use a browser and experience the same full viewer mode as experienced on a desktop computer, so you will be able to change dashboard filters and drill into fields.

Using the Sisense Mobile App

You can download the Sisense mobile app for Android mobile phones or iPhones.

Note: The app only works on mobile phones. For tablets, see Using Sisense in a Mobile Phone or Tablet Browser.

Supported Phones and Operating Systems

- ▶ iPhone –Requires iOS 8 or later, Supported iPhone 5 device or higher.
- ▶ Android –Requires Android versions 4.3 or later.

Get Sisense Mobile

To download the Sisense Mobile App for your device, click the relevant link below according to your version of Sisense.

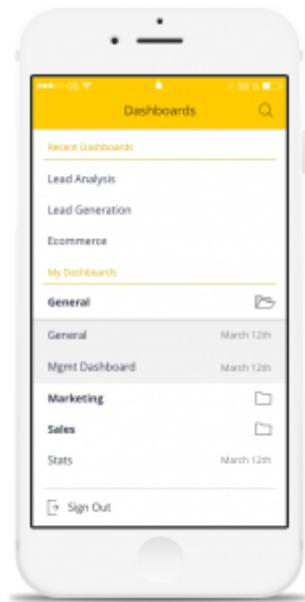
The latest version of the Sisense Mobile app supported by Sisense V6.5 offers improved performance and the ability to receive mobile notifications from Sisense Pulse.



Sisense V6.5 and later.



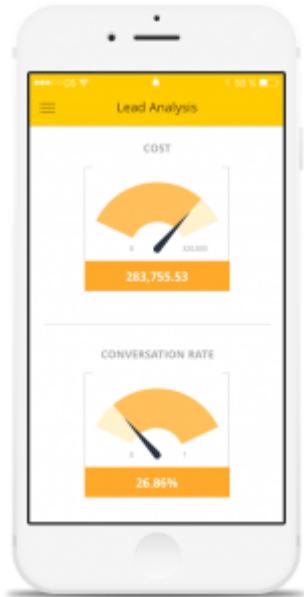
Working with the Mobile App



1. Tap the Sisense app icon to open the application.
2. If this is your first time logging in, type in the server address and then enter your username and password to log in.

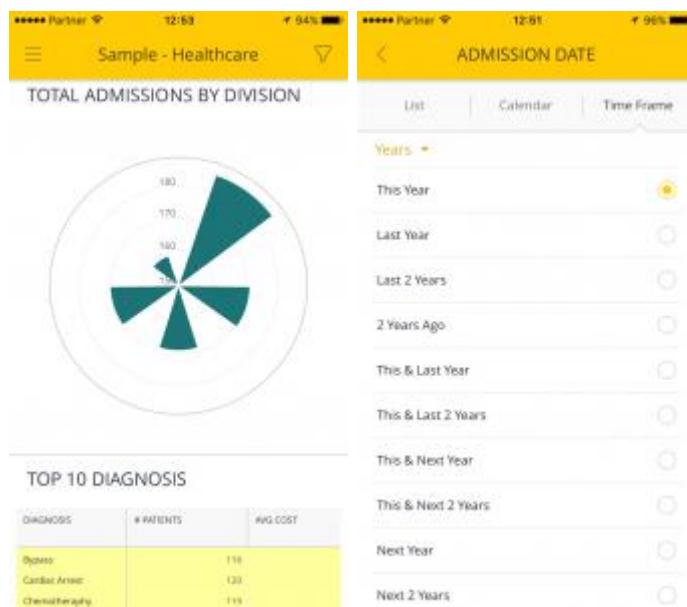
The server address should include the server IP or domain name (alias). If you are not using the default port (80) you will need to add the port as well, for example: xx.xxx.xx.xx:8081.
When working with SSL security protocol, please use the 'https' prefix.
3. If this is not your first time logging in, you will be prompted to enter your username and password. You can also tap **server** to enter the Sisense server address or choose the server from a list, and then sign in.
4. When you receive a notification from Sisense Pulse to your mobile device, tap the notification to open the dashboard from where the alert originated.
5. To open a dashboard, tap the dashboard's name or tap a folder name, and then on the relevant dashboard.

Recently used dashboards appear above the full list of dashboards (MY DASHBOARDS) for a quicker navigation.



6. To search for a dashboard, tap the magnifying glass icon and type in the name of the dashboard.
on iPhone devices, tap **Search**, and then tap the dashboard name.
7. In the dashboard, drag the dashboard up or down to view the widgets.
8. To get additional information on the presented dashboard, tap on the dashboard name (iPhone) or click on the 'i' sign at the top right (Android).
9. Within a widget, you can tap on a value to see more details about the selected value. For example, tap on a slice in a pie chart to see the absolute value in addition to the percentage.
10. To select a different dashboard, tap the menu icon.
11. To filter a dashboard, you can tap the  icon. This displays the list of available filters. Tap the relevant filter to open all of its available options. Tap any of the filter options to apply the

filters to your data.



You can sign out by tapping **Sign Out** at the bottom of the Dashboards page.

Using Sisense in a Mobile Phone or Tablet Browser

The Sisense web app is built around a responsive design that automatically optimizes viewing for mobile phones and tablets. Sisense runs in any HTML5-supported browser.

On mobile phones, widgets appear one below the other in a view-only scrollable list.

On tablets, you get to see the same view as experienced on a desktop. Also, full viewer mode is supported, so you can change dashboard filters and drill into fields.



To easily access your dashboards, we recommend adding a shortcut to Sisense on your phone's screen.

For iPhone:

1. Open Safari and browse to your Sisense web app.
2. Tap on the Share icon at the bottom of the screen. 
3. Tap Add to Home Screen.
4. Edit the name of the icon, for example: *Sisense*, and tap Add in the top-right corner.

For Android:

1. Open your browser and browse to the Sisense web app.
2. Depending on your browser, you can do one of the following:
 - ▶ In your browser's menu, tap **Add to home screen**.
 - or
 - ▶ In your browser, add the page to your bookmarks (for example, in Chrome, click the  icon). Edit the bookmark's name, for example *Sisense*.

- ▶ Access your widget tray, tap and drag **Bookmark** to your home screen, and select the Sisense bookmark.