

precisely

Spectrum Discovery

User Guide

Version 2020.1.0



Table of Contents

1 - Getting Started

Understanding Your Organization's Data Assets...	5
A First Look at Spectrum Discovery.....	6

2 - Connecting to Data

Connections.....	12
Defining Connections.....	12
Compression Support for Cloud File Servers.....	65
Deleting a Connection.....	66
How to Video - Configuring Connections.....	66

5 - Modeling

Logical Model.....	69
Physical Model.....	73
Context Graph Model.....	80
Mapping Models.....	96
Model Store.....	121
How to Video - Modeling.....	132

6 - Query

Understanding the Home Page.....	134
Creating a Query.....	135
Exporting a Query.....	139
Scheduling a Query.....	141
How to Video - Query Business Entities.....	144

7 - Preparing Quality Rules

Accessing Prepare Home Page.....	147
----------------------------------	-----

Selecting a source.....	151
Creating Rules.....	156
How to Video - Prepare Dashboard.....	191
How to Video - Preparing Manual Matching and Consolidation Rule.....	191
How to Video - Preparing Smart Consolidation Rule.....	191
How to Video - Preparing Smart Match Rule....	191

8 - Profiling and Scorecarding

Profile & Monitor Home page.....	194
Generating Score cards.....	195
Editing a scorecard.....	208
Scheduling Scorecard Analysis.....	209
Viewing scorecard trends.....	211
Creating a Profile.....	211
Specifying Profiling Default Settings.....	226
Analyzing a Profile.....	227
Viewing Data Profiling Results.....	232
Collaborating on Data Profiling Results.....	239
How to Video - Profiling.....	240
How to Video - Applying Suggestions to Profiled Data.....	240
How to Video - Scorecard.....	240
How to Video - Consuming Spectrum Services in Profiling and Scorecard.....	240

9 - Lineage and Impact Analysis

Viewing Lineage and Impact Analysis.....	242
Icons in Lineage and Impact Analysis.....	243
Uses.....	248
How to Video - Lineage and Impact Analysis....	255

10 - Accessing Services through API

API calls for Glossary and Profiling services257

How to Video - Discovery APIs.....258

1 - Getting Started

In this section

Understanding Your Organization's Data Assets.....	5
A First Look at Spectrum Discovery.....	6



Understanding Your Organization's Data Assets

If your organization is like most, you have a large number of data assets containing everything from customer contact information and purchase history, to financial data, to transaction records, and more. These systems can run on different platforms, sometimes managed by different departments with different security controls. There is potentially a wealth of data available to you to answer business questions, but it is a challenge to figure out which systems contain the data you need, which systems to trust, and how they're all connected.

Spectrum Discovery provides the visibility you need to identify the most trusted sources of data to use to satisfy a business request.

1. Start by connecting the physical data assets in your organization to Spectrum Technology Platform. See [Connections](#) on page 12.
2. Then, define a physical data model to represent your data assets in Spectrum Discovery. Going through this process will help you understand how your data assets are structured, such as the tables and columns in each database, and the relationships between tables. See [Adding a Physical Data Model](#) on page 73.
3. With an understanding of the physical data assets available to you, you will want to make sure that the underlying data is of good quality. Use profiling to scan your data assets, identify the types of data contained in them (such as names, email addresses, and currency), and identify incomplete and malformed data. See [Creating a Profile](#) on page 211.

Tip: Using the reports from profiling, you can create Spectrum Technology Platform flows to improve data quality. If you have not licensed one of the data quality modules for Spectrum Technology Platform, contact your Precisely Account Executive.

4. With a physical data model created and a clear understanding of the state of your data through profiling, you can create logical models to represent the business entities that your business wants to understand, such as customers, vendors, or products. In this process you select the sources for the data you want to use to populate each entity, such as customer addresses and purchase history. See [Creating a New Logical Model](#) on page 69.
5. To maintain your data assets you need to understand how they're all connected and how data flows from source to destination. Use the lineage and impact analysis feature of Spectrum Discovery to view the dependencies between data sources, destinations, and the processes that use the data. With this information, you can make informed decisions about the impact of a change to data sources, troubleshoot unexpected results, and understand how Spectrum Technology Platform entities like flows, subflows, and Spectrum databases affect each other. For more information, see [Viewing Lineage and Impact Analysis](#) on page 242.

A First Look at Spectrum Discovery

Spectrum Discovery gives you the control you need to deliver accurate and timely data-driven insights to your business. Use Spectrum Discovery to develop data models, view the flow of data from source to business application, and assess the quality of your data through profiling. With this insight, you can identify the data resources to use to answer particular business questions, adapt and optimize processes to improve the usefulness and consistency of data across your business, and troubleshoot data issues.

To access Spectrum Discovery, open a web browser and go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

Spectrum Discovery functions are divided into these areas: discovery, modeling, profiling, and lineage and impact analysis.

Discovery

Metadata gives a context to the associated data and improves its understanding. It helps organizations take informed decisions. That is the reason most of the data driven organizations are moving towards metadata management.

Benefits of metadata management:

- Better understanding of the business data, its creation, manipulation, and the business rules associated with it.
- Ease in locating specific data in varied data set across the organization
- Tracking data upstream and downstream to know how a change impacts the link and workflow
- Ensuring data quality: Since metadata defines data, managing it correctly ascertains data correctness and consistency. It is the key to identifying malformed data and preventing database integrity issues.

Wide penetration of big data and data governance rules are other factors that have made metadata management a necessity.

How Discovery helps you manage metadata

The **Catalog** page of **Spectrum Discovery** gives you a unified view of all your configured connections. You can discover the assets from all the connections and perform these tasks on the discovered metadata:

- Search for specific assets (table, column, or view) in your discovered connections.
- Add tags to assets to give those a relevant context and make those easily accessible later.

Define Glossary

For data to make sense in an organizational context, it needs to be consistent, contextual, and transparent. Users through the organization need to have a common understanding of the definition and lineage of the data they are using. And what can be a better way to achieve this than arriving at the definitions, policies, and rules governing data collaboratively through work flows and multilevel approvals.

The **Define Glossary** page in Spectrum Discovery allow you to do all this with an intuitive, user-friendly interface.

Ease of use	Business Glossary helps you search for the required data without having to master the tables, column names, fields, and metadata.
Transparency	The data comes with its full context and lineage, so you can trust it without an iota of doubt.
Eliminates conflicts, confusions	You have a comprehensive view of all the business terms with the related data and metadata. This avoids all the conflicts in terms of meaning and its usage.
Swift, powerful, simple search	The advanced search features make searching swift, simple, and powerful. You can use any aspect of the data including relationship, tags, and attribute to search it.

These powerful features of Define Glossary make it most suitable for your business needs if you deal with data, and inconsistency in its definition across the enterprise is hindering you. If you are not able to find the right data, at the right time, and reconciling the differences in data definition in various departments is making life chaotic for you, ultimately eroding the trust in your data.

Modeling

The Modeling view is where you create physical and logical data models and deploy those into a model store, thus creating a layer of abstraction over the underlying data sources on the Spectrum Technology Platform server.

A *physical model* organizes your organization's data assets in a meaningful way. A physical model makes it possible to pull data from individual tables, columns, and views to create a single resource that you can then use to supply data to logical models or to perform profiling.

View Physical Model - phone_file

The interface displays a physical model for 'phone_file'. On the left, there is a search bar. In the center, a list of columns for 'Phone_data' is shown:

Column	Type
AddedBy	string
CustomerId	integer
DateAdded	string
DateUpdated	string
PhoneNumber	string
PhoneNumberType	string

On the right, the 'Properties' panel for 'Phone_data' is visible, showing a table of columns with their types, sizes, nullability, and updatability:

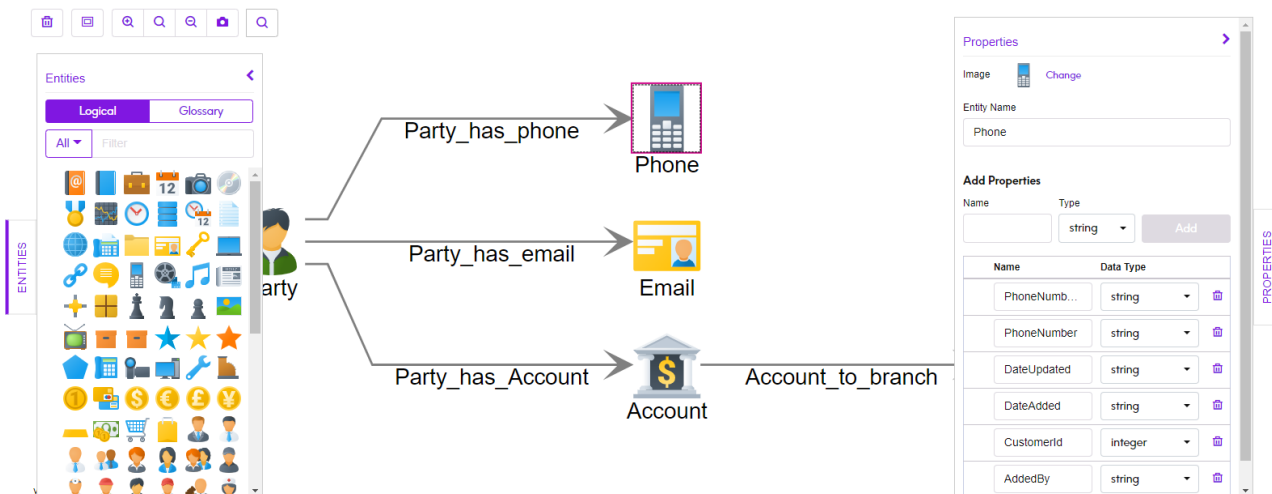
Name	Type	Size	Nullable	Updatable
AddedBy	string		<input type="checkbox"/>	<input checked="" type="checkbox"/>
CustomerId	integer		<input type="checkbox"/>	<input checked="" type="checkbox"/>
DateAdded	string		<input type="checkbox"/>	<input checked="" type="checkbox"/>
DateUpdated	string		<input type="checkbox"/>	<input checked="" type="checkbox"/>
PhoneNumber	string		<input type="checkbox"/>	<input checked="" type="checkbox"/>

Below the table, there are sections for 'Primary keys' and 'Foreign keys'.

A *logical model* defines the objects that your business is interested in and the attributes of those objects, as well as how objects are related to each other. For example, a logical model for a customer might contain attributes for name and date of birth. It might also have a relationship to a home address object, which contains attributes for address lines, city, and postal code. Once you have defined the attributes of the objects your business is interested in, you can map physical data sources to the logical model's attributes, thereby identifying the specific data asset that will be used to populate that attribute.

Edit Logical Model C_360_Banking

Save Cancel



Profiling

Successful decision making is heavily dependent on reliable, correct, complete, and valid data. Incomplete records, malformed fields, and a lack of context can result in misleading or inaccurate data being delivered to your business users, which can result in flawed decisions.

Data profiling scans your data records from all the data sources - irrespective of its volume and complexity. It identifies problems related to correctness, completeness, and validity in the data, and

suggests actions to fix the issues. Thus, it improves the quality and utility of your data with very little manual effort.

Benefits of data profiling

- It is the first step in analyzing your data and predicting how much effort is needed to make it usable
- It improves your trust on the data set you have
- It is one of the mandatory steps for taking control of your organizational data and using it across the enterprise.

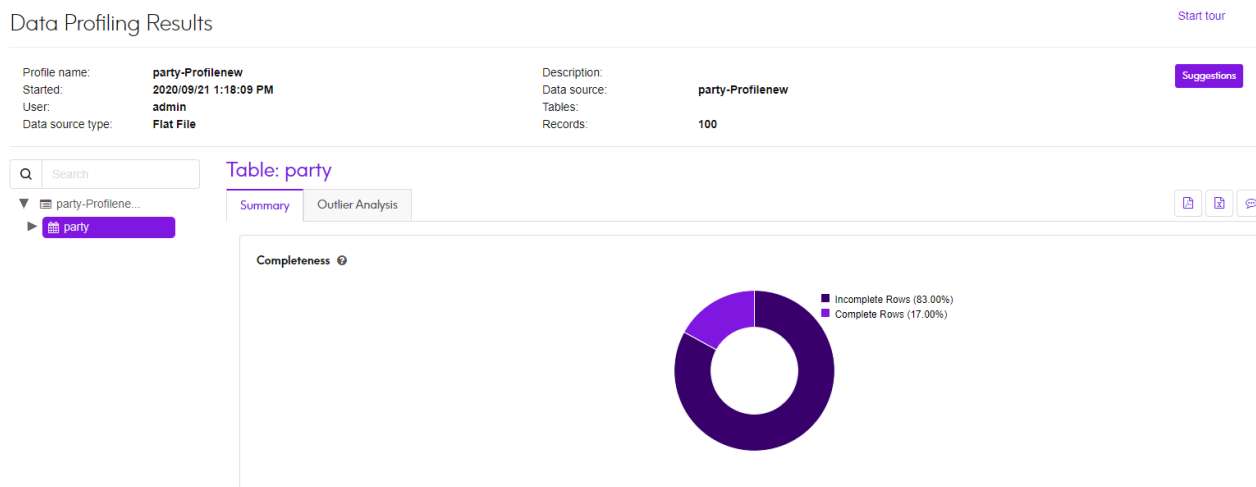
Scorecard

Monitoring is one of the indispensable aspects of data governance and good decision making. Constant monitoring of data quality helps you to take measures to improve the consistency, reliability, and accuracy of your data. Scorecards help you measure the quality of your data and assign score cards to it by tracking key matrices defined by you. It represents your data health in a graphical form, making assessment all the more easy and swift for you.

Trends

Trends help you measure the improvement in the quality of your data over a period of time. For example, in a merger between two companies (Company A and Company B), company A needs to migrate customer data in its system from company B. It will run a scorecard on the data at the beginning of the process to see the quality and may thereafter do this scoring after every cleansing. Trends plot a chart based on your recent runs or for a specified date range to give you a clearer picture on the improvement index.

Note: You need to have the **Data Integration** module installed in order to use the profiling feature of **Spectrum Discovery**.



Lineage and Impact Analysis

The Lineage and Impact Analysis view shows how data flows from data sources to data destinations and through Spectrum Technology Platform flows. Lineage and impact analysis are similar concepts that describe different ways of tracing the flow of data.

Lineage shows where data comes from. You can use it to trace the path of data back to its source, showing all the systems that process and store the data along the way, such as Spectrum Technology Platform flows, databases, and files.

Impact analysis shows where data goes and the systems that depend on data from a selected data resource. You can use it to view the flows, databases, and files that use a data resource directly or indirectly. Looking at impact analysis is useful if you want to understand how a modification to a database, file, or flow will affect the processes and systems that use the data.

Spectrum Discovery can show lineage and impact analysis in a single diagram that shows the complete flow of data from source to destination. You can also choose to view lineage only or impact only. By viewing data lineage and impact analysis together you can pinpoint issues in your data processes and plan for upgrades and modifications to your data processes.

2 - Connecting to Data

In this section

Connections.....	12
Defining Connections.....	12
Compression Support for Cloud File Servers.....	65
Deleting a Connection.....	66
How to Video - Configuring Connections.....	66



Connections

A connection is a database, file server, cloud service, or other source of data that you want to process through Spectrum Technology Platform. Spectrum Technology Platform can connect to over 20 types of data sources.

To connect Spectrum Technology Platform to a data source, you need to define the connection first, before you can define the input XML. Similarly, if you want to write dataflow output to a database, you must first define the database as an external resource.

Let's say, in your organization, data resides in disparate sources, such as Salesforce, Apache Cassandra, Hadoop, Dynamo DB, SQL server, as well as CSV files.

To access your data set:

1. You need to first connect to all these data sources. Spectrum Technology Platform allows you to create connections to all these and many more data sources, which you will see in the subsequent sub-sections.
2. Once you successfully establish a connection, you can access it in:
 - Various stages of **Spectrum Enterprise Designer**, such as:
 - **Read From DB**
 - **Read From File**
 - **Read from Hadoop Sequence File**
 - **Read From Hive File**
 - **Read from HL7 File**
 - **Read from NoSQL DB**
 - **Read from SAP**
 - **Read from Spreadsheet**
 - **Read from Variable Format File**
 - **Read From XML**
 - The Catalog, Modeling, and Profiling processes in **Spectrum Discovery**

Defining Connections

To define a new connection in Spectrum Technology Platform, use one of these interfaces:

- The **Connections** page in Management Console.
- **Connect** menu option of Spectrum Discovery

Note: If you want to read from or write to data located in a file on the Spectrum Technology Platform server itself there is no need to define a connection.

Connecting to Amazon

Connecting to Amazon DynamoDB

1. Access the **Connections** page using one of these:

Spectrum Management Console:	Access Spectrum Management Console using the URL: <code>http://server:port/management console</code> , where <i>server</i> is the server name or IP address of your Spectrum Technology Platform server and <i>port</i> is the HTTP port used by Spectrum Technology Platform.
-------------------------------------	--


Note: By default, the HTTP port is 8080.

Click **Resources > Connections**.

Spectrum Discovery:	Access Spectrum Discovery using the URL: <code>http://server:port/discovery</code> , where <i>server</i> is the server name or IP address of your Spectrum Technology Platform server and <i>port</i> is the HTTP port used by Spectrum Technology Platform.
----------------------------	--

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Amazon DynamoDB**.
5. In the **Access Key ID** field, enter the 20-character alpha-numeric sequence provided to you to access your Amazon AWS account.
6. In the **Secret Access Key** field, enter the 40-character key needed to authenticate the connection.
7. In the **Region** field, select the region of the Amazon AWS account.
8. To test the connection, click **Test**.

9. Click **Save**.

Amazon DynamoDB Limitations

1. Hierarchical data types like lists, sets and maps are interpreted as String data types. This is because these data types are not supported.
2. Null values in a DynamoDB data source are interpreted as empty column values.
3. The `count` aggregate function is not supported in query on Model Store.

Connecting to Amazon S3

1. Access the **Connections** page using one of these:

Spectrum Management Console:

Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Cloud**.
5. In the **Cloud service** field, choose **AmazonS3**.
6. In the **Bucket name** field, enter the bucket name as defined in your Amazon S3 cloud service. This is the bucket where Spectrum Technology Platform will read and write files.
7. Enter your access key and secret key assigned to you by Amazon.
8. In the **Storage Type**, field select the level of redundancy that you want to allow for data storage.

Standard

The default level of redundancy provided by Amazon S3.

Reduced redundancy Stores non-critical and easily-reproducible data at lower levels of redundancy. This provides fairly reliable storage at a lower cost.

9. In the **Encryption** section, select the encryption method for the data. You can select server side encryption, client side encryption, or both.

Server side key The data is encrypted and decrypted at the server side. Your data is transmitted in plain text to the Amazon cloud service where it is encrypted and stored. On retrieval, the data is decrypted by the Amazon cloud service then transmitted in plain text to your system.

You have two options for specifying the key:

- **AWS managed:** The key is automatically generated by the Amazon S3 cloud service.
- **Customer provided:** Enter the key to be used by the Amazon S3 cloud service to encrypt and decrypt the data on the server side.

Client side key The data is encrypted and decrypted at the client side. The data is encrypted locally on your client system then transmitted to the Amazon S3 cloud storage. On retrieval, the data is transmitted back in an encrypted format to your system and is decrypted on the client system.

Client side key: Enter the key to be used by your client system to encrypt and decrypt the data.

If you select both **Server side key** and **Client side key**, encryption and decryption is performed at both server and client sides. Data is first encrypted with your client side key and transmitted in an encrypted format to Amazon, where it is again encrypted with the server side key and stored. On retrieval, Amazon first decrypts the data with the server side key, transmitting the data in an encrypted format to your system, where it is finally decrypted with the client side key.

Note: To use the encryption feature of Amazon S3 cloud, you need to install the Amazon S3 Security JAR files. For more information, see [Using Amazon S3 Cloud Encryption](#) on page 16.

For more information about Amazon S3 encryption features, see:

docs.aws.amazon.com/AmazonS3/latest/dev/UsingEncryption.html

10. If you want to set access permissions, in the **Permissions** section, click .

The three kinds of Grantees are:

Everyone	Every one else other than Authenticated Users and Log Delivery group.
AuthenticatedUsers	For users who are logged into Amazon.
LogDelivery	For users who write activity logs in a user-specified Bucket, if Bucket Logging is enabled.

For each Grantee, select the desired permissions:

Open/Download	Allow the user to download the file.
View	Allow the user to view the current permissions on the file.
Edit	Allow the user to modify and set the permissions on the file.

11. To test the connection, click **Test**.
12. Click **Save**.

Using Amazon S3 Cloud Encryption

To use the encryption security feature of the Amazon S3 cloud service, you need to download security JAR files and place them on the Spectrum Technology Platform server. Using encryption is optional.

1. Go to the download site.

For Windows and Linux platforms using Java 7, the JAR files can be downloaded from:

<http://www.oracle.com/technetwork/java/javase/downloads/jce-7-download-432124.html>

For AIX platforms using Java 7, the JAR files can be downloaded from:

<https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=jcesdk>

2. Download these two JAR files:

- local_policy.jar
- US_export_policy.jar

3. Place the JAR files in the location:

```
%JAVA_HOME%\jre\lib\security
```

4. Restart the server.

Connecting to Amazon SimpleDB

1. Access the **Connections** page using one of these:

Spectrum Management Console:	Access Spectrum Management Console using the URL: http://server:port/management console , where <i>server</i> is the server name or IP address of your Spectrum Technology Platform server and <i>port</i> is the HTTP port used by Spectrum Technology Platform.
-------------------------------------	---

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:	Access Spectrum Discovery using the URL: http://server:port/discovery , where <i>server</i> is the server name or IP address of your Spectrum
----------------------------	--

Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Amazon SimpleDB**.
5. In the **Access Key ID** field, enter the 20-character alpha-numeric sequence provided to you to access your Amazon AWS account.
6. In the **Secret Access Key** field, enter the 40-character key needed to authenticate the connection.
7. To test the connection, click **Test**.
8. Click **Save**.

Amazon SimpleDB Limitations

Write Limitation

In the Write to DB stage, the write mode **Update** is not available when writing to an Amazon SimpleDB table. The **Insert** option handles both insert and update operations. It differentiates between an insert and an update using the unique value of the `ItemName` column which is present in all Amazon SimpleDB tables.

Reason: An update query requires a Primary Key for each record of the table to be updated, which is not supported by Amazon SimpleDB databases.

Read Limitation

The aggregate functions `SUM` and `AVG` are not supported while executing queries on Model Store.

Connecting to Apache Cassandra

1. Access the **Connections** page using one of these:

Spectrum Management Console:

Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Apache Cassandra**.
5. In the **Host** field, enter the machine name or the IP on which the Apache Cassandra database is installed.
6. In the **Keyspace** field, enter the name of the keyspace of the data center you wish to access.
7. In the **Port** field, enter the port on which the Apache Cassandra database is configured.
8. Enter the user name and password to use to authenticate to the Cassandra database.
9. In the **Consistency Level** field, select how consistent data rows must be across replica nodes for a successful data transaction. This can be at least one, or all, or a combination of available nodes.
10. In the **Fetch Size**, enter the number of resultset rows you wish to fetch on each read transaction.
11. To test the connection, click **Test**.
12. Click **Save**.

Apache Cassandra Limitation

The `count` aggregate function is not supported in query on Model Store.

Connecting to Azure Cloud

1. Access the **Connections** page using one of these:

Spectrum Management Console: Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.


Note: By default, the HTTP port is 8080.

Click **Resources > Connections**.

Spectrum Discovery: Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Cloud**.
5. In the **Cloud service** field, choose **AzureBlobStorage**.
6. In the **Protocol** field select whether you want the connection between Azure and Spectrum Technology Platform to use HTTP or HTTPS.
7. In the **Account Name** field, enter the name of your Azure storage account.
8. In the **Access Key** field, enter the access key to your Azure account.
9. To test the cloud connection, click **Test**.
10. Click **Save**.

Connecting to Context Graph

This procedure describes how to use a Context Graph model as a data source. You can use the Context Graph connector to read entity and relationship model data to be used in Spectrum Discovery.

1. Access the **Connections** page using one of these:

**Spectrum
Management
Console:**

Access Spectrum Management Console using the URL:
`http://server:port/management console`, where *server* is the server name
 or IP address of your Spectrum Technology Platform server and *port* is
 the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

**Spectrum
Discovery:**

Access Spectrum Discovery using the URL: `http://server:port/discovery`,
 where *server* is the server name or IP address of your Spectrum
 Technology Platform server and *port* is the HTTP port used by Spectrum
 Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Data Hub**.
5. In the **Model Name** field, enter the Context Graph model name.
6. To test the connection, click **Test**.
7. Click **Save**.

Context Graph connector limitation

- You must specify the column name in the `count` aggregate function as `count(column_name)`.
 You cannot use the wildcard, as in `count(*)`.

Connecting to a Flat File

Connecting to a Delimited Flat File

1. Access the **Connections** page using one of these:

Spectrum Management Console:

Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:


Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** box, select Flat File.
5. Enter the **File Path** by clicking the **Browse** button  and locating the file.

Note: While building a model store on a flat file connection, *periods* (.) if any, in the column name automatically gets converted to *underscores* (_).

6. Select the **Character Encoding** of the flat file from the drop-down.
7. Select the **Record Type** as Delimited.
8. In **Field Delimiter**, select the expected separator between any two fields of a file record.
If the delimiter does not appear on the list, click the Add field delimiter  button to define the delimiter.
9. Optional: If there is one, select the **Text Qualifier** that encloses field values.
10. In **Line Separator**, the value `Default` is selected, indicating that the expected line separator depends on whether Spectrum Technology Platform is running on a Windows system.
11. To specify whether the first row of the file is a header row, shift the **First row is header row** slider to either **ON** or **OFF**.
12. To specify whether the data type of the various fields in any record of the file should be automatically detected, shift the **Detect data type from file** slider to either **ON** or **OFF**.
13. To skip malformed records during file parsing, shift the **Skip Malformed Records** slider to **ON**.

14. Click the **Preview** button to view field values based on selections you made in previous steps of this procedure.
15. Click **Test**.
A message confirms the successful test of the connection.
16. Click **Save**.
A message confirms the successful creation of the connection.

In order to view a sample record fetched using the created Delimited Flat File connection, click **Preview** in the header bar. File records will be fetched and the Fields sorted according to the details provided by you.

Connecting to a Fixed-Width Flat File

1. Access the **Connections** page using one of these:

Spectrum Management Console: Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.


Note: By default, the HTTP port is 8080.

Click **Resources > Connections**.

Spectrum Discovery: Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** box, select Flat File.
5. Enter the **File Path** by clicking **Browse** and selecting the directory of the file.
6. Select the **Character Encoding** of the flat file from the drop-down.
7. Select the **Record Type** as Fixed Width.
8. In the **Record Length** field, enter the total number of characters in a file record.
9. Click **Add Field** to add a row for a field in a file record.

10. In the **Name** column, enter the name for the field value.
11. In the **Type** column, select the data type of the field value.
12. In the **Start Position** column, enter the position in the file record at which of the field value begins.
For the first field in a file record, the **Start Position** counting begins from 1.
13. In the **Length** field, enter the total number of characters the field covers, including the character at the **Start Position**.
The sum of the **Start Position** and **Length** values for any field should be less than or equal to the **Record Length**
If the File Record is:

```
01234Rob Smith29Precisely
```

Record Length = 25

For the field 'Name':

Start Position = 6

Length = 9

```
Name = Rob Smith
```

14. Check the **Trim** check box if you wish to trim any white spaces at the beginning and/or end of a field value.
15. Repeat step 10 through step 15 to add details for all fields expected in file records.
16. Click the **Preview** button to view field values based on selections you made in previous steps of this procedure.
17. Click **Test**.
A message confirms the successful test of the connection.
18. Click **Save**.
A message confirms the successful creation of the connection.

In order to view a sample record fetched using the created Fixed Width Flat File connection, click **Preview** in the header bar. File records will be fetched and the Fields sorted according to the details provided by you.

Date Time Formats in a File Connection

Date and time values read from files using a File Connection in Spectrum Technology Platform need to adhere to date-time formats described here.

Accepted Date Time Formats

- Date: *yyyy-mm-dd"*
- Datetime: *yyyy-mm-dd HH:mm:ss*

- Time: *HH:mm:ss*

Delimited Files

If the **Detect type** feature is turned on while configuring the Delimited File Connection, then the date and time values in the file records, which adhere to the above formats, are automatically detected as Date type.

If a date-time value does not adhere to one of the accepted formats, the value is read as a String type value instead of a Date type value.

Fixed Width Files

For Fixed Width files, date type values are configured while creating the Fixed Width File Connection. Hence these values are read as Date type values, irrespective of whether they adhere to the accepted formats or not.

If the date-time value in a Fixed Width file does not adhere to the accepted formats, it needs to be handled using **Transformations** at the logical model creation stage by applying this *Conversion* category function to the value:

```
parsedate(String date, String format)
```

In this, the *date* is the value received from the file, while the *format* is the date-time format in which the value is received from the file. This helps to parse the date-time value correctly.

For example, if the date = 23-Feb-2008, then the format = dd-MMM-yyyy.

Resulting Value Formats

While previewing data in a model store:

- If the value has been read as a date/time value, it is reflected in one of the accepted date/time formats in the preview.
- If the value has been read as a String value, it is reflected as it is in the preview.

Connecting to an FTP Server

In order for Spectrum Technology Platform to access files on an FTP server you must define a connection to the FTP server using Spectrum Management Console. Once you do this, you can create dataflows in Spectrum Enterprise Designer that can read data from, and write data to, files on the FTP server.

Before connecting to an FTP server, verify that the timeout settings on your FTP server are appropriate for the jobs that will use this connection. Depending on the design of a job, there may be periods of time when the connection is idle, which may cause the connection to time out. For example, you

may have a dataflow with two Read from File stages connected to an Import To Model stage. While the Import To Model stage is reading records from one Read from File stage, the other will be idle, possibly causing its connection to the FTP server to time out. Consider setting the timeout value on your FTP server to 0 to prevent connections from timing out.

Note: The FTP server must be running in active connection mode. Passive connection mode is not supported.

1. Access the **Connections** page using one of these:

Spectrum Management Console: Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.


Note: By default, the HTTP port is 8080.

Click **Resources > Connections**.

Spectrum Discovery: Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **FTP**.
5. In the **User name** and **Password** fields, enter the credentials to use to authenticate to the FTP server. This is required only if the FTP server requires it.
6. In the **Host** field, enter the host name or IP address of the FTP server.
7. In the **Port** field, enter the network port number the server uses for FTP.
8. Click **Test** to verify that the Spectrum Technology Platform server can connect to the FTP server.
9. Click **Save**.

Connecting to an SFTP Server

In order for Spectrum Technology Platform to access files on an SFTP server you must define a connection to the SFTP server using Spectrum Management Console.

Before connecting to an SFTP server, verify that the timeout settings on your SFTP server are appropriate for the jobs that will use this connection. Depending on the design of a job, there may be periods of time when the connection is idle, which may cause the connection to time out. For example, you may have a dataflow with two Read from File stages connected to an Import To Model stage. While the Import To Model stage is reading records from one **Read from File** stage, the other will be idle, possibly causing its connection to the SFTP server to time out. Consider setting the timeout value on your SFTP server to 0 to prevent connections from timing out.

Note: The SFTP server must be running in active connection mode. Passive connection mode is not supported.

1. Access the **Connections** page using one of these:

**Spectrum
Management
Console:**

Access Spectrum Management Console using the URL:
http://*server:port*/management console, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

**Spectrum
Discovery:**

Access Spectrum Discovery using the URL: http://*server:port*/discovery, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **SFTP**.
5. In the **Host** field, enter the host name or IP address of the SFTP server.
6. In the **Port** field, enter the network port number that the server uses for SFTP. The default is 22.

7. Toggle the **Strict Host Key Checking** button to **Yes**, if you want to enable it. Default is **No**. If the strict host key checking is enabled, *ssh* does not automatically add host keys to the known host file. This is an additional security feature.
8. In the **Known Host File** field, browse the location of the file that maintains known hosts details, and select it.

Note: If you disable **Strict Host Key Checking**, this field is not displayed. However, this detail is mandatory if strict host key checking is enabled.

9. Enter the **Username**.
10. Select the preferred **Authentication Type**. It can be *Password* or *Key-Based*. Default is *Password*.
 - **Password:** If you selected this as the authentication type, **Password** field is displayed below the **Authentication Type** field. Enter the required password in this field.
 - **Key-Based:** If you selected a key-based authentication, browse and select the **Private Key File**, and enter the **Passphrase** shared by the host server administrator. The pass phrase is needed only if the key is configured using this option.
11. Click **Test** to verify that the Spectrum Technology Platform server can connect to the SFTP server.
12. Click **Save**.

Connecting to Google Cloud Storage

1. Access the **Connections** page using one of these:

Spectrum Management Console:

Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

- Click the Add connection button .
- In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

- In the **Connection Type** field, choose **Cloud**.
- In the **Cloud service** field, choose **GoogleCloudStorage**.
- In the **Bucket name** field, enter the bucket name as defined in your Google cloud service. This is the bucket where Spectrum Technology Platform will read and write files.
- Enter your the application name, service account, and private key file provided by Google.

Note: Ensure the private key file is present on the Spectrum Technology Platform server.

- You can set access permissions in the **Permissions** section.

Manage your data and permission	Allows the user to manage the data and permissions.
View your data	Allows the user to view data.
Manage your data	Allows the user to manage data.

- To test the connection, click **Test**.
- Click **Save**.

For more information, see Google's [Service Account Authentication](#) documentation.

Connecting to Hadoop

Connect to the Hadoop system to use the stages, such as [Read from Hadoop Sequence File](#), [Write to Hadoop Sequence File](#), [Read From File](#), [Write to File](#), [Read From XML](#), [Write to XML](#), [Read From Hive File](#), [Write to Hive File](#), and [Read from HL7 File](#), in **Spectrum Enterprise Designer**.

Attention: Spectrum Technology Platform does not support *Hadoop 2.x* for Kerberos on Windows platforms.

Follow these steps to connect to the Hadoop system:

- Access the **Connections** page using one of these:

Spectrum Management Console:	Access Spectrum Management Console using the URL: <code>http://server:port/management console</code> , where <i>server</i> is the server name or IP address of your Spectrum Technology Platform server and <i>port</i> is the HTTP port used by Spectrum Technology Platform.
-------------------------------------	--

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **HDFS**
5. In the **Host** field, enter the host name or IP address of the NameNode in the HDFS cluster.
6. In the **Port** field, enter the network port number.
7. In **User**, select one of these options:

Server user	Choose this option if authentication is enabled in your HDFS cluster. This option will use the user credentials that the Spectrum Technology Platform server runs under to authenticate to HDFS.
User name	Choose this option if authentication is disabled in your HDFS cluster.
8. Check **Kerberos** if you wish to enable Kerberos authentication feature for this HDFS file server connection.
9. If you have opted to enable **Kerberos** authentication, then enter the path of the keytab file in the **Keytab file path** field.

Note: Ensure the key tab file is placed on the Spectrum Technology Platform server.

10. In the **Protocol** field, select one of:

WEBHDFS	Select this option if the HDFS cluster is running HDFS 1.0 or later. This protocol supports both read and write operations.
HFTP	Select this option if the HDFS cluster is running a version older than HDFS 1.0, or if your organization does not allow the WEBHDFS protocol. This protocol only supports the read operation.

HAR Select this option to access Hadoop archive files. If you choose this option, specify the path to the archive file in the **Path** field. This protocol only supports the read operation.

11. Expand the **Advanced options**.

12. If you selected the WEBHDFS protocol, you can specify these advanced options as required:

Replication factor Specifies how many data nodes to replicate each block to. For example, the default setting of 3 replicates each block to three different nodes in the cluster. The maximum replication factor is 1024.

Block size Specifies the size of each block. HDFS breaks up a file into blocks of the size you specify here. For example, if you specify the default 64 MB, each file is broken up into 64 MB blocks. Each block is then replicated to the number of nodes in the cluster specified in the **Replication factor** field.

File permissions Specifies the level of access to files written to the HDFS cluster by Spectrum Technology Platform. You can specify read and write permissions for each of these options:



Note: The *Execute* permission is not applicable to Spectrum Technology Platform.

User This is the user specified above, either **Server user** or the user specified in the **User name** field.

Group This refers to any group of which the user is a member. For example, if the user is john123, then Group permissions apply to any group of which john123 is a member.

Other This refers to any other users as well as groups of which the specified user is not a member.

13. Use the **File permissions** descriptions below to define the server properties for Hadoop to ensure that the sorting and filtering features work as desired when the connection is used in a stage or activity. To add properties, complete one of these steps:

- Click  and add the properties and their respective values in the **Property** and **Value** fields.
- Click  and upload your configuration XML file. The XML file should be similar to `hdfs-site.xml`, `yarn-site.xml`, or `core-site.xml`.

Note: Place the configuration file on the server.

File permissions and parameters - Hadoop 1.x

This section applies to this stage and activity:

- Stage - **Read from Sequence File**
- Activity - **Run Hadoop Pig**

fs.default.name

Specifies the node and port on which Hadoop runs. For example,
`hdfs://152.144.226.224:9000`

mapred.job.tracker

Specifies the host name or IP address, and port on which the MapReduce job tracker runs. If the host name is entered as local, then jobs are run as a single map and reduce task. For example, `152.144.226.224:9001`

dfs.namenode.name.dir

Specifies where on the local files system a DFS name node should store the name table. If this is a comma-delimited list of directories, then the name table is replicated in all of the directories, for redundancy. For example,
`file:/home/hduser/Data/namenode`

hadoop.tmp.dir

Specifies the base location for other temporary directories. For example,
`/home/hduser/Data/tmp`

File permissions and parameters - Hadoop 2.x

This section applies to this stage and activity:

- Stage - **Read from Sequence File**
- Activity - **Run Hadoop Pig**

fs.defaultFS

Specifies the node and port on which Hadoop runs. For example,
`hdfs://152.144.226.224:9000.`

NOTE: For Spectrum versions 11.0 and earlier, the parameter name `fs.defaultfs` must be used. Note the case difference. For versions 11 SP1 and later, both the names `fs.defaultfs` and `fs.defaultFS` are valid. We recommend using parameter name `fs.defaultFS` for releases 11.0 SP1 and later.

yarn.resourcemanager.resource-tracker.address

Specifies the host name or IP address of the Resource Manager. For example,
`152.144.226.224:8025`

yarn.resourcemanager.scheduler.address

Specifies the address of the Scheduler Interface. For example,
`152.144.226.224:8030`

yarn.resourcemanager.address

Specifies the address of the Applications Manager interface that is contained in the Resource Manager. For example, `152.144.226.224:8041`

mapreduce.jobhistory.address

Specifies the host name or IP address, and port on which the MapReduce Job History Server is running. For example, `152.144.226.224:10020`

mapreduce.application.classpath

Specifies the CLASSPATH for Map Reduce applications. This CLASSPATH denotes the location where classes related to Map Reduce applications are found. The entries should be comma separated.

For example:

```
$HADOOP_CONF_DIR, $HADOOP_COMMON_HOME/share/hadoop/common/*,  
$HADOOP_COMMON_HOME/share/hadoop/common/lib/*,  
$HADOOP_HDFS_HOME/share/hadoop/hdfs/*,  
$HADOOP_HDFS_HOME/share/hadoop/hdfs/lib/*,  
$HADOOP_MAPRED_HOME/share/hadoop/mapreduce/*,  
$HADOOP_MAPRED_HOME/share/hadoop/mapreduce/lib/*,  
$HADOOP_YARN_HOME/share/hadoop/yarn/*,  
$HADOOP_YARN_HOME/share/hadoop/yarn/lib/*
```

mapreduce.app-submission.cross-platform

Handles various platform issues that arise if your Spectrum server runs on a Windows machine, and you install Cloudera on it. If your Spectrum server and Cloudera are running on different Operating Systems, then enter the value of this parameter as `true`. Otherwise, mark it as `false`.

Note: Cloudera does not support Windows clients. Configuring this parameter is a workaround, and not a solution to all resulting platform issues.

File permissions and parameters - Kerberos

This section applies to this stage and activity:

- Stage - **Read from Sequence File**
- Activity - **Run Hadoop Pig**

If you have selected the **Kerberos** check box, add these Kerberos configuration properties:

hadoop.security.authentication

The type of authentication security being used. Enter the value `kerberos`.

yarn.resourcemanager.principal

The Kerberos principal being used for the resource manager for your Hadoop YARN resource negotiator. For example: `yarn/_HOST@HADOOP.COM`

dfs.namenode.kerberos.principal

The Kerberos principal being used for the namenode of your Hadoop Distributed File System (HDFS). For example, `hdfs/_HOST@HADOOP.COM`

dfs.datanode.kerberos.principal

The Kerberos principal being used for the data node of your Hadoop Distributed File System (HDFS). For example, `hdfs/_HOST@HADOOP.COM`

File permissions and parameters - Hadoop 1.x

This section applies to these stages:

- Stage **Read from File**
- Stage **Write to File**
- Stage **Read from Hive ORC File**
- Stage **Write to Hive ORC File**

fs.default.name

Specifies the node and port on which Hadoop runs. For example,
`hdfs://152.144.226.224:9000`

File permissions and parameters - Hadoop 2.x

This section applies to these stages:

- Stage **Read or write from File**
- Stage **Read or write from Hive ORC File**

fs.defaultFS

Specifies the node and port on which Hadoop runs. For example,
`hdfs://152.144.226.224:9000`

NOTE: For Spectrum versions 11.0 and earlier, the parameter name `fs.defaultfs` must be used. Note the case difference. For versions 11 SP1 and later, both the names `fs.defaultfs` and `fs.defaultFS` are valid. We recommend using parameter name `fs.defaultFS` for releases 11.0 SP1 and later.

14. To test the connection, click **Test**.

15. Click **Save**.

After you have defined a connection to an HDFS cluster, it becomes available in source and sink stages in Spectrum Enterprise Designer, such as Read from File and Write to File. You can select the HDFS cluster when you click **Remote Machine** when defining a file in a source or sink stage.

Compression Support for Hadoop

Spectrum Technology Platform supports the compression formats `gzip` (.gz) and `bzip2` (.bz2) on Hadoop. While using the **Read from File** and **Write to File** stages with an HDFS connection, include the extension corresponding to the required compression format (.gz or .bz2) in the **File name** field. The file is decompressed or compressed based on the specified compression extension. Spectrum Technology Platform handles the compression and decompression of the files.

Connecting to Hive

You can connect to Hive databases through the driver provided by Spectrum Technology Platform or by adding the Apache JDBC driver. The driver provided by Spectrum Technology Platform

(*hive-jdbc-1.2.2-batch-18.2.jar*) is an extended version of Apache Hive driver and supports batch processing. It is located here:

SpectrumDirectory\server\modules\bigdata\drivers\hive, where *SpectrumDirectory* is the folder where you have installed the Spectrum Technology Platform server.

For information about manually adding a JDBC driver, see [Manually Adding a JDBC Driver](#) on page 36.

Note: The class path for the Hive JDBC driver is:
com.pb.spectrum.hive.jdbc.batch.HiveBatchDriver

Connecting to a JDBC Database

Define a connection using the **Connections** page. You can go to this page through **Spectrum Management Console** or through **Spectrum Discovery**.

1. Access the **Connections** page using one of these:


Spectrum Management Console:	Access Spectrum Management Console using the URL: <code>http://server:port/management console</code> , where <i>server</i> is the server name or IP address of your Spectrum Technology Platform server and <i>port</i> is the HTTP port used by Spectrum Technology Platform.
-------------------------------------	---

Note: By default, the HTTP port is 8080.

Spectrum Discovery:	Click Resources > Connections . Access Spectrum Discovery using the URL: <code>http://server:port/discovery</code> , where <i>server</i> is the server name or IP address of your Spectrum Technology Platform server and <i>port</i> is the HTTP port used by Spectrum Technology Platform.
----------------------------	---

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, select type of database you want to connect to.

Spectrum Data Federation includes JDBC drivers for SQL Server, Oracle, and PostgreSQL databases. If you want to connect to a different database type, you must add the JDBC driver before defining a connection.

5. In the **URL** field, enter the JDBC connection URL. Your database administrator can provide this URL.

For example, to connect to a MySQL database named "SampleDatabase" hosted on a server named "MyServer" you would enter:

```
jdbc:mysql://MyServer/SampleDatabase
```

6. There may be additional fields you need to fill in depending on the JDBC driver. The fields represent properties in the connection string for the JDBC driver you selected in the **Type** field. See the JDBC driver provider's documentation or your database administrator for information about the specific connection properties and values required by the connection type.
7. Click **Save**.
8. Test the connection by checking the box next to the new connection and clicking the **Test** button.

Importing a JDBC Driver

Spectrum Technology Platform can access data from any database using a JDBC driver. Drivers for SQL, Oracle, and PostgreSQL are provided with the Spectrum Data Federation, which also includes drivers for other types of databases. If Spectrum Technology Platform does not come with a driver for the type of database you need, you can add a JDBC driver.

In this procedure you will import a JDBC driver by copying the driver files to the Spectrum Technology Platform server. When you complete this procedure the driver will be available to use when defining a JDBC database connection in Spectrum Management Console.

Note: This procedure works for *JDBC 4.x* drivers. If the driver you want to add uses an older version of JDBC you must add the driver manually in Spectrum Management Console. For more information, see [Manually Adding a JDBC Driver](#) on page 36

1. Put all the JDBC driver files for the database into a folder named:

```
Name.jdbc
```

Where *Name* is any name you want. The folder name must end with `.jdbc`.

2. Log in to the server running Spectrum Technology Platform.
3. Copy the folder containing the driver to this folder:

```
SpectrumDirectory\server\drivers
```

The driver is automatically imported.

4. To verify that the driver was successfully imported, log in to Spectrum Management Console and go to **System > Drivers**. The driver should be listed.


If the driver is not listed, open the System Log in Spectrum Management Console and look for errors related to deploying JDBC drivers.

Manually Adding a JDBC Driver

Spectrum Technology Platform can access data from any database using a JDBC driver. Drivers for SQL, Oracle, and PostgreSQL are provided with the Spectrum Data Federation, which also includes drivers for other types of databases. If Spectrum Technology Platform does not come with a driver for the type of database you need, you can add a JDBC driver.

In this procedure you will add JDBC driver files to the server then manually define the connection string and connection properties. Before you begin, be sure that you understand the connection string format and properties required by the driver. You must define these accurately in order for the driver to function. You can typically find information about a driver's connection string and properties from the driver provider's website.

Note: We recommend that you use this procedure only when adding a JDBC driver that uses *JDBC 1.x*, *2.x*, or *3.x*. If the driver uses *JDBC 4.x*, we recommend that you use the import method to add the driver. For more information, see [Importing a JDBC Driver](#) on page 35.

1. Open Spectrum Management Console.
2. Go to **System > Drivers**.
3. Click the Add button .
4. In the **Name** field, enter a name for the driver. The name can be anything you choose.
5. In the **JDBC driver class name** field, enter the Java class name of the driver. You can typically find the class name in your JDBC driver's documentation.

For example, to use the Microsoft JDBC driver, you might enter the following:

```
com.microsoft.sqlserver.jdbc.SQLServerDriver
```

6. In the **Connection string template** field, enter the JDBC connection URL to use to connect to the database, including any properties you want to set in the connection string. Different database vendors use different connection strings so check your database's documentation for more information about the connection string.

If the driver will be used by more than one database connection, consider using property tokens in the connection string instead of hard-coding property values that may be different for each connection. For example, if you want to have some connections use encryption and others not, you may want to define a property token for the encryption property.

To use a property token in the connection string, use this syntax:

```
${PropertyToken}
```

Any property tokens you include in the connection string template will be required fields when defining a database connection.

Note: Use the property token name `${password}` for the property that will contain the database password. By using this token name, the password will be masked in the field in Spectrum Management Console and will be encrypted in the database.

For example, this connection string for SQL contains property tokens for host, port, instance, and encryption:

```
jdbc:sqlserver://${host}:${port};databaseName=${instance};encrypt=${encryption};TrustServerCertificate=true
```

These tokens are required fields when defining a database connection that uses this driver:

[Home](#) > [Resources: Connections](#) > [Add Connection](#)

Add Connection

*Connection Name


*Connection Type

*Host

Port

Instance

Encryption

7. If there are properties that you want to make optional for database connections, define them in the **Connection Properties** section.
 - a) In the **Connection properties** section, click the Add button .
 - b) In the **Label** field, enter a user-friendly description of the property. The label you enter here is used as the field label in the connections window when creating a connection using this driver.
 - c) In the **Property token** field, enter the token for the optional property. See the database driver's documentation for the properties supported by the driver.

Note: Use the property token name `password` for the property that will contain the database password. By using this token name, the password will be masked in the field in Spectrum Management Console and will be encrypted in the database.

For example, if you want to make encryption optional for database connections that use this driver, you could define the encryption property like this:

Home > System: Drivers > Edit Driver

Edit Driver

*Name

com.mysql.jdbc.Driver.5.1

* JDBC driver class name ⓘ

com.mysql.jdbc.Driver

* Connection string template ⓘ

jdbc:mysql://\$(host)/\$(instance)

Properties & Drivers

Connection properties ⓘ



Label	Property Token
<input type="checkbox"/> username	user
<input type="checkbox"/> password	password
<input type="checkbox"/> Use SSL	useSSL

When a database connection uses this driver, the encryption property would be displayed as an optional property in the database connection:

Home > Resources: Connections > Add Connection

Add Connection

*Connection Name

MyConnection

*Connection Type

com.mysql.jdbc.Driver.5.1


*Host

User Name

Password

Use SSL

Test

- Log in to the server running Spectrum Technology Platform and place the database driver file in a folder on the server. The location does not matter.
- In the **Driver files** section, click the Add button .
- In the **File path** field, enter the path to the database driver file on the server.
- Click **Save**.

Deleting an Imported JDBC Driver

JDBC drivers cannot be deleted using Spectrum Management Console if the JDBC driver was imported to Spectrum Technology Platform rather than being added manually in Spectrum Management Console. Instead, follow this procedure to delete the driver.

Important: Before deleting a driver, verify that there are no database connections using the driver.

1. Stop the Spectrum Technology Platform server.
2. Go to this folder:

`SpectrumDirectory\server\drivers`

3. In the `drivers` folder, delete folder containing the driver.
4. Start the Spectrum Technology Platform server.
5. To verify that the driver has been deleted, log in to Spectrum Management Console, go to **System > Drivers**, and verify that the driver is no longer listed.

Supported Database Data Types

Spectrum Technology Platform supports these data types commonly used in databases:

bigdecimal	A numeric data type that supports 38 decimal points of precision. Use this data type for data that will be used in mathematical calculations requiring a high degree of precision, especially those involving financial data. The bigdecimal data type supports more precise calculations than the double data type.
boolean	A logical type with two values: true and false.
date	A data type that contains a month, day, and year. For example, 2012-01-30 or January 30, 2012. You can specify a default date format in Spectrum Management Console.
datetime	A data type that contains a month, day, year, and hours, minutes, and seconds. For example, 2012/01/30 6:15:00 PM.
double	A numeric data type that contains both negative and positive double precision numbers between 2^{-1074} and $(2-2^{-52}) \times 2^{1023}$. In E notation, the range of values is -1.79769313486232E+308 to 1.79769313486232E+308.
float	A numeric data type that contains both negative and positive single precision numbers between 2^{-149} and $(2-2^{-23}) \times 2^{127}$. In E notation, the range of values -3.402823E+38 to 3.402823E+38.
integer	A numeric data type that contains both negative and positive whole numbers between -2^{31} (-2,147,483,648) and $2^{31}-1$ (2,147,483,647).
long	A numeric data type that contains both negative and positive whole numbers between -2^{63} (-9,223,372,036,854,775,808) and $2^{63}-1$ (9,223,372,036,854,775,807).
string	A sequence of characters.
time	A data type that contains the time of day. For example, 21:15:59 or 9:15:59 PM.

Raw An Oracle data type for storing variable length binary data. Maximum size is 2000 bytes (the maximum length in Oracle 7 was 255 bytes).

Other database data types are automatically mapped to one of the supported data types as follows:

Database Data Type	Supported Data Type
Date/Time Types	
TIMESTAMP	datetime
String Types	
CHAR	string
CLOB	string
LONGVARCHAR	string
NCHAR	string
NVARCHAR	string
VARCHAR	string
Numeric Types	
BIGINT	long
DECIMAL	double
FLOAT	double
NUMERIC	bigdecimal
REAL	float
SMALLINT	integer
TINYINT	integer

Database Data Type	Supported Data Type
Boolean Types	
BIT	boolean

Supported Database Data Types for Spectrum Spatial

These database data types are automatically mapped to one of the supported data types for Spectrum Spatial.

Database Data Type	Supported Data Type
SQL Server	
tinyint	SHORT_INTEGER
smallint	SHORT_INTEGER
int	INTEGER
bigint	LONG_INTEGER
float	DOUBLE
real	DOUBLE
decimal(10, 5)	DOUBLE
numeric(10, 5)	DOUBLE
date	DATE
time	TIME
datetime	DATE_TIME
smalldatetime	DATE_TIME
char(10)	STRING

Database Data Type	Supported Data Type
varchar(10)	STRING
nchar(10)	STRING
nvarchar(10)G	STRING
binary(10)	BINARY
varbinary(10)	BINARY
PostGIS	
smallint	SHORT_INTEGER
integer	INTEGER
bigint	LONG_INTEGER
numeric(10, 5)	DOUBLE
real	DOUBLE
double precision	DOUBLE
serial	INTEGER
bigserial	LONG_INTEGER
bytea	BINARY
date	DATE
time	TIME
timestamp	DATE_TIME
character(10)	STRING

Database Data Type	Supported Data Type
character varying(10)	STRING
nchar(10)	STRING
Oracle	
NUMBER	DOUBLE
CHAR(10)	STRING
VARCHAR(10)	STRING
VARCHAR2(10)	STRING
NCHAR(10)	STRING
NVARCHAR2(10)	STRING
DATE	DATE_TIME
TIMESTAMP	DATE_TIME
BLOB	BINARY

JDBC Database connector limitations

- MongoDB/Cassandra connectors are not supported via PrestoDB in Spectrum Discovery. There are separate connectors for MongoDB and Cassandra.
- Write to Any DB via Presto is not recommended by Presto DB and so is not supported by Presto JDBC connector.

Connecting to Knox

An Apache Knox Gateway allows you to access a Hadoop service through the Knox security layer.

With this connection, you can create flows in the Spectrum Enterprise Designer using stages in Enterprise Big Data to read data from and write data to Hadoop via Knox.

1. Access the **Connections** page using one of these:

Spectrum Management Console:	Access Spectrum Management Console using the URL: <code>http://server:port/management console</code> , where <i>server</i> is the server name or IP address of your Spectrum Technology Platform server and <i>port</i> is the HTTP port used by Spectrum Technology Platform.
-------------------------------------	--


Note: By default, the HTTP port is 8080.

Click **Resources > Connections**.

Spectrum Discovery:	Access Spectrum Discovery using the URL: <code>http://server:port/discovery</code> , where <i>server</i> is the server name or IP address of your Spectrum Technology Platform server and <i>port</i> is the HTTP port used by Spectrum Technology Platform.
----------------------------	--

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose *Gateway*.
5. In the **Gateway Type** field, choose **Knox**.
6. In the **Host** field, enter the host name or IP address of the node in the HDFS cluster running the gateway.
7. In the **Port** field, enter the port number for the Knox gateway.
8. In the **User Name** field, enter the user name for the Knox gateway.
9. In the **Password** field, enter the password to authorize you access to the Knox gateway.
10. In the **Gateway Name** field, enter the name of the Knox gateway you wish to access.
11. In the **Cluster Name** field, enter the name of the Hadoop cluster to be accessed.
12. In the **Protocol** field, choose *webhdfs*.
13. In the **Service Name** field, enter the name of the Hadoop service to be accessed.
14. To test the connection, click **Test**.
15. Click **Save**.

After you have defined a Knox connection to an HDFS cluster, the connection can be used in Spectrum Enterprise Designer, in the stages **Read from File** and **Write to File**. You can select the HDFS cluster when you click **Remote Machine** when defining a file in a source or sink stage.

Connecting to a Windows Mapped Drive

When Spectrum Technology Platform is running on a Windows server, it can access data on the server's mapped drives. Since the Spectrum Technology Platform server runs as a Windows service under a particular user account (often the Local System account) you need to define the mapped drive in the server's start-up process in order for it to be visible in Spectrum Enterprise Designer and Spectrum Management Console.

1. Stop the Spectrum Technology Platform server.
2. Under the folder where the Spectrum Technology Platform server is installed, go to `server\bin\wrapper`. For example, `C:\Program Files\Precisely\Spectrum\server\bin\wrapper`.
3. Open the file `wrapper.conf` in a text editor.

Important: In the following steps you will add new properties to this file. It is important that you follow these instructions precisely and only add and modify the properties described in the following steps. Do not modify any of the other properties in this file.

4. Add these lines:

```
wrapper.share.1.location
wrapper.share.1.target
wrapper.share.1.type
wrapper.share.1.account
wrapper.share.1.password
```

5. In the `wrapper.share.1.location` property, specify the location of the mapped drive in UNC format.

Note: Do not include a trailing backslash in the UNC.

For example,

```
wrapper.share.1.location=\\myserver\share
```

6. In the `wrapper.share.1.target` property, specify the drive letter to assign to this mapped drive.

For example,

```
wrapper.share.1.target=Y:
```

7. In the `type` property, specify `DISK`.

For example,

```
wrapper.share.1.type=DISK
```

8. If the share you are connecting to requires a user name and password, specify the user name in the `wrapper.share.1.account` property and specify the password in the `wrapper.share.1.password` property.

For example,

```
wrapper.share.1.account=domain\user123
wrapper.share.1.password=mypassword1
```

Note: If the Spectrum Technology Platform server service is running under the Local System user, you cannot specify a user name and password. If the share requires a user name and password you must modify the service to run under a different account.

Example

This example shows two mapped drives being defined in the `wrapper.conf` file.

```
wrapper.share.1.location=\\myserver\data
wrapper.share.1.target=Y:
wrapper.share.1.type=DISK
wrapper.share.1.account=sample\user
wrapper.share.1.password=samplepass
wrapper.share.2.location=\\myserver\moredata
wrapper.share.2.target=Z:
wrapper.share.2.type=DISK
wrapper.share.2.account=sample\user
wrapper.share.2.password=samplepass
```

Connecting to Marketo

1. Access the **Connections** page using one of these:

Spectrum Management Console:

Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


**Spectrum
Discovery:**

Click **Resources > Connections**.

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

- Click the Add connection button .
- In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

- In the **Connection Type** field, choose **Marketo**.
- In the **Endpoint URL** field, enter the endpoint URL of your Marketo account.

To find your endpoint URL, log in to your Marketo account and go to **Admin > Integration > Web Services**. The endpoint URL is under the heading **REST API** and is in this format:

`https://AccountID.mktorest.com/rest`

Copy the portion of the URL before `/rest`. For example, `https://AccountID.mktorest.com`.

- Enter the client ID and secret key for your Marketo account.
To find your client ID and secret key, log in to your Marketo account and go to **Admin > Integration > LaunchPoint > API Rest > View Details**. The pop-up window displays the details.
- To test the connection, click **Test**.
- Click **Save**.

Marketo Limitations

- This query applies only to `List` and `Activity_type` entities. For others, provide the filter type.

```
Select * from Marketo_Table
```

- Does not support joins except between `Lead` and `Lead_List` entities. The join query between `Lead` and `Lead_List` for a `List_Id` is as follows:

```
Select Lead.* from Lead Inner Join Lead_List
On Lead.ID = Lead_List.Lead_ID
And Lead_List.List_ID = <List ID>
```

Supported Entities and Operations

The entities are of these types:

1. Entity
2. Entity Update: This is a virtual table used for Update on Lead entity. For example, **Merge_Leads** should be used for merging different Marketo Leads.

Connecting to Microsoft Dynamics 365

Connecting to Microsoft Dynamics 365 Online

Spectrum Technology Platform supports connecting to Microsoft Dynamics 365 (version 9) only.

1. Access the **Connections** page using one of these:

Spectrum Management Console: Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.


Note: By default, the HTTP port is 8080.

Click **Resources > Connections**.

Spectrum Discovery: Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Microsoft Dynamics 365**.
5. In the **Deployment Type** field, select **Online**.
6. In the **Username** field, enter your Microsoft Dynamics user name.

7. In the **Password** field, enter your Microsoft Dynamics password.
8. In the **Organization Name** field, enter your organization unique name, which identifies your CRM instance.

To find your organization unique name, log in to Microsoft Dynamics and go to **Settings > Customization > Customizations > Developer Resources**. Your organization unique name is displayed.

9. In the **Region** field, select the geographical region of your Microsoft Dynamics account.
10. To test the connection, click **Test**.
11. Click **Save**.

Connecting to Microsoft Dynamics 365 On Premise

This connector from Spectrum Technology Platform supports claims based authentications for Microsoft Dynamics 365 On Premises.

Prerequisites:

Import certificate to the keystore file: To import the Dynamics CRM server certificates to the Spectrum Java distribution Keystore, perform these tasks:

1. Copy the server certificates to a local folder
2. Browse this path to the Spectrum JAVA distribution:
`SpectrumDirectory\JavaLocation\jre\lib\security`
3. Run this command to import the certificates:
 - Windows - `keytool -importcert -alias certificatealiasname -file "certificatepath\certificatename" -keystore keystore.jks`
 - Linux - `keytool -import -alias certificatealiasname -file "certificatepath/certificatename" -keystore keystore.jks`.

Defining Microsoft Dynamics 365 On Premise connection

Perform these steps to define a **Microsoft Dynamics 365 On Premises** connection:

1. Access the **Connections** page using one of these:

Spectrum Management Console:	Access Spectrum Management Console using the URL: <code>http://server:port/management console</code> , where <i>server</i> is the server name or IP address of your Spectrum Technology Platform server and <i>port</i> is the HTTP port used by Spectrum Technology Platform.
-------------------------------------	---

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

**Spectrum
Discovery:**

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** box, lick Microsoft Dynamics 365.
5. In the **Deployment Type** box, click **On Premise**.
6. Enter your Microsoft Dynamics user name in the **Username**.
7. Enter your Microsoft Dynamics password in the **Password**.
8. Enter the name of the host in the **Host Name**.
9. Enter the name of the port In the **Port Name**.
10. Enter the URL of the STS in the **STS URL**.
11. Click **Test** to test the connection.
12. Click **Save**.

Limitations

Create/Update: Create/Update can fail if a column in an Entity is mapped to multiple Reference Entities. For example, `ParentCustomerId` in `Customer` can be associated to `Account`, `Lead`, and others. To resolve this, the data for this column needs to be in the following format:

`ReferenceEntityName:GUID` in place of `GUID`.

Supported Entities and Operations

The entities are of these types:

- User-owned
- Organization-owned
- Business-owned
- None

Connecting to a Model Store

Connect to a model store to use the data federated from various sources such as databases, file servers, and cloud services. Once a connection is defined, you can use the data in the logical and physical models of a model store (created and deployed in Spectrum Discovery) in the **Read from DB** and **Write to DB** stages of Spectrum Enterprise Designer.

1. Access the **Connections** page using one of these:

**Spectrum
Management
Console:**

Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

**Spectrum
Discovery:**

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Model Store**.
5. In the **Model store** field, enter the name of the model store that you are establishing connection with.

To find the names of the available model stores, access **Spectrum Discovery**, go to **Model**, and click the **Model Store** tab.

6. To test the connection, click **Test**.
7. Click **Save**.

Note: Using the **Write to DB** stage over a Model Store connection has certain limitations, such as *Create Table*, *Truncate table before inserting data*, and *Drop and recreate the table if it already exists* not being supported.

Connecting to NetSuite

While reading from and writing to a NetSuite connection, both interactive and batch modes are supported. Spectrum Technology Platform supports these NetSuite entity types:

- Standard records
- Custom records
- Saved searches
- Joins between Standard records

To connect to NetSuite:

1. Access the **Connections** page using one of these:

**Spectrum
Management
Console:**

Access Spectrum Management Console using the URL:
http://*server:port*/management console, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

**Spectrum
Discovery:**

Access Spectrum Discovery using the URL: http://*server:port*/discovery, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **NetSuite**.

5. In the **Email** field, enter the email addressed linked to the NetSuite account to be used for the connection.
6. In the **Password** field, enter the password of the NetSuite account.
7. In the **Account** field, enter the user name for the NetSuite account.
8. In the **Role** field, select the appropriate role for this connection from the roles mapped to the particular NetSuite user account.

The **Role** field is optional. If you leave the **Role** field blank, the default role is used to log in through the connection.

Attention: Only standard roles are supported. Custom roles are not supported.

9. To test the connection, click **Test**.
10. Click **Save**.

Note: To `INSERT` a record using a NetSuite connection, use an `UPSERT` query with the primary key (`internalId`) blank.

NetSuite Limitations

- When querying using joins, you must cite specific columns. For example, this query is not supported:

```
select * from CUSTOMER_M
```

- Simultaneous connections to NetSuite are not supported because NetSuite allows only a single sign in for one account.
- You can only write Standard and Custom records.
- For both `UPDATE` and `UPSERT` queries, an `UPSERT` operation is performed.
- In the Write to DB stage, the maximum batch size permitted for an `insert` operation is 200 and the maximum batch size for an `update` operation is 100.

Supported Entities and Operations

The entities are of these types:

- Standard records
- Custom records
- Joins
- Saved searches

Note: In a NetSuite connection table, the primary key column is `internalId`.

Connecting to NoSQL

The supported NoSQL database types are:

- Couchbase
- MongoDB

Connect to the Hadoop system to use the stages, such as **Read from Hadoop Sequence File**, **Write to Hadoop Sequence File**, **Read From File**, **Write to File**, **Read From XML**, **Write to XML**, **Read From Hive File**, **Write to Hive File**, and **Read from HL7 File**, in **Spectrum Enterprise Designer**.

Attention: Spectrum Technology Platform does not support *Hadoop 2.x* for Kerberos on Windows platforms.

- Query NoSQL DB
- Read from NoSQL DB
- Write to NoSQL DB

1. Access the **Connections** page using one of these:

**Spectrum
Management
Console:**

Access Spectrum Management Console using the URL:
http://*server:port*/management console, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

**Spectrum
Discovery:**

Access Spectrum Discovery using the URL: http://*server:port*/discovery, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, select any one of:

- Couchbase
 - MongoDB
5. Specify the **Host**, **Port**, **Database**, **Username** and **Password** of the specific NoSQL database you wish to access.
 6. Click **Test** to check that the connection to the database is successful.
 7. Click **OK**.

Connecting to Salesforce

1. Access the **Connections** page using one of these:

**Spectrum
Management
Console:**

Access Spectrum Management Console using the URL:
`http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

**Spectrum
Discovery:**

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Salesforce**.
5. In the **Username** field, enter the email ID registered on the Salesforce data store.
6. In the **Password** field, enter a combination of the Salesforce portal password and the security token generated through the Salesforce portal.

For example, if your password is Sales@Test, and the security token provided to you by Salesforce is 56709367, then the Password to authenticate this Salesforce connection would be Sales@Test56709367.

- Set the **Use default endpoint** toggle to **No** if you want to use a specific endpoint URL to access the Salesforce data. Enter the required URL in the **Salesforce URL** field displayed just below the **Use default endpoint** toggle.

Note: The **Salesforce URL** is mandatory.

- Set the **Use bulk read** toggle to **Yes** if you want to fetch bulk data from Salesforce. Default is **No**.

Note: Query for bulk data fetching does not work for address and geolocation compound fields. For more information about the considerations and limitations of bulk query, see [Compound Field Considerations and Limitations](#) and [Bulk API Limits](#).

- To test the connection, click **Test**.
- Click **Save**.

Note: Audit fields are enabled on all tables by default. The Salesforce audit fields are:

- created date
- last modified date
- created by
- last modified by

Attention: Physical model created in Spectrum Technology Platform version 10 and earlier using Salesforce connections need to be opened and saved again in order to enable audit fields on their tables.

Salesforce Limitation

The `Aggregate functions` are not supported while executing queries on Model Store.

Connecting to SAP NetWeaver

Creating a SAP NetWeaver connection in Management Console using OData Services allows you to read, write and synchronize your CRM and ERP data. While reading from and writing to a SAP connection, both interactive and batch modes are supported.

To define a SAP NetWeaver connection, perform these steps:

- Access the **Connections** page using one of these:

**Spectrum
Management
Console:**

Access Spectrum Management Console using the URL:
`http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **SAP**.
5. In the **Username** field, enter the user name to access the SAP web service.
6. In the **Password** field, enter the password of the SAP web service.
7. In the **OdataURL** field, enter the address of the Odata web service to be used for this connection.
8. Click **Test**.
A message confirms the successful test of the connection.
9. Click **Save**.
A message confirms the successful creation of the connection.

Note: To perform fetch operations, an OData service must support the `$skip` and `$top` operations. If the service does not support these operations, the fetched records show inconsistencies in the model store preview.

SAP NetWeaver Limitations

For both `UPDATE` and `UPSERT` operations, an `UPDATE` operation is performed.

Supported Entities and Operations

The entity columns are of two types:

- Native: Columns with native data types are displayed with their respective data types
- Custom-defined: Columns with custom-defined data types are displayed with a blank data type

To deploy a model store derived from an SAP connection, ensure its logical and physical models include only such entities whose columns are of native data types. If the models have entities of custom-defined data types, the model store cannot be deployed.

Connecting to SharePoint

1. Access the **Connections** page using one of these:

Spectrum Management Console: Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.


Note: By default, the HTTP port is 8080.

Click **Resources > Connections**.

Spectrum Discovery: Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **Cloud**.
5. In the **Cloud service** field, choose **Sharepoint**.
6. In the **Version** field, select **v2010**. Spectrum Technology Platform currently supports SharePoint version 2010.
7. In the **Protocol** field, select the protocol required to connect SharePoint.
8. In the **Server address** field, enter the host name or IP address of the SharePoint server you want to connect to.
9. Enter the user name and password to use to authenticate to SharePoint.
10. In the **Project** field, enter the specific project whose SharePoint location you want to access.
11. To test the connection, click **Test**.

- Click **Save**.

Example

For example, say you want to create a connection to this SharePoint URL:

```
https://sharepoint.example.com/sites/myportal
```

You would fill in the **Protocol**, **Server address**, and **Project** fields as follows:

- **Protocol:** https
- **Server address:** sharepoint.example.com
- **Project:** myportal

Connecting to Splunk

- Access the **Connections** page using one of these:

Spectrum Management Console:

Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

- Click the Add connection button .
- In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.
- Note:** Once you save a connection you cannot change the name.
- In the **Connection Type** field, choose **Splunk**.
- In the **Username** field, enter the Splunk account user name to authenticate the Splunk instance.

6. In the **Password** field, enter the password of the Splunk account.
7. In the **Host** field, enter the address or host name of the server on which the Splunk data source is hosted.
8. In the **Port** field, enter the port number of the Splunk data source.
9. To test the connection, click **Test**.
10. Click **Save**.

Splunk Limitations

This query is not supported:

```
select count(*) from SplunkTable
```

Supported Entities and Operations

Supported Operations

LIKE, ORDER BY, LIMIT, IN, BETWEEN, !=, <=, >=, <,>, multiple AND/OR operators.

Supported Functions

- String Functions: upper, lower, length, len, ltrim, rtrim, substring, max, min
- Mathematical Functions: abs, ceil, exp, floor, sqrt, round

Note: For all other query operations, use the Splunk `search` column as explained below.

Spectrum Technology Platform provides a column `search` in the Splunk table using which you can look up the required data in the Splunk connection.

While executing a `select` query on the `SplunkTable`, use the `search` column in the `where` clause in any of these scenarios:

1. To include such search criteria which cannot be specified using ANSI SQL syntax.
2. To include such Splunk-specific search criteria which cannot be included as part of the main SQL query.

For example, this query looks for such a `_raw` value which contains the key `opp` with the value `ACC`.

```
select "_raw" from SplunkTable where "search"='search opp=ACC'
```

Connecting to SugarCRM

1. Access the **Connections** page using one of these:

Spectrum Management Console:

Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

2. Click the Add connection button .
3. In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

4. In the **Connection Type** field, choose **SugarCRM**.
5. Enter your SugarCRM user name and password.
6. In the **URL** field, enter the URL of the SugarCRM account to be used for this connection.
7. Enter the **Client Id** and **Client Secret** of your SugarCRM account.
8. To test the connection, click **Test**.
9. Click **Save**.

SugarCRM Limitations

1. For both `UPDATE` and `UPSERT` queries, an `UPSERT` operation is performed.
2. The **Nullable** and **Updatable** columns in the table properties, as displayed in the **physical model Schema** of the connection, may not represent the correct operation. For example, a column not marked as Updatable may throw system exception when you try to update it and conversely, a column marked as Nullable might not throw an exception in updating.

- When querying using joins you need to use an alias.

Supported Entities and Operations

Supported Operations

LIKE (its operation is limited to picking up options starting with the specified value, such as in the statement `WHERE name LIKE 's%'`, which picks up all the names starting with the alphabet S.), **IS NULL**, **IS NOT NULL**, **IN**, **NOT IN**, **>**, **>=**, **<**, **<=**, **=**, **<>**, **AND**, **OR**

Connecting to Oracle Eloqua

- Access the **Connections** page using one of these:

Spectrum Management Console:

Access Spectrum Management Console using the URL: `http://server:port/management console`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.


Click **Resources > Connections**.

Spectrum Discovery:

Access Spectrum Discovery using the URL: `http://server:port/discovery`, where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port used by Spectrum Technology Platform.

Note: By default, the HTTP port is 8080.

Click **Connect**.

- Click the Add connection button .
- In the **Connection Name** box, enter a name for the connection. The name can be anything you choose.

Note: Once you save a connection you cannot change the name.

- In the **Connection Type** field, click **Oracle Eloqua**.
- Enter the **Site Name** which is same as Company Name.
- Enter the user name in the **Username** field.
- Enter the password in the **Password** field.

8. Click **Test** to test the connection..
9. Click **Save**.

Special Operations

- Use this join query to fetch contacts in a Contact List:

```
select * from Contacts inner join ContactListMembers on
Contacts.Eloqua_Contact_ID = ContactListMembers.Contact_Id where
ContactListMembers.ContactList_Id = '<id>'
```

Use this join query to fetch contacts in a Contact Segment:

```
select * from Contacts inner join ContactSegmentMembers on
Contacts.Eloqua_Contact_ID = ContactSegmentMembers.Contact_Id where
ContactSegmentMembers.Contactlist_Id = '<id>'
```

- Use this statement to insert contacts in a Contact List:

```
insert into ContactListMembers (ContactList_ID,Contact_ID) values
('<contactlist_id>','<contact_id>')
```

- Use this statement to delete contacts from a Contact List:

```
delete from ContactListMembers where ContactList_ID = '<contactlist_id>'
and Contact_ID = '<contact_id>'
```

Limitations

- **Create/Update:**
 - Insert/Upsert fails if Not Null columns are blank or do not exist
 - Insert/Upsert fails if values of Unique columns are not unique for a particular batch
 - In order to avoid a rollback exception, keep the value of **Batch count to commit** to 1
- **Read:** For custom entities, Select operation is only applicable on joins with Contacts entity
- **Filter:**
 - Supported filters are =, !=, >, <, >=, <=
 - There is no support for IN and NOT IN condition operators when providing more than one values
 - There is no support for Joins between entities
 - There is support for AND and OR operation conditional operators only for Accounts and Contacts entities
 - There is support for only AND conditional operator. It does not work for rest of the entities on the column
 - = filter does not always work on fields having timestamp data type

Supported Entities and Operations

These entities are supported:

- **Entity:** Denotes a table representing business entity.
- **Activity:** Denotes a table representing a business entity where data is generated based on some activity.
- **Custom Entity:** Denotes entities that are used as part of special operations provided with the Connector.

This table lists the entities and the operations supported for these.

Entity Name	Create	Read	Update	Delete	Batch Support	Maximum Batch Size
Accounts	X	X	X	X	Insert/Update*	1000
Account Groups		X				
Campaign		X				
Contacts	X	X	X	X	Insert/Update*	1000
Contact List	X	X	X	X		
Contact Segment	X	X	X	X		
Emails		X				
Email Folders		X				
Email Groups		X				
Microsites		X				
Users		X				
Visitors		X				
Activity						
Email Open		X				

Entity Name	Create	Read	Update	Delete	Batch Support	Maximum Batch Size
EmailClickthrough		X				
Email Send		X				
Subscribe		X				
Unsubscribe		X				
Bounceback		X				
WebVisit		X				
PageView		X				
FormSubmit		X				
Custom Entities						
ContactListMembers	X	X		X	Insert/Delete	1000
ContactSegmentMembers		X				

* Update operation works as Insert.

Compression Support for Cloud File Servers

The file servers Amazon S3, Google cloud storage, and MS Azure Blobstore support the compressed formats `gzip (.gz)` and `zip (.zip)`.

The Spectrum Technology Platform handles the compression and decompression of the files written to and read from the file servers.

Note: You can use the same file server to handle both normal reads and writes of files as well as compression and decompression of files.

Reading a Compressed Format File

While reading a file from the server, its compression format is derived from the metadata key property `Content-Encoding` received from the server.

Writing a Compressed Format File

While writing a file to a server, mention the required compression format: `.gz` or `.zip`. The file is compressed based on the specified compression extension.

The metadata key property `Content-Encoding` is also set according to the selected compression format. This property value is passed to the cloud file server while writing the file to it.

Deleting a Connection

You can delete the connection using any of these modules:

- Spectrum Management Console
- Spectrum Discovery

1. Access the **Connections** page of the required process.
 - In Spectrum Management Console, click **Resources** > **Connections**.
 - In Spectrum Discovery, click **Connect**.
2. Check the box next to the connection you want to delete, then click the Delete connection button



How to Video - Configuring Connections

This video describes how to connect to the various types of data sources and use them in the Spectrum Technology Platform.

<https://www.youtube.com/embed/Tp-GRa-iZAY>

5 - Modeling

Producing data-driven insights requires that technical teams and business teams have a common understanding of the organization's data assets and how those assets will be used to support business decisions. Technical teams understand the design of databases while business teams understand the business objects (such as customer, store, or vendor) that are of interest. Spectrum Discovery helps bridge this gap by providing physical modeling and logical modeling tools that are visually rich and independent of each other, enabling you to create both a technical view of data assets and a business view of objects of interest, and to link the two through mapping.

Physical Model

A *physical model* organizes your organization's data assets in a meaningful way. A physical model makes it possible to pull data from individual tables, columns, and views to create a single resource that you can then use to supply data to logical models or to perform profiling.

Logical Model

A *logical model* defines the objects that your business is interested in and the attributes of those objects, as well as how objects are related to each other. For example, a logical model for a customer might contain attributes for name and date of birth. It might also have a relationship to a home address object, which contains attributes for address lines, city, and postal code. Once you have defined the attributes of the objects your business is interested in, you can map physical data sources to the logical model's attributes, thereby identifying the specific data asset that will be used to populate that attribute.

Mapping

The built-in suggestion capabilities of Spectrum Discovery make mapping of logical to physical models an intuitive, effortless job. During the mapping process, the system suggests all the discovered tables and columns that map with the selected logical entities. For the suggested assets the semantic and data type are also displayed giving you the much needed context.

In this section

Logical Model.....	69
Physical Model.....	73


Context Graph Model.....	80
Mapping Models.....	96
Model Store.....	121
How to Video - Modeling.....	132

Logical Model

A *logical model* is a representation of business-specific entities. It defines the objects that your business is interested in and the attributes of those objects, as well as how these objects are related to each other.

Creating a New Logical Model

A logical model is created in the **Modeling** category of the Spectrum Discovery application. You define a logical model by adding business entities to it, defining relationships among the entities, and defining properties of the entities.

1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Click the Add Logical Model button .
This opens **Create Logical Model** with an empty canvas.
3. Enter a name for the model in the **Name** field.
4. From the **Entities** palette, select whether to use the **Logical** entities or the entities defined in the **Glossary** to create the model.

Tip: Use the drop-down list in the palette to filter **Logical** or **Glossary** entities based on categories.

5. Drag an entity to the canvas.

The entity is added to the canvas and its properties are displayed in the **Properties** panel on the right side of the page.

6. Enter a name for the entity in the **Entity Name** field.
By default, entities of the same type are named in ordinal sequence (for example, "Document", "Document1", "Document2", ...).
7. To define a new property for an entity:
 - a) Select the entity, and if necessary click the **PROPERTIES** tab to expand the **Properties** panel.
 - b) On the **Properties** panel, enter details for the selected entity:

Field	Description
Name	Name of the attribute, for example, age.

Field	Description
Semantic Type	Semantic type of the data for this attribute. Note: When you select the semantic type, the related data type is displayed in the Type field.
Type	Data type for the attribute.

- c) Click the **Add** button. The defined attribute, and its semantic and data type are displayed below the entry fields.

Note: You can delete a property by clicking the Delete button  adjacent to it.


8. To create relationships among entities, perform these steps:

Note: An entity can have multiple relationships to other entities as well as to itself.

To add a relationship to	Do this
Another entity	Drag from the entity to the other entity.
The same entity	Drag from the entity to itself.

A relationship is indicated by a relationship link pointing from the primary entity to the related entity, or to the same entity for a self-relationship. By default, relationships are named in ordinal sequence (relation, relation1, relation2, ...).

Note: As in case of entities, you can add or edit properties for a relationship in the **Properties** panel. For more information, see step [#unique_5/unique_5_Connect_42_d151e113](#).

9. To delete an entity or relationship, select it and click the Delete  button.

When you delete an entity, any relationships with it are also deleted.



10. Click **Save**.

The logical model is saved, and you are navigated back to the **Logical Model** tab of the **Modeling** page. The details displayed for this logical model are:

Name	The name of the logical model.
Default Mapping	This value is populated only after you map your logical model to the physical models. For more information, see Mapping Models - Introduction on page 96.
Last Modified	Last modified date and time of the logical model.

Actions


Actions that you can perform on a logical model.

- Click Create mapping  to map your logical model to the physical models.
- Click Create Context Graph Model  to create a Context Graph model based on the logical model. For more information, see [Context Graph Model](#) on page 80.

Creating a Logical Model from an Existing Logical Model

You can copy an existing logical model and use it as a template to create a new logical model.

Use this procedure when you want to create a logical model that is a variation of an existing logical model.


1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Select the check box next to a logical model and click the Copy Model button .
3. In the **Copy Logical Model** window, rename the model in the **Name** field.
4. Select **Yes** in the **Copy Mappings** slider to copy all the mappings along with the logical model.
5. Click **Save**.

The copied model now shows on the **Logical Model** tab. The details displayed for this logical model are:

You can now rename and edit the structure of new model. For more information, see [Editing a Logical Model](#) on page 71.

Editing a Logical Model

You can modify details of a logical model using the **Edit Logical Model** page.

1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Select the check box next to the model you want to modify.
Note: You can search for a specific model using the **Filter** field on the top of the page. It searches the models by the model name.
3. Click the Edit Model button .
4. You can edit the name for the model in the **Name** field.
5. From the **Entities** palette, select whether to use the **Logical** entities or the entities defined in the **Glossary** to create the model.

Tip: Use the drop-down list in the palette to filter **Logical** or **Glossary** entities based on categories.

6. Drag an entity to the canvas.

The entity is added to the canvas and its properties are displayed in the **Properties** panel on the right side of the page.

7. Enter a name for the entity in the **Entity Name** field.

By default, entities of the same type are named in ordinal sequence (for example, "Document", "Document1", "Document2", ...).

8. To define a new property for an entity:

- a) Select the entity, and if necessary click the **PROPERTIES** tab to expand the **Properties** panel.
- b) On the **Properties** panel, enter details for the selected entity:

Field	Description
Name	Name of the attribute, for example, age.
Semantic Type	Semantic type of the data for this attribute. Note: When you select the semantic type, the related data type is displayed in the Type field.
Type	Data type for the attribute.

- c) Click the **Add** button. The defined attribute, and its semantic and data type are displayed below the entry fields.

Note: You can delete a property by clicking the Delete button  adjacent to it.


9. To create relationships among entities, perform these steps:

Note: An entity can have multiple relationships to other entities as well as to itself.

To add a relationship to	Do this
Another entity	<u>Drag from the entity to the other entity.</u>
The same entity	<u>Drag from the entity to itself.</u>

A relationship is indicated by a relationship link pointing from the primary entity to the related entity, or to the same entity for a self-relationship. By default, relationships are named in ordinal sequence (relation, relation1, relation2, ...).

Note: As in case of entities, you can add or edit properties for a relationship in the **Properties** panel. For more information, see step 8 on page 72.

10. To delete an entity or relationship, select it and click the Delete  button.

When you delete an entity, any relationships with it are also deleted.

11. Click **Save**.


The selected model is updated and listed in the **Models** tab of the **Modeling** page.

Deleting a Logical Model

If you delete a logical model which has a dependent physical model, the dependent physical model becomes unusable.

1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Select the check box next to the model you want to modify.

Note: You can search for a specific model using the **Filter** field on the top of the page. It searches the models by the model name.

3. Click the **Delete Model** button .
4. Click **Yes** to confirm.

Physical Model

A *physical model* organizes your organization's data assets in a meaningful way. A physical model makes it possible to pull data from individual tables, columns, and views to create a single resource that you can then use to supply data to logical models or to perform profiling.


Adding a Physical Data Model

Physical models are created by selecting the tables and columns you want to include from the data source.


- You can create more than one physical model from one data source (connection). For example, a model for a system administrator will include the system tables from the data source, while the model created for end users will have the required data tables from the same data source.

- You cannot use multiple data sources to create a physical model.

To create a physical model, perform these actions:

1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Click the **Physical Model** tab.
3. Click the Add Physical Model button .

You can also choose to create a new model by copying an existing model and renaming it:

- a) Select the check box next to the existing model and click on the Copy Model button .
- b) If necessary, rename the model on the **Copy Physical Model** page.
- c) Click the **Save** button.

For more information, see [Editing a Physical Model](#) on page 75.

4. In the **Name** field, enter a name for the model.
5. In the **Source connection type** field, choose the type of data source that you want to model, then select the specific data source in the **Source connection name** field.

The **Source connection name** field displays the data sources defined for the selected connection type. For example, if you select Apache Cassandra in the **Source connection type** field, the **Source connection name** field lists all the Apache Cassandra data sources you have defined in the system.

In case you have not already defined a connection or want to create a new connection for this physical model, you can do this using the **Add Data Sources** page. Access this page through the **Data Sources** tab or via **Management Console** module.

6. Click **Fetch Metadata**.
7. Select the data you want to include.

To select a table or view, check the box in the **Include** column for the table or view you want to include. To select all the tables listed in the page, select the **Include** check-box in the header row. When you select a table, all its columns are included in the model.

To select specific columns to include in the model, hover over the table containing the columns you want and click the **Select columns** button that appears.

You can search for tables based on their name or type using the **Filter** field above the list of tables. For example, if you want to view only those tables that have names containing the word "new", enter `new` in the filter field. Similarly, you can filter out tables based on their type. However, you can use only one of the filter criteria, either name or type.

8. Click **Save**.

The new model is created and listed on the **Physical Models** tab of the **Modeling** page.


Editing a Physical Model

1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Click the **Physical Model** tab.
3. Select the check box next to the model you want to modify.

Note: You can search for a specific model using the **Filter** field on the top of the page. It searches the models by the model name.

4. Click the Edit Model button .

The **Edit Physical Model** page is displayed for the selected model with the **Include** check-box displaying the already selected tables and columns.

Note: Physical models are displayed in two views: list and tabular. The list view displays the selected table as a list below the connection specification fields. The tabular view shows only the selected tables and their details. To use the tabular view, click the Tabular View button  on the top of the page. For more information, see [Tabular View of Physical Model](#) on page 75.

5. Modify the connection for the model, if required, using the **Source connection name** drop down list.
6. Remove the added table, or add a table, as required, by clicking the respective rows.
7. Click **Save**.

The selected model gets updated and listed on the **Models** tab of the **Modeling** page.

Tabular View of Physical Model

The tabular view of a physical model displays details of all the selected data tables. Each table displays the attributes and its data type. Other properties of the tables are displayed in the **Properties** panel.

1. In a web browser, go to:


`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
3. Select the check box next to the model you want to view.

Note: You can search for a specific model using the **Filter** field on the top of the page. It searches the models by the model name.

4. Click the View Model button .

Note: You can also go to the tabular view by clicking the Tabular View button  on the **Edit Physical Model** page.

5. To view the properties of a table, select it, and click the **Properties** collapsible tab on the right side of the page.

Note: You cannot modify these properties.

The **Properties** panel displays these details:

Note: The details displayed here are based on the type of the selected data source. For more information, see [Connection-specific Details](#) on page 76.

- **Columns:** Column details, such as the name of the column, the contained data type, its size and whether it is nullable and updatable.
- **Primary Keys:** Name of the primary key (if defined) and related column and key sequence
- **Foreign Keys:** Name of the foreign key (if defined) and related column and table name

Connection-specific Details

Table properties, such as columns, primary keys, and foreign keys displayed in the **Properties** collapsible tab on the **Edit Physical Model** page are based on the type of source connection selected in the physical model. The detail for various connection types are given below.

Note: For the steps to view the **Properties** of tables, see [Tabular View of Physical Model](#) on page 75.

JDBC connection	Columns	Displays the name, data type, and size of each column. The section also displays whether a column is nullable and/or a primary key.
	Primary Keys	Displays the primary key, primary key name, and key sequence.
	Foreign Keys	Displays the foreign key column name, foreign key name, primary key catalog name, primary key table name, primary key column name, and primary key name.
	Indexes	Displays the index name, column name, and whether the indexes are unique or non-unique. If an index is non-unique then the NU check box is displayed as checked otherwise it is displayed as unchecked.

Amazon DynamoDB connection	Columns	Displays the name and data type of each column. The section also displays whether a column can be updated, searched, and/or is a primary key.
	Primary Keys	Displays the primary key and primary key name.
Amazon SimpleDB connection	Columns	Displays the column name and data type for the table you select.
Apache Cassandra connection	Columns	Displays the column name and data type. The section also displays whether a column can be updated, searched, and/or is a primary key.
	Primary Keys	Displays the primary key and primary key name.
Flat File connection	Columns	Displays the column name and data type for the table you select.
Marketo connection	Columns	Displays the column name and data type. The section also displays whether a column is a primary key, readable or updatable.
	Primary Key	Displays the primary key and primary key name.
	Foreign Keys	Displays the foreign key column name, foreign key name, and the reference table name.
	Permissions	Indicates which functions can be performed on the selected physical model table. The functions Read, Insert, Update, Upsert and Delete can be performed on Marketo tables.
	Filter Types	Indicates the column on which search is applicable. For example, <i>id</i> as Filter Type in Lead indicates Lead can be searched with <i>id</i> .
	Note: Marketo does not support multiple Filter Types in Select query.	

For example:

Valid Query:

```
Select * from Lead where id = 1234
```

```
Select * from Lead where email =  
'abc@txyz.com'
```

Invalid Query:

```
Select * from Lead where id = 1234 and  
email = 'abc@txyz.com'
```

MS Dynamics 365 Online connection	Columns	Displays the column name and data type. The section also displays whether a column is nullable and/or a primary key.
	Primary Keys	Displays the primary key and primary key name.
	Foreign Keys	Displays the foreign key column name, foreign key name, and the reference table name.
	Permissions	Indicates which functions can be performed on the selected physical model table. Currently, the functions create, read, write, and delete can be performed on an MS Dynamics 365 Online table.
NetSuite connection	Columns	Displays the name, data type, and size of each column. The section also displays whether a column is nullable and/or a primary key, and which of the functions create, read, and update can be performed on the column.
	Primary Keys	Displays the primary key and primary key name.
Salesforce connection	Columns	Displays the name, data type, and size of each column. The section also displays whether a column is nullable and/or a primary key.
	Primary Keys	Displays the primary key and primary key name.
	Foreign Keys	Displays the foreign key column name, foreign key name, and the reference table name.
SAP NetWeaver connection	Columns	Displays the name, data type, and size of each column. The section also displays whether a column is nullable and/or a primary key.
	Primary Keys	Displays the primary key and primary key name.
	Foreign Keys	Displays the foreign key column name, foreign key name, and the reference table name.
	Permissions	Indicates which functions can be performed on the selected physical model table. Currently, the functions create, write, and delete can be performed on an SAP NetWeaver table.
Splunk Connection	Columns	Displays the name and data type of each column. The section also displays whether a column can be updated, searched, and/or a primary key.
	Primary Keys	Displays the primary key and primary key name.
SugarCRM connection	Columns	Displays the name, type and size of each column. It also shows whether a column is nullable and updatable.
	Primary Keys	Displays the primary key names, column and key sequence.
	Foreign Keys	Displays the foreign key column name, foreign key name, and the reference table name.

Siebel connection	Columns	Displays the name, type and size of each column. The section also displays whether a column is nullable and updatable.
	Primary Keys	Displays the primary key names, column and key sequence.
	Foreign Keys	Displays the foreign key column name, foreign key name, and the reference table name.

Read and Write Limitations

Read limitations

- Spectrum Discovery does not interpret case-sensitive field names as different names. If two tables or column names use the same name but different cases, they are interpreted as duplicate names. If a model store contains such field names, the deployment of that model store fails.

For example, fields like `DeDup`, `Dedup`, and `DEdup` are interpreted as duplicate names. The deployment of a model store containing such case-differentiated fields fails.

- If a Data Source, physical model, logical model or model store has a dependency on a JDBC-specific data source, it cannot be imported or exported using the Administration Utility.

Write limitations

While writing to a database source using the **Write to DB** stage in **Enterprise Designer**, there are certain limitations.

- Creation of tables while writing to a database source using a Spectrum Discovery connection is not supported. If you click **Create Table** while using a Spectrum Discovery connection, a runtime exception is displayed.
- Spectrum Discovery does not support unified writes to multiple connections simultaneously. You can write to a single source at a time.
- Dropping and recreating any table while writing to a database using a Spectrum Discovery connection is not supported.
- Delete and truncate operations are not supported while writing to a database using Spectrum Discovery. Only insert and update operations are supported.
- Spectrum Discovery performs a commit after completion of each batch insert. The **Batch count to commit** cannot be changed to any other value other than 1.

In the **Runtime** tab of the **Write to DB** stage in **Enterprise Designer**, the **Batch count to commit** field can only have the value 1 for Spectrum Discovery.

Note: A batch consists of 1000 records by default. You can configure the batch size according to your system, in the **Runtime** tab of the **Write to DB** stage in **Enterprise Designer**.

- For batch inserts, the feature **Commit at the end** is not supported. This feature refers to performing a commit after all batches are complete. In Spectrum Discovery, a commit is performed at the end of each batch insert completion.


Deleting a Physical Model

Follow this procedure to delete a physical model from Spectrum Discovery.

Note: If you delete a physical model that is mapped to a logical model, the logical model becomes unusable.

1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Select the check box next to the model you want to delete.

Note: You can search for a specific model using the **Filter** field on the top of the page. It searches the models by the model name.

3. Click the Delete Model button .
4. Click **Yes** to confirm.

Context Graph Model

A *Context Graph model* represents entities and relationships. It is used to identify connections among people, groups, and organizations as well as patterns of connectedness, distance, centrality, influence status and importance. In Spectrum Discovery, you can create an entirely new Context Graph model, or you can use a Logical Model as a template for a Context Graph model. You can also edit or delete any existing Context Graph model.

Creating a new Context Graph Model in Spectrum Discovery

A Context Graph model is created by adding business entities to it, defining relationships among the entities, and defining properties of the entities and relationships.

Context Graph must be installed on the Spectrum server for you to perform this procedure.

1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Click the **Context Graph Model** tab.
3. Click the Add model button **+**.
This opens **Create Graph Model** and displays an empty canvas on the **Schema** tab.
4. Next to **Name**, enter a name for the Context Graph model.
The name must begin with a letter and only include alphanumeric characters and the underscore.
5. Perform activities described in [Editing a Context Graph Model in Spectrum Discovery](#) on page 82 to assemble the model.
 - [Add an entity](#) on page 83
 - [Edit entity properties](#) on page 83
 - [Add a relationship](#) on page 84
 - [Add relationship properties](#) on page 85
 - [Index entity properties](#) on page 86

Note: To add virtual metadata to a Context Graph model, see [Using Virtual Metadata in Context Graph Models](#) on page 87.


6. When you finish editing the model, click **Save**.


This saves the Context Graph model and opens the **Context Graph Model** tab of the **Modeling** page. The details displayed for the new Context Graph model are:

Name	The name of the Context Graph model.
Logical Model	This is empty for an entirely new model. When a logical model is used to create a Context Graph model it shows the name of the logical model.
Last Modified	The date and time that the model was last modified on the server.

Creating a Context Graph Model from a Logical Model in Spectrum Discovery

This procedure describes how to use a logical model as a template to create a Context Graph model in Spectrum Discovery.

1. On the **Discovery** menu bar, click **Modeling**.
The **Logical Model** tab is displayed by default.
2. Select the check box for the logical model you want to use, and click the Create Context Graph model button  on the toolbar.

Alternatively, in the **Actions** column for a logical model, you can click the Create Context Graph model button .

This switches to the **Create Graph Model** tab and displays the logical model entities and relationships in a new Context Graph model. The Context Graph model initially has the same name as the logical model.

3. Next to **Name**, you can edit the name of the Context Graph model.
The name must begin with a letter and only include alphanumeric characters and the underscore.
4. On the canvas, you can edit entities and relationships in the model.
For more information, see [Editing a Context Graph Model in Spectrum Discovery](#) on page 82.
5. Click **Save**.

This saves the Context Graph model and opens the **Context Graph Model** tab of the **Modeling** page. The details displayed for the new Context Graph model are:


Name	The name of the Context Graph model.
Logical Model	This shows the name of the logical model that was used to create the Context Graph model.
Last Modified	The date and time that the model was last modified on the server.

Editing a Context Graph Model in Spectrum Discovery

You can modify entity types and relationship labels. You can also add, modify, and delete properties for entity types and relationship labels.

You cannot modify or remove metadata elements if data loaded into a model includes the elements that you are trying to modify or remove. An error message is displayed if you try to remove entities or relationships that have data. Options to delete properties or change the data type are unavailable for entity or relationship properties with data.

This procedure describes how to edit a Context Graph model from within Spectrum Discovery.

1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Click the **Context Graph Model** tab.
3. Select the check box next to the Context Graph model that you want to edit.
Alternatively, you can click the name of the model and skip to step **5** on page 83.
4. Click the Edit model button  on the toolbar.
5. Perform any of the following activities.
 - **Add an entity** on page 83
 - **Edit entity properties** on page 83
 - **Add a relationship** on page 84
 - **Rename a relationship** on page 84
 - **Add relationship properties** on page 85
 - **Delete an entity or relationship** on page 85
 - **Index entity properties** on page 86

To add virtual metadata to a Context Graph model, see **Using Virtual Metadata in Context Graph Models** on page 87.

6. When you finish editing the model, click **Save**.

This updates the model and navigates to the **Context Graph Model** tab.

Add an entity

This procedure describes how to add a standard entity to a Context Graph model.

1. To add an entity, drag an icon from the **Entities** panel to the canvas.
2. In the **Properties** panel, type the name for the entity in the **Entity Name** box.
3. Leave **Virtual Entity** set to false.
To add a virtual entity, see **Create Virtual Entities** on page 87.
4. For each entity property, enter the property name in the **Name** box, select the data type in the **Type** box, and click **Add**.
For more information, see **Edit entity properties** on page 83.

Edit entity properties

Note: You cannot edit properties that are currently populated with data.

1. Select the entity.
2. If necessary, click the **PROPERTIES** tab to expand the **Properties** panel.
3. In the **Properties** panel, enter the following details for the selected entity.

Field	Description
Name	Name of the property. For example, Age.
Type	Data type for the property. For example, integer. This field displays valid data types for a Context Graph entity.

Note: Properties with the same name in different entities always share the same data type. Changing the data type for a property will change the data type for that property with the same name in every other entity. If you want to create a property with a different data type, you must assign it a different name.

4. Click **Add**.

The property and its data type are displayed in the table below the entry fields.

Note: You can delete a property by clicking the **Delete** button  next to the property.

Add a relationship

This procedure describes how to add a relationship between a pair of standard entities.

When either one of the entities is a virtual entity, follow the instructions in [Connect relationships to virtual entities](#) on page 88.

1. To add a relationship between entities or from one entity to itself, do the following:

Note: An entity can have multiple relationships to other entities as well as to itself.

To add a relationship to	Do this
Add a relationship between entities	Drag from one entity to another entity.
Add a relationship to itself	Drag from an entity to itself.

This adds a relationship link pointing from the primary entity to the related entity or from and to the same entity when the relationship is to itself.

2. In the **Properties** panel, type the name for the relationship in the **Relation Name** box.
3. For each relationship property, enter the property name in the **Name** box, select the data type in the **Type** box, and click **Add**.

Rename a relationship

This procedure describes how to rename a relationship in a Context Graph model.

Note: You cannot rename a relationship that is currently populated with data.

To rename a relationship, enter a new **Relation Name** in the **Properties** panel.

Note: All relationships with the same name share the same properties.

New name is	Then the relationship properties
Unique	Remain the same as with the previous name but are now independent of all other relationships.
Shared by other entities	Are replaced by the properties associated with relationships that share its new name.

Add relationship properties

1. Click a relationship.
This selects all relationships with the same name. Adding or removing a property on one relationship adds or removes the same property for all relationships with the same name.
2. In the **Properties** panel, under **Add Properties**, enter the following details for the new property.

Field	Description
Name	Name of the property. For example, Date .
Type	Data type for the property. For example, DateTime . The list displays valid property types for a Context Graph relationship.

Note: Properties with the same name in different relationships always share the same data type. Changing the data type for a property will change the data type for that property with the same name in every other relationship. If you want to create a property with a different data type, you must assign it a different name.


3. Click **Add**.

The property and its data type are added to the table.

Note: You can delete a property by clicking the **Delete** button  next to the property.

Delete an entity or relationship

Note: You cannot delete an entity or relationship from a model that is currently populated with data.

- To delete an entity or relationship from the canvas, select it and click the Delete button  on the toolbar or press the Delete key.

Any relationship links to an entity are also deleted.

Index entity properties


- To index an entity property, click the **Configure Indexes** tab and select the **Index Type** for any property in the table.

Note: Properties with the same name in different entities always share the same index type. When you change this setting for a property in one entity, the change will apply for the same property in every other entity.

Index Type	Description
None	Indexing a property decreases search time for data in the corresponding field. However, you normally do not want to index a field that is rarely queried, because indexing many different fields slows performance when writing to a model.
Exact	The search results must match the case of all characters within a search string.
Case Insensitive	The search results do not have to match the case of characters within a search string. This typically results in more matches for a search.


Copying a Context Graph Model in Spectrum Discovery

This procedure describes how to copy a Context Graph model within Spectrum Discovery.

- On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
- Click the **Context Graph Model** tab.
- Select the check box next to the Context Graph model that you want to copy.
- Click the Copy model button  on the toolbar.
- You can change the name of the model in the **Name** field.
The default name for a copied model is "Copy_of_*modelName*".
- Click **Save**.

Deleting a Context Graph Model

Performing this task permanently deletes selected Context Graph models.

1. On the **Discovery** menu bar, click **Model**.
The **Logical Model** tab is displayed by default.
2. Click the **Context Graph Model** tab.
3. Select the check box next to Context Graph models that you want to delete.
4. Click the Delete model button  on the toolbar.
5. Click **Yes** to confirm.

Using Virtual Metadata in Context Graph Models

In addition to creating models using metadata that stored locally, you can also use virtual metadata that comes from a Spectrum Discovery source in your model. Using this feature you can link virtual entities (from federated data from outside of Context Graph) to physical entities (which do exist in Context Graph). You can also create models that consist solely of virtual entities. Federated data can come from an application such as Salesforce or Siebel, a database, or even a flat file, among other sources.

Create Virtual Entities on page 87

This procedure describes how to create virtual entities based on a connection to a data source.

Connect relationships to virtual entities on page 88

This procedure describes how to add a relationship to a virtual entity.

Example: Virtual Metadata in Context Graph Models on page 89

This example depicts how you might combine physical and virtual entities to graph relationships between customers, purchases, and products.

Create Virtual Entities

This procedure describes how to create virtual entities based on a connection to a data source.

1. On the **Discovery** menu bar, click **Model**.
2. Click the **Context Graph Model** tab.
3. In the **Name** column, click the model name to which you want to add virtual metadata.
4. Click the **Schema** tab.
5. Drag an entity icon from the **Entities** panel to the model canvas.

This displays the entity **Properties** panel.

6. In the **Entity Name** box, type a meaningful name for the entity.
7. Click the **Virtual Entity** box to switch it to **TRUE**.


A blue box  on the entity icon that you dragged to the canvas denotes that it is a virtual entity.

8. In the **Connection** box, click the source for the federated (or virtual data).
This is a connection created in Management Console to a Virtual Data Source. For more information, see [Connections](#) on page 12.
9. In the **Schema** box, select the schema containing the table that you want to use.
10. In the **Table** box, select the name of the baseview or metaview **Table** whose properties you want to use for this entity.
This populates the property **Name** and **Data Type** columns of the entity properties table.
11. In the **Primary Key** box, select the property for the primary key.
For each entity, the data in this field will become its label and its `_stp_id` within the model.
For example, if a property named "Customer ID" is selected as the primary key, the label for each entity of this type will be its customer ID.
12. You can uncheck the check box for properties that you do not want to include in the entity.
13. Repeat steps [5](#) on page 87 through [12](#) on page 88 to add additional virtual entities.

After you add virtual entities to a Context Graph, you can add relationships to connect them to other entities. For more information, see [Connect relationships to virtual entities](#) on page 88.

Connect relationships to virtual entities

This procedure describes how to add a relationship to a virtual entity.

1. To add an relationship to or from a virtual entity, click and drag from the source entity to the target entity.
A blue box  denotes a relationship to or from a virtual entity.
This displays the **Properties** panel with options to define the virtual relationship.
2. In the **Relationship Name** box, enter the name for the new relationship.
This labels the relationship between the two entities you are connecting.
For example, if you were building a model of insurance data, the name might be "Visited" (between a patient entity and provider entity) or "Billed" (between a provider entity and an insurance company entity).
3. The **Virtual** box will be checked if either or both of the entities connected by this relationship are virtual. It will be unchecked if both entities are physical.
4. In the **Connection** box, select the connection for the virtual relationship.
5. In the **Schema** box, select the schema containing the table that you want to use.
6. In the **Table** box, select the name of the baseview or metaview join table whose properties you want to use for this relationship.

If you do not have a separate table that links these two entities, as is often the case for one-to-one relationships, you can reuse the table you selected for the virtual entity that is being linked. In this case, you will choose the field that will be used as the primary key for the Source Link ID and Target Link ID created in step 10 on page 88 of the procedure **Create Virtual Entities** on page 87. If you are linking two virtual entities, you can reuse either table as long as the source entity ID and target entity ID are the properties that were chosen as primary keys in the entity metadata.

This populates the property **Name** and **Data Type** columns of the relationship properties table.

7. Select the **Link Source ID** and the **Link Target ID**.

These are properties of the source and target entities that have the values that will be matched in the join table. For example, if you were using insurance data and you had a source entity of medical providers and a target entity of patients treated by those providers, you might select "TIN" for the source and "ClaimNo" for the target.

8. You can uncheck the check box for properties that you do not want to include in the entity.
9. Click **Save** to save changes to the model.

Example: Virtual Metadata in Context Graph Models

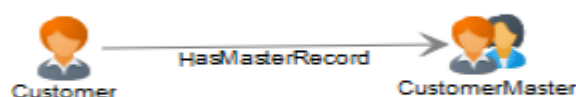
A use case for this virtual metadata might be when you want to access a large amount of customer information at one time, such as profile data, purchasing history, and so on. Storing and replicating all of this information "physically" in a Context Graph model presents potential barriers:

- The volume of some data is too large or too widely distributed to be contained in a central graph database.
- The data changes frequently and would need to be replicated in the graph database too often to be practical.
- There are business or privacy concerns over sensitive information such as Social Security numbers, payroll figures, pricing data, and so on.

By storing some of the information externally and then accessing it virtually, you are still able to obtain a complete view of your customer while mitigating the concerns of duplicating data on the Context Graph server.

The Context Graph Model

You have an existing model containing two types of entities, **Customer** and **CustomerMaster**, with a relationship of **HasMasterRecord**.



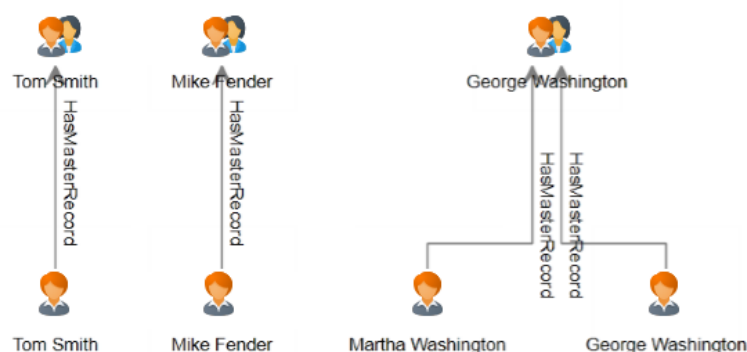
Customer entities contain the following information:

- Customer ID (unique for each customer)
- Name
- Date of birth
- Gold Club status (indicates whether the customer is a member of the Gold Club)

CustomerMaster entities contain the following information:

- Master ID (unique for each household)
- Customer ID
- Name
- Date of birth

The image below shows a model that is built using this metadata. Notice that George and Martha Washington have separate Customer entities, but they share the same CustomerMaster entity, because the master ID represents a household, not an individual.



The External Tables

We want to link customers to their purchase history, which we will create using three external tables—one for customers, one for purchases, and one for products.

We use the same information for the **Customers** table as we did to populate the Customer entities:

CustomerID	Name	DOB	GoldClub
C15	Tom Smith		1
C23	Martha Washington	1967-12-30	1
C76	Mike Fender	1973-07-19	0
C88	George Washington	1967-12-08	0

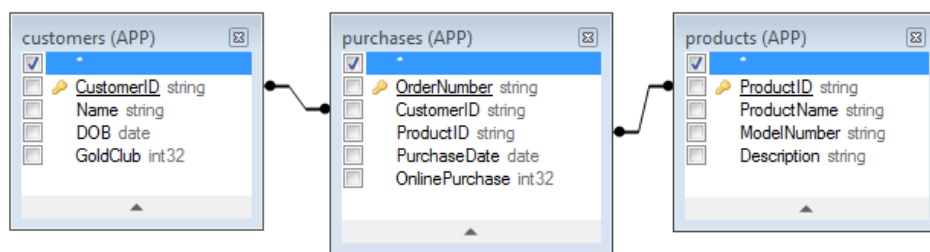
The **Purchases** table contains order numbers, product IDs, and purchase dates, and it indicates whether the purchase was made online. It ties this data to the customer ID, which was also used in the Customer table.

OrderNumber	CustomerID	ProductID	PurchaseDate	OnlinePurchase
45678	C15	1235	2015-10-29	1
45679	C23	1244	2015-10-30	1
45680	C76	1245	2015-10-30	0

The **Products** table contains product names, model numbers, and descriptions. It ties this data to the product ID, which was also used in the Purchases table.

ProductID	ProductName	ModelNumber	Description
1234	Decathlete 150	DCT150	GPS sports watch with 8 hour battery
1235	Decathlete 155	DCT155	GPS sports watch with 8 hour battery and 4g
1244	Decathlete 250	DCT250	GPS sports watch with 16 hour battery
1245	Decathlete 255	DCT255	GPS sports watch with 16 hour battery and 4g

The Purchases table will serve as a join table that establishes connections between records in the Customers and Products tables. The image below illustrates the relationships among the various fields in the three external tables.



Integrating all of this information would produce the following record for Martha Washington:

ProductID	ProductName	ModelNumber	Description	OrderNumber	PurchaseDate	OnlinePurchase	CustomerID	Name	DOB	GoldClub
1244	Decathlete 250	DCT250	GPS sportswatch	45679	2015-10-30	1	C23	Martha Washington	1967-12-30	1

Linking Physical and Virtual Data

To create relationships between entities in the Context Graph model and records in the external tables, the Context Graph entities must have properties that reference the external data. To link from the Context Graph model to records in the Customers table there must be a property in the external table that contains the same values that are stored in the entities' CustomerID field. You can see in the following image that the Martha Washington entity has a property called **CustRef**, which contains the same value as the CustomerID field in the Customers table.



Label: Martha Washington

Type: Customer

Properties:

Property	Value
CustRef	C23
DOB	12/30/1967
GoldMember	1
Name	Martha Washington

To establish the link between the Customer entity and the external table, we must first create a virtual entity for the Customers table. The procedure for doing this is described in [Create Virtual Entities](#) on page 87. The following image shows the completed screen for the new entity type. Notice that "CustomerID" is selected as the Primary Key, which is the property used to link between the virtual and physical entities.

Properties >

Entity Name

ERP_Customer

Virtual Entity ☒

Connection

VirtualDataSource

Schema

sales

Table

sales.customers

Primary Key

CustomerID

Name	Data Type
<input checked="" type="checkbox"/> CustomerID	string
<input checked="" type="checkbox"/> Name	string
<input checked="" type="checkbox"/> DOB	date
<input checked="" type="checkbox"/> GoldClub	integer

After the virtual entity is added, we need to add a relationship linking that entity to the physical entity already in the model. The procedure for doing this is described in [Connect relationships to virtual entities](#) on page 88. The following image shows the completed screen for the new relationship type. Notice that the **Entity Source ID** refers to the CustRef property and the **Entity Target ID** refers to the CustomerID property. This is the link that ties the virtual entity to the physical entity because both properties include the customer ID.

Properties >

Relation Name
CustomerErpRef

Connection
VirtualDataSource

Schema
sales

Table
sales.customers

Entity Source ID
CustRef

Entity Target Id
CustomerID

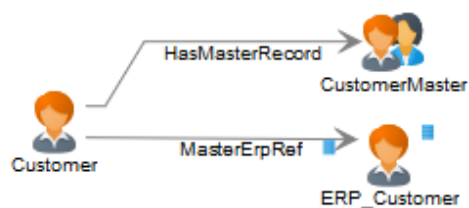
Link Source Id
CustRef

Link Target Id
CustomerID

	Name	Data Type
<input checked="" type="checkbox"/>	CustomerID	string
<input checked="" type="checkbox"/>	Name	string
<input checked="" type="checkbox"/>	DOB	date
<input checked="" type="checkbox"/>	GoldClub	integer

In this example, the relationship between customers is a one-to-one relationship, meaning for every physical entity there should be at most one virtual entity and for each virtual entity there should be at most one physical entity. For one-to-one relationships, as well as one-to-many relationships, there is no need for a separate join table, so we specify Customers as the table and CustomerID for both the **Link Source ID** and the **Link Target ID**.

The metadata for the model now includes a link to the virtual data. The blue box ■ next to the ERP_Customer entity denotes that the entity is virtual. Similarly, the blue box ■ next to the MasterErpRef relationship denotes that it connects to a virtual entity.



Many-to-Many Relationships

Now that we have linked the virtual customer to the physical customer, we need to link purchases and products as well. The optimal approach is to create a virtual entity for products and create a many-to-many virtual relationship between physical Customers and virtual Products using the Purchases table as a join table. We first need to create a virtual entity for the Products table that uses the ProductID as the Primary Key.

Properties

Entity Name

Product

Virtual Entity ☒

Connection

VirtualDataSource

Schema

sales

Table

sales.products

Primary Key

ProductID

	Name	Data Type
<input checked="" type="checkbox"/>	ProductID	string
<input checked="" type="checkbox"/>	ProductName	string
<input checked="" type="checkbox"/>	ModelNumber	string
<input checked="" type="checkbox"/>	Description	string

We next add a new relationship that uses the Purchases table as a join table to achieve a many-to-many relationship. Customer entities are linked with Product entities based on information in the Purchases table. Notice that the CustomerID property is used as the link ID for the Customer entities and the ProductID property is used as the link ID for the Product entities. The Purchases table contains both of these properties, which maps information from Customer to Product.

The **Entity Source ID** (CustRef) is the property of the Customer entity that will be matched to the **Link Source ID** column (CustomerID) in the Purchases table. The **Link Target ID** (ProductID) is the column in the Purchases table that will be matched to the **Entity Target ID** (also ProductID) property of the Products entity. The following diagram illustrates how each field is used to link Customer to Product.

Properties >

Relation Name
Purchased

Connection
VirtualDataSource

Schema
sales

Table
sales.purchases

Entity Source Id
CustRef

Entity Target Id
ProductID

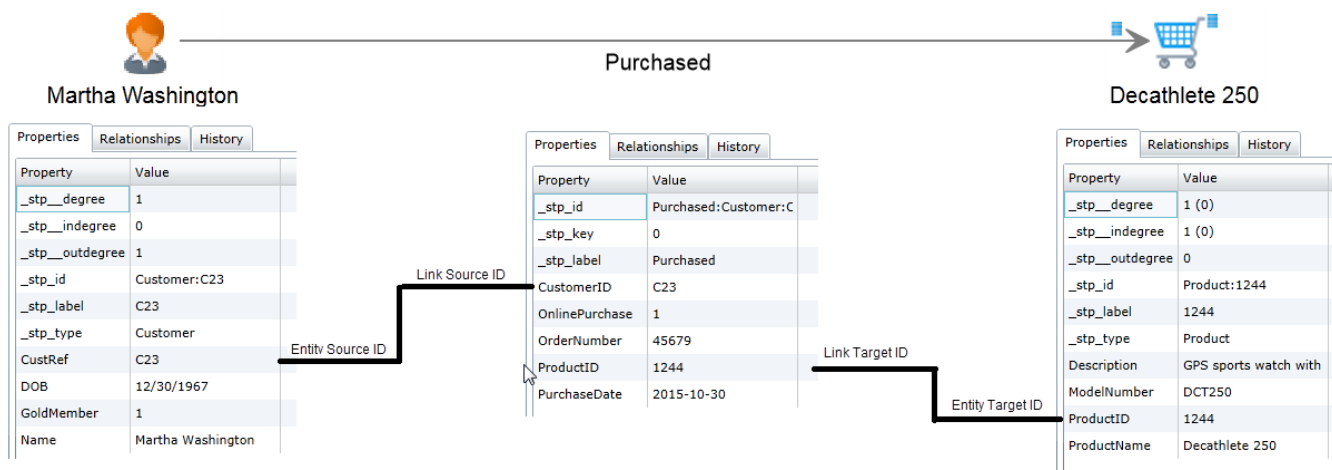
Link Source Id
CustomerID

Link Target Id
ProductID

	Name	Data Type
<input checked="" type="checkbox"/>	OrderNumber	string
<input checked="" type="checkbox"/>	CustomerID	string
<input checked="" type="checkbox"/>	ProductID	string
<input checked="" type="checkbox"/>	PurchaseDate	date

When a query traverses from Customer to Product, it will begin with the property that was selected as the **Entity Source ID**, which in our example is the CustRef property. For Martha Washington the value for this property is C23; this value will be used to find rows in the Purchases table where the **Link Source ID** (in our case, the CustomerID) has a value of C23.

The value in the **Link Target ID** (the ProductID) will be used to look up the corresponding row in the Products table. For each row returned from the Purchases table, the value of the ProductID field will be matched to the **Entity Target ID** property (the ProductID) of the Products table. The following diagram illustrates how each field is used throughout the flow.



Advanced Configuration

One alternative to the approach shown here is to use the Purchases table to create the many-to-many relationship between the virtual customer and the virtual product entities. Doing this involves an additional and unnecessary virtual hop from physical customer to virtual customer. Context Graph is much more efficient at traversing relationships than SQL data sources because it is built upon a graph database; therefore, the extra hop should be avoided if possible.

A second alternative is to create virtual entities for the Purchases table in addition to those for the Products table. This approach would require one-to-many relationships between physical customer entities and virtual purchases entities as well as one-to-many relationships between virtual purchases and virtual products. This is inefficient because it creates unnecessary virtual hops to get from physical customer to virtual product. There is no advantage in representing a purchase as an entity because the relationship can hold all the Purchases fields as properties. The only reason to create an entity for Purchases is if it connects more than two entity types, such as if there were a third foreign key in the table linking purchases to a store.

Mapping Models

Mapping Models - Introduction

The logical models can be mapped to the tables and columns of:

- One or more physical data models.
- Any of the configured and discovered connections.

You have these two options to map your models:

- Mapping through the **Entity Transformation** wizard
- Mapping by **drag and drop** on the canvas: Drag a table from the physical model to the canvas and establish a link between the columns of the physical and logical tables.

Related concepts

[The Entity Transformation Wizard](#) on page 97

Related tasks

[Mapping through drag drop on canvas](#) on page 108

The Entity Transformation Wizard

Use the **Entity Transformation** wizard to map attributes in your selected entity by any or a combination of all these options.

- Mapping Suggestions
- Mapping by browsing the physical models
- Mapping by searching for the required column

The arrows on the top right corner of the wizard navigates you from mapping to test mapping. You can commit your updates in any of the stages in the wizard.



Related concepts

[Mapping Models - Introduction](#) on page 96

Related tasks

[Mapping through drag drop on canvas](#) on page 108



Mapping Suggestions

Mapping suggestions helps you by suggesting the possible tables and columns that match your entity attributes. The suggestions are based on various logic and scoring methods run on the assets you have already discovered.



1. In a web browser, go to:

```
http://server:port/discovery
```

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in using your credentials.
3. Click **Model**.
4. Select the logical model you want to map, and do one of these:
 - Click the **Map Model** icon  in the tool bar.
 - Click the **Map Model** icon  in the **Actions** column.

The **Create Mapping** page is displayed, showing the selected logical model in a graphical form. The page also allows you to give a name to the mapping you are about to define.

5. Enter a name for the mapping in the **Mapping Name** text box.
6. Click the **Entity Transformation** icon  corresponding to the entity you want to map. Alternatively, select the entity to be mapped, and click the **Entity Transformation** icon  in the toolbar.

The **Entity Transformation** wizard is displayed, showing **Suggestions** tab by default.



- **Attribute List:** This column displays the list of attributes in the entity, their data type, and semantic type, if defined by you during model creation.
- **Mapping Suggestions:** Shows the various tables and columns that map with the attributes in the selected entity. Each of the suggestions shows name of the table, name of the connection, and the connection type. For the assets, semantic type and data type are displayed.


7. To filter the attributes based on whether those are **Mapped** or **Not Mapped**, use the **Attribute List** filter.


Note: By default, all the attributes are displayed.

8. To move through the various mapping suggestions, use the right and left arrows on the first and the third **Mapping Suggestions** respectively.

Note: The pop up displays three mapping suggestions at a time, in a decreasing order of matching score. For example, if columns in the *Lead* table in a *Salesforce* connection called as *FrameallSalesforce* matches 95 percent with the attributes of the logical entity you have selected and another table *ContactInformation* in the same connection matches 90 percent, the *Lead* table will precede *ContactInformation* table in mapping suggestions.

9. To refresh mapping suggestions, click the **Reload unmapped suggestions** icon . The suggestions get updated for the unmapped entities. No suggestions are displayed for mapped entities.
10. To map all the suggested columns in any given **Mapping Suggestions**, click the **Map all the suggested columns** button on the top right.
11. To map a specific column from **Mapping Suggestions**, click the mapping icon  corresponding to the column.

The mappings are displayed below the attributes with the mapping icon . The details displayed are: <Connection name>/<Table name>/<Column name>.

Note: To delete the mapping, click  corresponding to it. To cancel the updates made to this pop-up, click the **Cancel** button.

12. Click **OK**.
The mappings are displayed on the **Create Mapping** page.
13. To save details, click **Save** on the **Create Mapping** page.

Mapping by Browsing Models

Use this option to browse through the tables and columns in the physical models and discovered connections to find out the right match for your entity attributes.

You can also convert the values from the tables and columns of the physical model before mapping it. You can perform these conversions:

Transformation Use this to transform the values before populating it to the logical model tables. It includes these types:

- **Column** transformation: Derive the value from a physical model table column. For more information, see [Applying Column Transformation](#) on page 99.
- **Constant** transformation: Define the resulting value as a constant. For more information, see [Applying Constant Transformation](#) on page 100.
- **Function** transformation: Define the resulting value using String, Numeric, Conversion, and Datetime functions. For more information, see [Applying Function Transformation](#) on page 100.

Building Criteria Use this to filter the data retrieved from a physical model column or columns. You can define various filtering criteria.

Joining columns and tables This function combines data from columns of two or more physical model tables before feeding it to the logical model columns.

Applying Column Transformation

A column transformation links the columns of a data source table to define an attribute of a logical entity or one of the parameters of a function transformation. The data source can be a configured connection of any defined physical model.

1. Click the **Browse** tab on the **Entity Transformation** wizard.
2. From the **Attribute List**, select the required attribute.
3. Select the **Column** transformation type option.
The **Data source** dropdown is displayed.
4. Select the required **Physical Model** or **Connection**.

Note: Every data source in the dropdown displays if it is a **Connection** or a **Physical Model**.

The **Table Name** dropdown is displayed. It displays all the tables in the selected data source.

5. Select the required table.
The **Column Name** dropdown is displayed. It displays all the columns in the selected table.
6. Select the column you want to map to the attribute.
The selected column is mapped and these details are displayed below the attribute: *<Physical model name><Table name>/<column name>*.

Note: The data type of the attribute is inherited from the mapped physical model column.

Note: To delete the mapping, click  corresponding to it. To cancel the updates made to this pop-up, click the **Cancel** button.

7. Click **OK**.
The mappings are displayed on the **Create Mapping** page.
8. To save details, click **Save** on the **Create Mapping** page.

Applying Constant Transformation

A constant transformation allows you to apply a constant value to define an attribute of a logical entity or one of the parameters of a function transformation.

1. Click the **Browse** tab on the **Entity Transformation** wizard.
2. From the **Attribute List**, select the required attribute.
3. Select the **Constant** transformation type option.
The **Type** dropdown list is displayed.
4. Select the type of constant to be applied.
5. Specify the constant **Value**.
6. Click **OK**.
The specified constant transformation is applied on the selected attribute and this detail is displayed: *<Constant type>/<Constant value>*.

Note: To delete the mapping, click  corresponding to it. To cancel the updates made to this pop-up, click the **Cancel** button.

7. Click **OK**.
The mappings are displayed on the **Create Mapping** page.
8. To save details, click **Save** on the **Create Mapping** page.

Applying Function Transformation

A function transformation applies functions of string, numeric, conversion, and datetime to define an attribute of a logical entity, or one of the parameters of a parent function transformation.

1. Click the **Browse** tab on the **Entity Transformation** wizard.
2. From the **Attribute List**, select the required attribute.
3. Select the **Function** transformation type option.
4. From the **Category** drop-down, select one of these function transformations, as needed:
 - **String** function transformations, such as concatenation of two string columns and left or right trim.
 - **Numeric** function transformations, such as obtaining absolute value of a number from a column and returning an ASCII or Unicode integer representation of a character.

- **Conversion** function transformation, such as convert or parse data types
- **Datetime** function transformations

To apply a String function transformation: In the **Category** field, select `String`.
Select the **Name** of the required string transformation.
Click the **string** arrow, and use **Column**, **Constant**, or **Function** option to define the parameters of the string function. For example, if you selected `lcase(string string)` as the function **Name**, select the string that needs to be transformed to lower case.

Note: In selecting function **Name**, take care that the **Return type** of the function matches the data type of the column you selected for transformation. The **Return type** is displayed below the **Name** field after you select the function name.

To apply a Numeric function transformation: In the **Category** field, select `Numeric`.
Select the **Name** of the required numeric transformation.
Click the **number** arrow, and use **Column**, **Constant**, or **Function** option to define the parameters of the numeric function. For example, if you selected `abs(integer number)` as the function **Name**, select the integer that needs to be transformed to its absolute value.

Note: In selecting function **Name**, take care that the **Return type** of the function matches the data type of the column you selected for transformation. The **Return type** is displayed below the **Name** field after you select the function name.

To apply a Conversion function transformation: In the **Category** field, select `Conversion`.
Select the **Name** of the required conversion transformation. Options to specify the value to be converted and the targeted output format is displayed below the **Name** field. For example, if you select `parsedouble(string double, string format)` as the function **Name**, **double** and **format** options are displayed to specify the value and format respectively.
Click **double** and use **Column**, **Constant**, or **Function** option to define the selected conversion function parameters.

Note: In selecting function **Name**, take care that the **Return type** of the function matches the data type of the column you selected for transformation. The **Return type** is displayed below the **Name** field after you select the function name.

Select the format to which the source value needs to be converted using one of these **Target** options: **Column**, **Constant**, or **Function**. For more information, see [Supported Datatypes for Conversion Transformations](#)

Note: The format conforms to the standard scheme as specified in Java Decimal formats. See this table for examples of **Type** and **Value** in case you choose **Constant** option for format specification:

Table 1: DecimalFormat.java Output

Value	Pattern	Output	Description
123456.789	###,###.###	123,456.789	The pound sign (#) denotes a digit, the comma is a placeholder for the grouping separator, and the period is a placeholder for the decimal separator.
123456.789	###.##	123456.79	The value has three digits to the right of the decimal point, but the pattern has only two. The format method handles this by rounding up.
123.78	000000.000	000123.780	The pattern specifies leading and trailing zeros, because the 0 character is used instead of the pound sign (#).
12345.67	\$###,###.###	\$12,345.67	The first character in the pattern is the dollar sign (\$). Note that it immediately precedes the leftmost digit in the formatted output.


Note: See more information about decimal formats here:

<https://docs.oracle.com/javase/7/docs/api/java/text/DecimalFormat.html> and details of special pattern characters here

<https://docs.oracle.com/javase/7/docs/api/java/text/DecimalFormat.html>.

- Click **OK**.

The applied function transformation is reflected below the attribute.

Note: To delete the mapping, click  corresponding to it. To cancel the updates made to this pop-up, click the **Cancel** button.

- Click **OK**.

The mappings are displayed on the **Create Mapping** page.

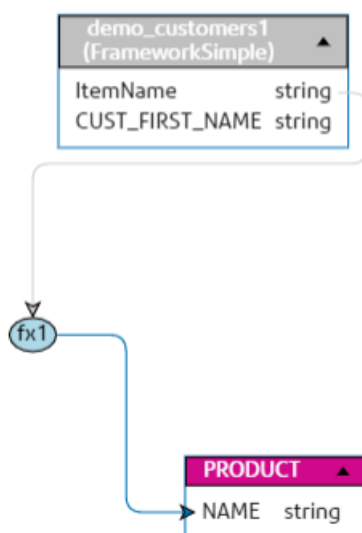
- To save details, click **Save** on the **Create Mapping** page.

Example 1- Applying a String Function Transformation

For example, if you wish to trim spaces from the right of each value of the column `ItemName` of the physical model table `demo_Customers1` and fill in the trimmed value into the linked `Name` column of the logical table `Product`:

1. Select the `Product` table on the canvas.
2. In the **Transformations** panel, click Δ adjacent to the `Name` column, and in the **Transformation of column 'NAME'** pop-up window, select the **Function** option.
3. In the **Category** drop-down list, select `String`.
4. In the **Name** drop-down list, select the `rtrim(string string)` function.
5. The **String** function gets added to the pop-up below the entry fields. Click this function, and define the parameter on which this function needs to be applied by using the **Column** option such that the resulting data type of the parameter is a string.
6. Click **OK**.

The applied function transformation is depicted on the canvas as below:

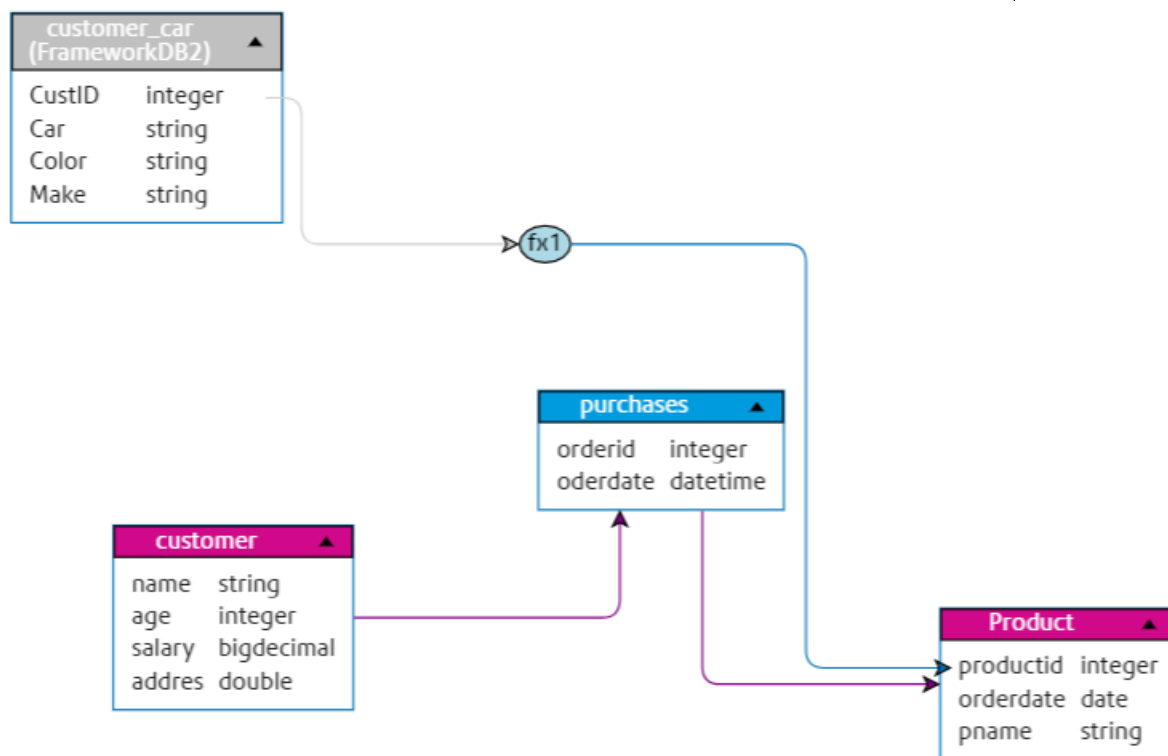


Example 2: Applying Numeric function transformation.

1. Select the `productid` column on `Product` table on the **Create Mapping** page.
2. In the **Transformations** panel, click Δ adjacent to the `productid` column, and in the **Transformation of column productid (integer)** pop-up window, select the **Function** option.
3. In the **Category** drop-down list, select `Numeric`.
4. In the **Name** drop-down list, select `abs(integer number)`. Option to specify the required number is displayed below the **Name** field.

5. Click the **number** arrow, and use the column option to select `CustID` (Integer) column of `customer_car` table in the `FrameworkDB2` physical model.
6. Click **OK**.

This mapping is displayed on the canvas.



Searching for specific asset to map



Use this option when you want to search for a specific table or column across the physical models and discovered connections.


1. Click the **Search** tab on the **Entity Transformation** wizard.
2. From the **Attribute List**, click on the attribute you want to map.
3. In the **Search for an asset** text box, enter the name of the asset you want to map the attribute to, and press **Enter**.

Note: You can also search for an asset with its tag.

A list of tables with the specified asset is displayed.

4. Click the required table.
A list of columns in the selected table is displayed.

- Click the mapping icon  corresponding to the required column. It gets mapped to the selected attribute. The details (table name/column name) are displayed below the attributes with the mapping icon .

Note: To delete the mapping, click  corresponding to it. To cancel the updates made to this pop-up, click the **Cancel** button.

- Click **OK**.
The mappings are displayed on the **Create Mapping** page.
- To save details, click **Save** on the **Create Mapping** page.




Using Join Suggestions

Use the **Joins builder** to combine data from columns of multiple discovered tables before it is populated to the logical model columns.

In **Spectrum Discovery**, if you are using the **Discovery** module, the **Joins** page gives you suggestions (**Join Suggestion** tab). These suggestions are based on the primary and foreign key linkages in the tables you already discovered.

When you accept the suggestion, it displays in the **Join Definition** tab.




To use Join Suggestions, perform these steps:


- On the **Create Mapping <logical model name>** page, select the required logical entity, and click the **Join builder** icon on the tool bar . Alternatively, click the **Entity Transformation** icon  of the required logical entity and use the forward arrow  on the top right of the page to go to the **Join Definition for logical table <table name>** page.
- Click the **Join Suggestion** tab.
The suggestions for joins among the discovered tables mapped to the selected logical model is displayed in the **Left table**, **Right table**, and **Join type** fields.
- Select the required suggestion and click **Apply Join**.
The join is applied and it becomes visible in the **Join Definition** tab.
- To create another join manually, click the **Add table** button and see the steps of manual join creation in [Creating Join Definitions manually](#) on page 105.


Creating Join Definitions manually

Use the **Joins builder** to combine data from columns of two physical model tables before it is populated to the logical model columns.

To create a join definition, perform these steps:

- On the **Create Mapping <logical model name>** page, select the required logical entity, and click the **Join builder** icon on the tool bar . Alternatively, click the **Entity Transformation** icon  of the required logical entity and use the forward arrow  on the top right of the page to go to the **Join Definition for logical table <table name>** page.

2. On the **Join Definition** tab, click the **Add table** button.
A table is displayed with **Left table**, **Right table**, and **Join type** columns.
3. Use the **<select table>** and **<select column>** drop-down options of the **Left table** and **Right table** to specify the columns from which data is to be joined.
4. From the **Join type** drop-down list, select the type of join you want between the columns. The options are:
 - **Inner join**: Returns matching records between the selected left and the right table columns.
 - **Left join**: Returns all records from the left column, in addition to any matching records from the right column.
 - **Right join**: Returns all the records from the right column, in addition to any matching record from the left column.
 - **Full outer join**: Returns all the records from the left and the right columns.
5. To establish joins on multiple columns of the table, hover cursor over the table row, and click the **Add comparison** icon  that appears.
A new **<select column>** row is displayed.
6. Repeat steps 4 and 5 to define the join criteria.
7. To establish join between another set of tables, click the **Add table** button.
A new set of **<Select table>** and **<Select column>** rows are displayed.
8. Repeat steps 3 and 4 to define the join.
9. Click **OK**.
The join between the tables is established and data is populated in the logical model column from the selected columns on the basis of the defined join.

Note: Joins are not represented by any physical link between the tables on the canvas.
10. To delete any join, hover the required row, and click the **Delete** icon  that appears.

Criteria Builder

Components of a Criterion

The criteria builder of a logical entity consists of a parent group, which comprises of a hierarchy of child criteria and child groups.

Group A group is defined using multiple criteria and child groups. You can create a hierarchy of child groups and criteria up to any desired level within the parent group.

For each group, you need to specify whether the data fetched from the source physical model tables needs to satisfy all, at least one, or none of the defined criteria of that group.




Criterion A criterion is defined using either one or two operands and a logical operator.

The selected logical operator either compares the left operand to the values defined in the right operand, or places a condition on the left operand.


Left Operand	The column of the physical model table on which you want to define the criterion.
Right Operand	The right operand is defined as any one of these: <ul style="list-style-type: none"> • A constant • A set of constants • A column of any of the physical model tables • A set of columns of any of the physical model tables
Operator	The logical operator defines the condition to be fulfilled for the criterion. The operator can: <ul style="list-style-type: none"> • Place a condition on the left operand. • Place a condition comparing the left operand with the right operand.

Using Criteria Builder

You need to have a transformation defined to build a criteria.

1. On the **Create Mapping <logical model name>** page, select the required logical entity, and click the **Criteria Builder** icon on the tool bar . Alternatively, click the **Entity Transformation** icon  of the required logical entity.
2. Select a grouping method for the criteria from the **Group satisfying criterion** drop-down list. Specify whether it is necessary to fulfill **Any**, **All**, or **None** of the criteria defined in the criteria group.
3. To add a criterion to a group, hover the mouse on the parent group, and click the **Add Comparison** icon . The added criterion shows below the group.
4. To build the criteria:
 - a) Click **<Column 1>** and select the column to which the criteria is to be applied.
 - b) Click the **<Comparison>** drop-down list to select the required logical operator.


Note: You can add multiple criteria to one criteria group. Use the **Add Comparison** icon to add another criteria to this group, and repeat steps a and b to build the criteria.

5. To add a child to this criteria group, click the **Add group** icon , and repeat steps 3 and 4 to build criteria.
6. Click **OK**.
The defined criteria are saved and is used to retrieve data from the mapped tables and columns to the logical model.



Preview

You can preview your mapping results before you save those. The **Preview** window fetches values from the columns mapped with the attributes and displays it to you. You can preview the mappings in two ways:

- Directly from the canvas
- By using the **Entity Transformation** wizard

To preview mapping results directly from the canvas, click the **Preview** icon .

To preview mapping results using the **Entity Transformation** wizard, perform these steps:



1. Use the forward arrow  in the **Entity Transformation** wizard to reach the **Preview** window.
2. Click the **Preview** button.
A preview of the values fetched from the columns mapped to each attribute displays.
3. Click **OK** to commit your updates in any of the wizard windows.
4. Click **Cancel** to cancel your updates.
5. Use the left arrow  to move back in the wizard.

Mapping through drag drop on canvas

1. In a web browser, go to:

`http://server:port/discovery`


Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in with your credentials.
3. Click **Modeling**.
4. Select the logical model you want to map and click the **Map Model** icon  in the tool bar or the mapping icon  in the **Actions** column.
The **Create Mapping** page is displayed, showing the selected logical model in graphical form. The page also allows you to give a name to the mapping you are about to define.
5. Enter a name for the mapping in the **Mapping Name** text box.
6. In the **Physical Model** panel, select the physical model tables to which you want to map the entities of your logical model. Use one of these mapping methods, as applicable:
 - **To map using drag-and-drop:**
 - a. From the list of models, click the required physical model. A list of all the tables contained in the model is displayed.
 - b. Drag the table you want to the canvas. The table displays the columns and the data type of each column.





- c. Map a column to an attribute of a logical entity by dragging the mouse from the column to the attribute.
- **To map using the search box** (useful when you know only the table name and not the model to which it belongs):
 - a. Enter the search term in the search field and from the adjacent drop-down list, select **Columns**, **Tables**, or **All**, based on whether the term needs to be only in columns or tables, or in columns, tables as well as the models.

The results are displayed below the search fields, showing the name of the column, data type, name of the table or view, and name of the physical model to which the table belongs.

- b. If you searched for a table, click it, and click the **Add to Canvas** button. The selected table gets added to the canvas and you can map it using the drag-and-drop method described above.
- c. If you searched for a column, click the column name. The **Entity** and **Attribute** fields appear. The **Entity** field lists all the entities in the currently mapped logical model, while the **Attribute** field lists all the attributes of the selected entity.

Select the entity and attribute you want to map, then click the Mapping icon .

Note: One attribute can be mapped to only one column at a time.

- **To convert a value in the physical model column before mapping** it to a logical entity, use transformation. For more information, see [Applying a Transformation](#).
 - **To filter data from a logical model**, click the Criteria Builder icon . For more information, see [Criteria](#) on page 117.
 - **To combine data from columns of two physical model tables** before populating the logical model columns, see [Joins](#) on page 119.
7. To create a new physical model, click the **Create Physical Model** icon  in the **Physical Model** panel, and specify details in the **Create Physical Model** pop-up window that is displayed. This model gets displayed in the **Physical Model** panel of the **Create Mapping** page. You can map it to the selected logical model.
 8. To delete any of the mappings or tables on the canvas, select it and click the **Delete mapping** icon  in the tool bar.
 9. Click **Save**.
The mapping is saved and you are navigated back to the **Logical Model** tab of the **Modeling** page.
 10. To view the mapping, click  corresponding to the logical model.
Mapping details are displayed under these headers:
 - Mapping Name: Name of the defined mapping
 - Mapped to: Name of the mapped physical model
 - Last modified: Date (*mm:dd:yyyy*) and time (*hh:mm:ss*) of last modification made to the mappings

- Actions: **Profile Model**  and **Delete**  icons to profile and delete the mapping respectively.

Related concepts

Mapping Models - Introduction on page 96

The Entity Transformation Wizard on page 97

Transformations

This feature gives you the flexibility of transforming a value before populating it to a column in a logical model entity. The values are sourced from the columns of physical model tables.

On the basis of the functions you need to apply on values before populating the logical model columns, you can do transformation on:

- **Columns:** To populate values from a column of a physical model table
- **Constant:** To apply a constant value to a logical model column or to one of the parameters of a function transformation
- **Function:** To apply functions of string, numeric, conversion, and datetime to a logical model column, or to one of the parameters of a function transformation.


You can create complex nested transformations within the parent transformation.


Multiple Mapping and Union

You can map a logical model multiple times, making it possible for you to pull data for an entity in your logical model from multiple physical models. For example, `ID` and `firstName` of `Customer` entity in your logical model can be mapped to `Customers_PostgresPhysicalModel` in `Mapping1` and `Customers_FrameworkSplunkPhysicalModel` in `Mapping2`.

You can later perform a union operation on `Mapping1` and `Mapping2` to combine the `ID` and `firstName` data in both the mappings, and use this combined data for other functions, such as profiling.

To map a logical model, perform these steps:


1. On the **Logical Model** tab of the **Modeling** page, select the logical model to be mapped, and follow the steps described in **Mapping through drag drop on canvas** on page 108.
2. To view the multiple mappings for your logical model, click . The mappings are displayed under the **Mapping Name** column. The mapped physical models and last modified date and time of the respective mappings are also displayed.

Note: The first mapping created on a logical model is its default mapping and is flagged out  under the **Actions** column. This is the mapping that becomes available in the model store by default when you select this logical model during model store creation.

3. To change the default mapping of a logical model, flag it out  in the **Actions** column.

To perform union of mappings, follow these steps:

1. Select the required logical model, and click the **Create Union** icon  in the tool bar.

Note: The **Create Union** icon  is enabled only for those logical models that have more than one mappings.

The **Create Union** pop-up window appears.


2. In the **Name** field, enter a name for this union.
3. From the **Type** option button, select if you want to perform **Union** or **Union All** operation.

Note: Union removes the duplicate rows while Union All operation retains those.



4. From the list below **Select mapping**, select the model maps on which you want to perform union, and click **Create Union**.

The union is saved and you are navigated back to the **Modeling** page.

5. To view the union, click  corresponding to the logical model.

The union is listed along with the mappings in the **Mapping Name** column. The union icon  distinguishes it from the mappings of the logical model.

Tasks that you can perform on the mappings and unions created on your logical model:



- Profile any of your mappings or union: To do this, click the corresponding **Profile**  icon. For details on profiling, see [Creating a Profile](#) on page 211.
- Delete any of your mappings or union: To do this, click the corresponding **Delete**  icon.

Note: If the mapping you want to delete is used in any union, those unions will need to be deleted before the mapping.

Copy mapping

You can create a new mapping between a logical and a physical model using an existing map as a base.

To copy a mapping:

1. On the **Logical Model** tab of the **Modeling** page, click  corresponding to the logical model to be mapped.
A list of all the mappings for this model is displayed.
2. Select the **Copy Mapping** icon  corresponding to the mapping you want to copy.
The **Copy Mapping** pop-up window is displayed showing name of the copied map as: **Copy_of_<name of the parent map>**.
3. Modify the name of the map, as needed and **Save**.
The saved map displays in the list of mappings for the logical model on the **Logical Model** tab.
4. Click on the map and make needed modifications in the **Edit Mapping- <name of the copied map>** page.


5. **Save** the mapping when done.
The saved map displays in the list of mappings for the model on the **Logical Model** tab.

Applying a Column Transformation

A column transformation allows you to link the required column of a physical model table to define an attribute of a logical entity or one of the parameters of a function transformation. You can do a column transformation by using the **Transformations** panel as described below or by linking a physical model column with an attribute in a logical entity directly on the canvas.

1. On the **Create Mapping <logical model name>** page, select the entity corresponding to the attribute to which you want the transformed value applied, and click the **Transformations** collapsible tab on the right side of the page.
The **Transformations** panel is displayed showing the name of all the attributes in the selected entity. The **Transformation** column remains blank initially. It displays the transformation type after you save the transformation.

Note: The **Transformation** panel doesn't display details of the physical model table. In case you click a physical model table after an entity and open the **Transformation** panel, it displays details of the last selected entity.

2. Select the attribute to which you want the transformed value applied, and click the **Transformation** icon .
The **Transformation of column '<column name>'** pop-up window is displayed.
3. Select the **Column** option, and from the **Physical Model Name** drop-down list, select the required physical model.
4. From the **Table Name** drop-down list, select the required table.
5. From the **Column Name** drop-down list, select the required column that you want to map with the attribute.
6. Click **OK**.
The selected column gets mapped with the attribute and it is indicated by a connecting arrow between the two on the canvas. The **Transformation** panel displays this as a **Simple** transformation.

Note: The data type of the attribute is inherited from the mapped physical model column.


Applying a Constant Transformation

A constant transformation allows you to apply a constant value to define an attribute of a logical entity or one of the parameters of a function transformation.

1. On the **Create Mapping <logical model name>** page, select the entity corresponding to the attribute to which you want the transformed value applied, and click the **Transformations** collapsible tab on the right side of the page.

The **Transformations** panel is displayed showing the name of all the attributes in the selected entity. The **Transformation** column remains blank initially. It displays the transformation type after you save the transformation.

Note: The **Transformation** panel doesn't display details of the physical model table. In case you click a physical model table after an entity and open the **Transformation** panel, it displays details of the last selected entity.

2. Select the attribute to which you want the transformed value applied, and click the **Transformation** icon  .
The **Transformation of column '<column name>'** pop-up window is displayed.
3. Select the **Constant** option, and from the **Type** drop-down list, select the data type of the constant value you want to define.
4. In the **Value** field, enter a constant value for the selected data type.
5. Click **OK**.

The applied constant transformation is reflected on the canvas with a transformation node  and it is indicated as **Constant** transformation in the **Transformation** panel.


Note: The **Transformation of column '<column name>'** pop-up window is displayed when you double-click this node.

Applying a Function Transformation

A function transformation allows you to apply functions of string, numeric, conversion, and datetime to define an attribute of a logical entity, or one of the parameters of a parent function transformation.

1. On the **Create Mapping <logical model name>** page, select the entity corresponding to the attribute to which you want the transformed value applied, and click the **Transformations** collapsible tab on the right side of the page.
The **Transformations** panel is displayed showing the name of all the attributes in the selected entity. The **Transformation** column remains blank initially. It displays the transformation type after you save the transformation.

Note: The **Transformation** panel doesn't display details of the physical model table. In case you click a physical model table after an entity and open the **Transformation** panel, it displays details of the last selected entity.

2. Select the attribute to which you want the transformed value applied, and click the **Transformation** icon  .
The **Transformation of column '<column name>'** pop-up window is displayed.
3. Select the **Function** option, and from the **Category** drop-down list, select the category for the function you want to apply. You can perform one of these function transformations: choose from **String**, **Numeric**, **Conversion**, and **Datetime**.
 - String function transformation, such as concatenation of two string columns and left or right trim.

- Numeric function transformations, such as obtaining absolute value of a number from a column and return an ASCII or Unicode integer representation of a character.
- Conversion function transformation, such as convert or parse data types

To apply a String function transformation: In the **Category** field, select `String`.
Select the **Name** of the required string transformation.
Click the **string** arrow, and use **Column**, **Constant**, or **Function** option to define the parameters of the string function. For example, if you selected `lcase(string string)` as the function **Name**, select the string that needs to be transformed to lower case.

Note: In selecting function **Name**, take care that the **Return type** of the function matches the data type of the column you selected for transformation. The **Return type** is displayed below the **Name** field after you select the function name.

To apply a Numeric function transformation: In the **Category** field, select `Numeric`.
Select the **Name** of the required numeric transformation.
Click the **number** arrow, and use **Column**, **Constant**, or **Function** option to define the parameters of the numeric function. For example, if you selected `abs(integer number)` as the function **Name**, select the integer that needs to be transformed to its absolute value.

Note: In selecting function **Name**, take care that the **Return type** of the function matches the data type of the column you selected for transformation. The **Return type** is displayed below the **Name** field after you select the function name.

To apply a Conversion function transformation: In the **Category** field, select `Conversion`.
Select the **Name** of the required conversion transformation. Options to specify the value to be converted and the targeted output format is displayed below the **Name** field. For example, if you select `parsedouble(string double, string format)` as the function **Name**, **double** and **format** options are displayed to specify the value and format respectively.
Click **double** and use **Column**, **Constant**, or **Function** option to define the selected conversion function parameters.

Note: In selecting function **Name**, take care that the **Return type** of the function matches the data type of the column you selected for transformation. The **Return type** is displayed below the **Name** field after you select the function name.

Select the format to which the source value needs to be converted using one of these **Target** options: **Column**, **Constant**, or **Function**. For more information, see [Supported Datatypes for Conversion Transformations](#)

Note: The format conforms to the standard scheme as specified in Java Decimal formats. See this table for examples of **Type** and **Value** in case you choose **Constant** option for format specification:

Table 2: DecimalFormat.java Output

Value	Pattern	Output	Description
123456.789	###,###.###	123,456.789	The pound sign (#) denotes a digit, the comma is a placeholder for the grouping separator, and the period is a placeholder for the decimal separator.
123456.789	###.##	123456.79	The value has three digits to the right of the decimal point, but the pattern has only two. The format method handles this by rounding up.
123.78	000000.000	000123.780	The pattern specifies leading and trailing zeros, because the 0 character is used instead of the pound sign (#).
12345.67	\$###,###.###	\$12,345.67	The first character in the pattern is the dollar sign (\$). Note that it immediately precedes the leftmost digit in the formatted output.

Note: See more information about decimal formats here:

<https://docs.oracle.com/javase/7/docs/api/java/text/DecimalFormat.html> and details of special pattern characters here

<https://docs.oracle.com/javase/7/docs/api/java/text/DecimalFormat.html>.

4. Click **OK**.
The applied function transformation is reflected on the canvas with a transformation node.
5. Click **Save**.

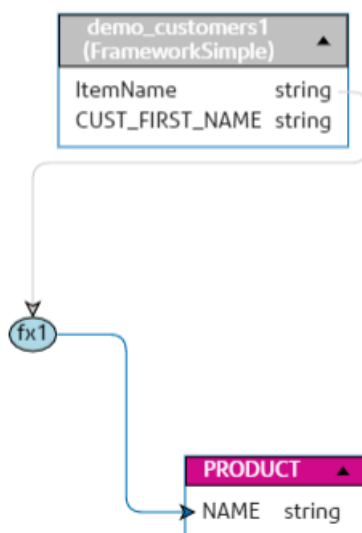
Example 1 - Applying a String Function Transformation

For example, if you wish to trim spaces from the right of each value of the column `ItemName` of the physical model table `demo_Customers1` and fill in the trimmed value into the linked `Name` column of the logical table `Product`:

1. Select the `Product` table on the canvas.

2. In the **Transformations** panel, click ▲ adjacent to the `Name` column, and in the **Transformation of column 'NAME'** pop-up window, select the **Function** option.
3. In the **Category** drop-down list, select `String`.
4. In the **Name** drop-down list, select the `rtrim(string string)` function.
5. The **String** function gets added to the pop-up below the entry fields. Click this function, and define the parameter on which this function needs to be applied by using the **Column** option such that the resulting data type of the parameter is a string.
6. Click **OK**.

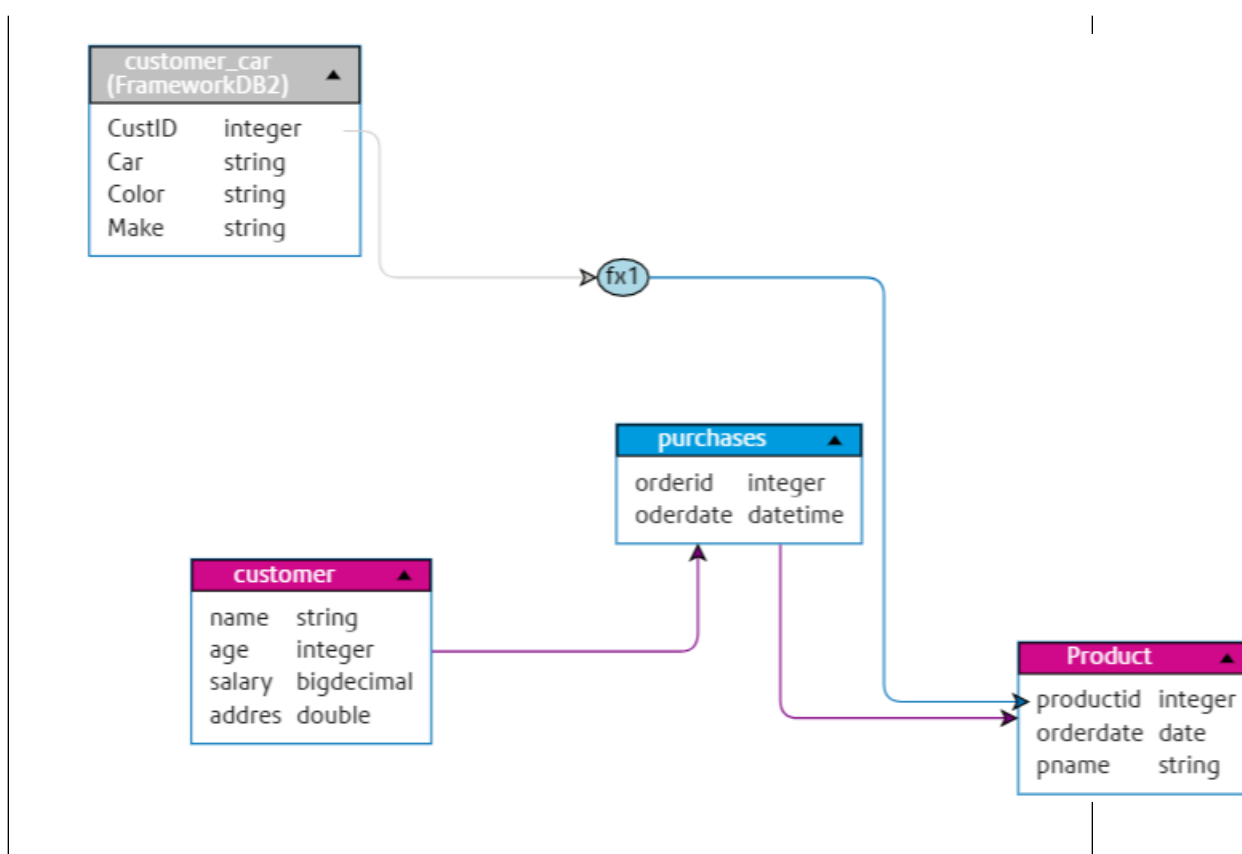
The applied function transformation is depicted on the canvas as below:



Example 2: Applying **Numeric** function transformation.

1. Select the **productid** column on Product table on the **Create Mapping** page.
2. In the **Transformations** panel, click ▲ adjacent to the `productid` column, and in the **Transformation of column productid (integer)** pop-up window, select the **Function** option.
3. In the **Category** drop-down list, select `Numeric`.
4. In the **Name** drop-down list, select `abs(integer number)`. Option to specify the required number is displayed below the **Name** field.
5. Click the **number** arrow, and use the column option to select `CustID (Integer)` column of `customer_car` table in the `FrameworkDB2` physical model.
6. Click **OK**.

This mapping is displayed on the canvas.



Editing a Transformation

For Column, Constant, and Function transformations, the editing can be done by clicking the Transformation icon in the **Transformations** panel for the relevant entity. Additionally, in case of Constant and Function transformations, you can double-click the Transformation node on the canvas and edit the values in the **Transformation of column '<column name>'** pop-up window that gets displayed.

Deleting a Transformation

1. Delete a transformation using either of these steps:
 - Select the transformation function node or the transformation link on the canvas and press **Delete** on your keyboard.
 - In the **Transformations** panel of the logical entity, click ☐ against the transformation you want to delete.
2. Click **Save**.

Criteria

Use Criteria Builder to define a criteria for filtering data retrieved from the physical model columns. The filtered data is fed to the logical model columns.

Note: This is an optional function.


Creating a Criterion



The **Criteria** section is displayed for the selected logical model entity only if at least one transformation is defined for it.

1. On the **Create Mapping <logical model name>** page, select the required logical entity, and click the **Criteria Builder** icon on the tool bar .

Note: The **Transformation** panel doesn't display details of the physical model table. In case you click a physical model table after an entity and open the **Transformation** panel, it displays details of the last selected entity.

The **Criteria Builder for logical table <name of the table>** pop-up window is displayed, with the **Group satisfying criterion** drop-down list.

Note: To add another group, hover the mouse on the group and click the **Add group** icon  that is displayed.

2. Select a grouping method for the criteria by clicking on **Group satisfying criterion** drop-down list. Specify whether it is necessary to fulfill all, at least one, or none of the criteria defined in the criteria group.
3. To add a child group to the parent group, hover the mouse on the parent group and click the **Add group** icon  that is displayed.
4. To add a criterion to a group, hover the mouse on the parent group, and click the **Add Comparison** icon . The added criterion shows below the group.
 - a) To define the left operand of the criterion, in the **Expression Builder**, click **<Column 1>** to select one of the physical table columns.
 - b) To select a logical operator from the provided list, in the **Expression Builder**, click **<comparison>**.
If the selected logical operator requires a right operand, a placeholder for the right operand is displayed to be defined.
 - c) If the right operand is required, click **<Column 2>**.

The right operand can be defined using either a constant or a physical model table column.

Constant Enter the constant, and click **Add constant**.

column(s) Select one of the available physical model columns.

Based on the number of operands the logical operator expects in the right operand, define the right operand as any of these:


- A constant
- A set of constants
- A column of any of the physical model tables

- A set of columns of any of the physical model tables



5. Repeat steps 3 and 4 as many times as is required to create the desired criteria.
6. Click **OK**.

All, any, or none of the defined set of child criteria need to hold true in order to retrieve the required data from the connected physical model tables into the target logical model columns. This is as defined for the parent criterion.

Editing a Criterion

1. Select the required logical entity, and click the **Criteria Builder** icon on the tool bar . The **Criteria Builder for logical table <name of the table>** pop-up window is displayed, with the defined groups and criteria.
2. Select the required group, child group or criterion.
3. Make the required changes.
4. Click **OK**.

Deleting a Criterion

1. Select the required logical entity, and click the **Criteria Builder** icon on the tool bar . The **Criteria Builder for logical table <name of the table>** pop-up window is displayed, with the defined groups and criteria.
2. Select the required group, child group or criterion.
3. Click the **Delete** icon .
4. Click **Save**.

Joins

Use the **Joins builder** to combine data from columns of two physical model tables before it is populated to the logical model columns.

Prerequisites of using joins is:

- You have at least two physical model tables dragged to the canvas
- You have defined one or more transformations on the logical table

Example: The `Name` column in the `Product` table of your logical model is linked to the `Product Name` column of `Product_A` physical table. However, you want to pull in data from another physical table `Product_B`.


In such a case, you can join the `Product Name` column of `Product_A` physical table with the `Product Name` column of `Product_B` physical table so that data populated in the logical model is a combination of data between the two physical tables (on the basis of the selected join type).

Creating a Join

1. On the **Create Mapping <logical model name>** page, select the required logical entity, and click the **Join builder** icon on the tool bar .

Note: The **Transformation** panel doesn't display details of the physical model table. In case you click a physical model table after an entity and open the **Transformation** panel, it displays details of the last selected entity.

The **Join Builder for logical table <name of the table>** pop-up window is displayed.

2. Click the **Add table** button.
A table is displayed with **Left table**, **Right table**, and **Join type** columns.
3. Use the **<select table>** and **<select column>** drop-down options of the **Left table** and **Right table** columns to specify the columns (of the physical model tables) from which data is to be joined.
4. From the **Join type** drop-down list, select the type of join you want between the columns. The options are:
 - **Inner join:** Returns matching records between the selected left and the right table columns.
 - **Left join:** Returns all records from the left column, in addition to any matching records from the right column.
 - **Right join:** Returns all the records from the right column, in addition to any matching record from the left column.
 - **Full outer join:** Returns all the records from the left and the right columns.
5. To establish joins on multiple columns of the table, hover cursor over the table row, and click the **Add comparison** icon  that appears.
A new **<select column>** row is displayed.
6. Repeat steps 4 and 5 to define the join criteria.
7. To establish join between another set of tables, click the **Add table** button.

Note: You need to have the set of table dragged on to the canvas to be able to establish join between those.

A new set of **<Select table>** and **<Select column>** rows are displayed.

8. Repeat steps 3 and 4 to define the join.
9. Click **OK**.

The join between the tables is established and data is populated in the logical model column from the selected physical model columns on the basis of the defined join.

Note: Joins are not represented by any physical link between the tables on the canvas.

Editing a Join

1. On the **Create Mapping <logical model name>** page, select the required logical entity, and click the **Join builder** icon on the tool bar .

Note: The **Transformation** panel doesn't display details of the physical model table. In case you click a physical model table after an entity and open the **Transformation** panel, it displays details of the last selected entity.

The **Join Builder for logical table <name of the table>** pop-up window is displayed.

2. Modify your selection, as needed, and click the **OK** button.

Note: For details on using the fields for modification, see [Creating a Join](#) on page 120.

The join definitions are modified for the selected tables and columns.


Note: Joins are not represented by any physical link between the tables on the canvas.

Deleting a Join

1. On the **Create Mapping <logical model name>** page, select the required logical entity, and click the **Join builder** icon on the tool bar .

Note: The **Transformation** panel doesn't display details of the physical model table. In case you click a physical model table after an entity and open the **Transformation** panel, it displays details of the last selected entity.

The **Join Builder for logical table <name of the table>** pop-up window is displayed.

2. To delete a column or a table row, hover the cursor over the row, and click the Delete icon .
3. Click the **OK** button.

Model Store

A model store is a virtual representation of mapped logical and physical models, which once deployed, can be accessed from Spectrum Technology Platform as well as from other client applications.

The mapped logical and physical models in a model store have data federated from multiple sources, such as databases, file servers, and cloud services. You can design and customize your model store according to your requirement and for each of the logical models in the model store, you can define the cache configuration which ensures data gets refreshed periodically. All these features make model store a convenient option to store and use customized data without having to endure time lag

due to network congestion. This local cache improves query performance when flows or services of Spectrum Technology Platform reference this data.

You can create multiple model stores to suit the different business requirements and deploy these. However, to access a deployed model store through Spectrum Technology Platform you need to first create a connection to it using the **Add Data Sources** page. For more information, see [Connecting to a Model Store](#) on page 51.

You can also edit and delete a model store, however, for it you need to undeploy the model store first.

Creating a Model Store


To create a new model store, you need to perform these tasks on the **Model Store** tab of the **Modeling** page.

Note: While building a model store on a flat file connection, *periods* (.) if any, in the column name automatically gets converted to *underscores* (_).

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in with your credentials.
 3. Click **Modeling > Model Store**.
 4. Click **Create Model Store** icon .
- The **Create Model Store** page is displayed, listing the **Name**, **Selected Mapping** (of the logical model), and **Type** (physical or logical) of all the available models.
5. In the **Name** text box, enter a name for the Model Store.
 6. Select the check box corresponding to the models you want to include in the model store. To select all the listed models, select the **Include** check box in the header row.

To remove any selected model, uncheck the corresponding row.

Note: You can use the **Filter** type ahead text box to customize the grid display. For example, if the models you want to select begin with the text "Cust", you can type `Cust` in the filter box, and it will display all the models having "Cust" in their name.

The selected models get included in the model store. For logical models having multiple mappings, the default selection is displayed when you click  for that logical model.

Note: If you click the **Save** button without including any model, an error message asks you to include at least one model from the table.

7. To change the selected mapping of a logical model before including it in the model store, select the option button corresponding to that mapping or union in the **Mapping Name** column.
8. To define the cache configuration of the logical models included in the model store, click the respective **Cache configuration** button.

Note: The **Cache configuration** button is displayed after you select the **Include** check box for the logical model.

The **Cache configuration for model <name of the model>** pop-up window is displayed. For details on configuring the cache, see [Configuring Cache](#) on page 131.

Note: You can configure the cache of only one logical model at a time.

9. To save as well as deploy the model store, click **Save & Deploy**. If you selected a logical model, but did not select all the mapped physical models, a message prompts you to add the dependent physical model as well. Click **Yes** and proceed.

For successful deployment of a model store, ensure you avoid these two scenarios.

- A constituent logical model is not mapped to any physical model.
- For a logical model, any of its mapped physical model has not been selected for addition.

The model store gets saved and deployed for use in the client application.

Note: In case you did not fulfill any of the criteria mentioned above, your model store still gets saved and an error message is displayed indicating that the model store has not been deployed. For more information, see [Deploying a Model Store](#) on page 124.

10. To save the model store and deploy it later, click **Save**.


A model store needs to be deployed to be accessed from Spectrum Technology Platform as well as other client applications.

- To deploy a model store, see [Deploying a Model Store](#) on page 124.
- To access a model store from within Spectrum Technology Platform, you need to create a connection to it using the **Add Data Sources** page. For more information, see [Connecting to a Model Store](#) on page 51 in the **Administration Guide**.
- To access a model store from a client application other than Spectrum Technology Platform, you need to download the JDBC driver and install it manually. For more information about downloading and installing the JDBC driver, see [Installing JDBC Driver](#) on page 126.

Editing a Model Store

To modify an existing model store, perform these tasks on the **Edit Model Store** page.

Only an undeployed model store can be modified. To modify a deployed model store, you need to first undeploy it.

1. On the **Spectrum Discovery** home page, click **Modeling > Model Store**.
2. From the list of model stores displayed, select the model store that you want to modify, and click the **Edit Model Store** icon .

Note: You can modify only those model stores that are not already deployed. For a deployed model store, the **Edit Model Store** icon will remain unavailable, till you undeploy it.

The **Edit Model Store** page is displayed.

3. Add or remove the physical or logical models, as required, by clicking the respective **Include** rows.
4. Click **Save** or **Save & Deploy**, as needed.

For successful deployment of a model store, ensure you avoid these two scenarios.

- A constituent logical model is not mapped to any physical model.
- For a logical model, any of its mapped physical model has not been selected for addition.

Deleting a Model Store

Only an undeployed model store can be deleted. To delete a deployed model store, you need to first undeploy it.


1. On the **Spectrum Discovery** home page, click **Modeling > Model Store**.
2. From the list of models displayed, select the model store that you want to delete, and click the **Delete** icon

A message confirms successful deletion of the selected model store.


Deploying a Model Store

You can deploy a model store for use in your application and undeploy it if not needed. To do this, perform these tasks on the **Model Store** page.


1. On the **Spectrum Discovery** home page, click **Modeling > Model Store**.
2. From the list of models displayed, select the model that you want to deploy (or undeploy), and click one of these icons, as needed.

- **Deploy Model Store** : To deploy the selected model store. This icon is enabled only when you select an undeployed model store from the list.

Note: The deployment will fail if the model store contains any logical model that has all the entities unmapped.

- **Undeploy Model Store** : To undeploy an already deployed model store. This icon is enabled only when you select a deployed model store from the list.

Note: You can deploy (or undeploy) only one model store at a time. If you select multiple model stores, the icons get disabled.

3. Track the progress of model store deployment using the **Status** column. It displays one of these statuses, as applicable:
 - **In progress:** Indicates that the process of deployment or undeployment is in progress
 - **Deployed:** Indicates that the model store has been deployed
 - **Undeployed:** Indicates that the model store has been undeployed
 - **Error:** Indicates that there is an error in the deployment or undeployment process
4. At any point, click the **Refresh** icon  in the toolbar to refresh the page.


Note: If a dependent physical or logical model is deleted from a model store, the model store becomes unusable.

Previewing Model Store

To preview a deployed model store, select it from the list of model stores, click **Preview**, and run the required SQL query. You can write your query directly in the **SQL query** text box or build it using the **Model Store Entities** panel.

The preview displays a maximum of 50 records.

Note: An undeployed model store cannot be previewed.

1. On the **Spectrum Discovery** home page, click **Modeling > Model Store**.
2. From the list of model stores displayed, select the model store you want to preview, and click the **Preview** icon .
The **Preview Model Store - <name of the model>** page is displayed.
3. Use one of these methods to build your SQL query for retrieving the desired data from selected model store.

Type your query in the SQL query text box

Enclose the table name within double-quotes. Example: `select * from "EMPLOYEE_SALARY";`

If the same table name exists in two different physical models, provide the `<physicalmodelname>.table` name to run the query successfully.

- Use Model Store Entities panel to build query**
- Place cursor at relevant position in the **SQL query** text box.
 - From the **Model Store Entities** panel on the right side of the page, select the required entities (**Model > Table > Column**)

Note: Use the side arrows (>) above the entity list to view all the models in the selected model store and tables in the selected model. It also allows you to change your selection of model and table, if needed.

- Click **Insert** to insert the selected entity in the **SQL query** text box.

Note: When you click **Insert**, the selected entity is inserted where your cursor is placed in the **SQL query** text box.

- Click **Preview** after building the query.

Accessing a Model Store

A deployed model store can be accessed from within the Spectrum Technology Platform as well as other client applications. If you are accessing it from Spectrum Technology Platform, you just need to create a connection to required model store using the **Add Data Sources** page, as the JDBC driver gets installed during the Spectrum Technology Platform installation. For more information about creating connection to a model store, see [Connecting to a Model Store](#) on page 51.

However, if you are accessing it from any other application, you need to download the JDBC driver and install it manually. Once installed, you need to create connection to Spectrum Discovery using simple code so that you are able to use the model store.

Installing JDBC Driver

Use these steps to download the JDBC driver, install it, and create connection to the model store.

- On the Spectrum Technology Platform welcome page, click **Platform Client Tools**. In the options on the right pane, expand the **Driver** cascading menu, and click the **Download Jar** button for JDBC Driver.
- Save the *modelstore-jdbc.jar* file in the default browser directory.

Note: You do not need to restart the server.

Once the JDBC driver is installed, use these details to connect to the Spectrum server using a simple code or any application:

- Spectrum user name

2. Password to authenticate the Spectrum user name
3. JDBC Driver Class Name =
`com.pb.spectrum.modelstore.driver.modelstoreDriver`
4. Connection String Template =
`jdbc:spectrum://${host}:${port}/${modelstore}.`

Fill in the server IP or machine name in place of `{host}`, the port for the Spectrum Discovery in `{port}`, and the name of the model store you wish to access in `{modelstore}`.

Note: Default for `{port}` is 32750.

You can now connect to any desired model store.

Installing ODBC Driver

You can access a model store through an ODBC connection in applications supporting an ODBC interface, such as MS Excel and MS Access.

1. The version of the ODBC driver depends on the architecture of the client application.
 - If the client application version is 32-bit, download the 32-bit PostgreSQL ODBC driver [here](#).
 - If the client application version is 64-bit, download the 64-bit PostgreSQL ODBC driver [here](#).

For example, if you are using the 32-bit version of MS Office, download the 32-bit ODBC driver.

2. Run the installer `psqlodbc.msi` in case of a 32-bit ODBC driver.
In case of a 64-bit ODBC driver, run the `.exe` installer.

Follow the steps to complete the driver installation.

The ODBC driver is installed.

Creating a Data Source Name (DSN)

These instructions are specific to Windows platforms.

Attention: Ensure you have logged in to the client machine with *administrative rights*.

1. From the **Start** menu, go to **Control Panel > Administrative Tools**.
2. Double-click **Data Sources (ODBC)**.
The **ODBC Data Source Administrator** window opens.
3. Click either the **User DSN** tab or the **System DSN** tab, depending on the type of DSN you wish to create.
4. Click **Add**.
5. From the list of installed drivers on the machine, select `PostgreSQL ANSI`, and click **Finish**.
The **PostgreSQL ANSI ODBC Driver (psqlODBC) Setup** dialog box opens.

Attention: Select `PostgreSQL ANSI` or `PostgreSQL Unicode`, based on the client application through which the ODBC connection is to be used.

6. Enter the following details to map this DSN to the desired model store:
 - a) In the **Data Source** field, enter the name you wish to assign to the DSN.
 - b) In the **Database** field, enter the name of the model store to which you wish to connect through this DSN.
 - c) In the **Server** field, enter the name of the machine on which the Spectrum Technology Platform server is running.
 - d) In the **Port** field, enter the value of the property `edf.odbc.socket.port`, as configured in the `federation_config.properties` file. See [Common Configuration Properties](#) on page 129.
By default, the port is 32751.
 - e) In the **User Name** field, enter the user name of the Spectrum Technology Platform server.
 - f) In the **Password** field, enter the password of the Spectrum Technology Platform server.
 - g) Click **Datasource** in the **Options** section.
The **Advanced Options (PostgreSQL30)** dialog box opens.
 - h) On **Page 1**, configure the options as below:

Advanced Options (PostgreSQL35W) 1/2

Page 1 Page 2

☐ Disable Genetic Optimizer ☐ CommLog (C:\psqlodbc_XXXX.log)

☐ KSQO(Keyset Query Optimization) ☒ Parse Statements

☒ Recognize Unique Indexes ☐ Cancel as FreeStmt (Exp)

☐ Use Declare/Fetch ☐ MyLog (C:\mylog_XXXX.log)

Unknown Sizes

☒ Maximum ☐ Don't Know ☐ Longest

Data Type Options

☒ Text as LongVarChar ☐ Unknowns as LongVarChar ☒ Bools as Char

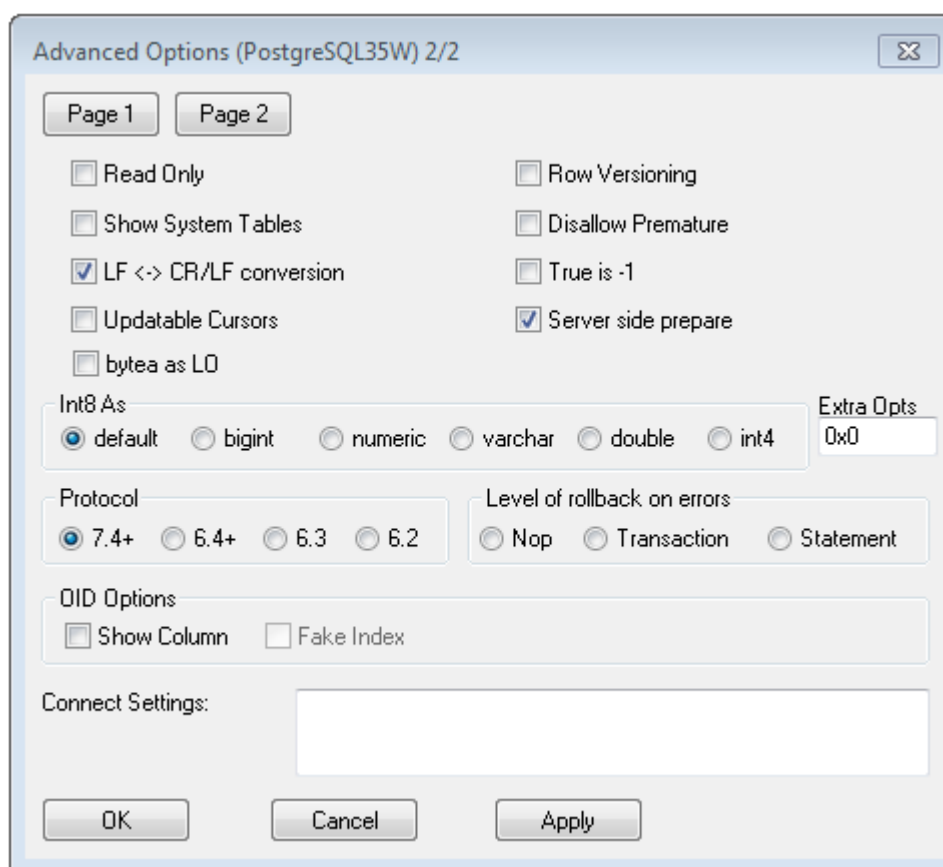
Miscellaneous

Max Varchar: 255 Max LongVarChar: 8190

Cache Size: 100 SysTable Prefixes: dd_;

OK Cancel Apply Defaults

- i) Click **Page 2**, and configure the options as below:



- j) Click **OK** to close the **Advanced Options (PostgreSQL30)** dialog box.
- k) Click **Test** to check if the created DSN connects to the configured model store successfully.

Important: Ensure the model store is deployed to test the DSN successfully.

- l) Click **Save**.

The created DSN allows you to connect to the configured model store using applications like MS Excel, MS Access.

Common Configuration Properties

Property	Description	Default Value
edf.log.file	Name of the log file to be generated for Spectrum Discovery.	edf_engine.log
edf.engine.UseDisk	Indicates if disk space is to be used as buffer for resource-heavy processes.	true
edf.socket.port	Port for the JDBC driver configurations.	32750

Property	Description	Default Value
<code>edf.odbc.socket.port</code>	Port for the ODBC driver configurations.	32751
<code>edf.socket.bindaddress</code>	The IP address or the host name on which the Spectrum Discovery application must listen, on a network. The value <code>0.0.0.0</code> indicates that the application listens on every available address.	<code>0.0.0.0</code>
<code>edf.jmx.socket</code>	The port used for Spectrum Discovery internal communication on the JMX interface. Note: This property is deprecated now, and would be removed from subsequent releases.	9877

Accessing a DSN through MS Excel

You can connect to a configured ODBC DSN from a client application, and access the mapped model store to perform the required operations. A DSN is accessible by various client applications, such as MS Excel and MS Access. This example illustrates the steps to connect to an ODBC DSN and access a model store in MS Excel.

1. Open MS Excel.
2. Go to **Data > From Other Sources > From Data Connection Wizard**.
The **Data Connection Wizard** opens.
3. Select **ODBC DSN** as the type of data source.
4. Click **Next**.
5. From the list of ODBC data sources, select the DSN you wish to connect to.
6. Click **Next**.
7. Ensure the correct model store is selected in the database dropdown.
8. If you wish to select any one of the listed tables of the model store, check the **Connect to a specific table** check box, and select the desired table.
To select all the listed tables, leave the check box unchecked.
9. Click **Next**.

The contents of selected tables are imported and displayed on the spreadsheet.

Note: The data displayed in MS Excel is read-only.

However, writes are possible while accessing a DSN using MS Access and other client applications.

Note: The example illustrates the process for a 32-bit ODBC driver.

Cache Configuration

You can cache data of all the logical models (and their entities) in a model store. This reduces the delays in accessing data due to network congestion.

Caching can be switched on or off for all or selected entities in a logical model. You can define the time for which the cached data will remain accessible, after which it will expire and will have to be reloaded.

Cache Load The cache for each model is loaded when the first query is run to access it after caching is switched on.

After caching is switched on for a model store, the execution of the first query on it might take longer than usual. This is because the cached data is first refreshed from the data source, after which the required data is fetched from the refreshed cache.

Model's Time to Live This is the duration between a cache load until the next purge of the cache. Once a cache expires, the cached data is reloaded from the data sources when the immediate next query is run.

The next **Time To Live** is calculated from the moment the cache data reload is completed. Once the expiry time is reached, the cached data is purged again.

The default **Time To Live** is 120 minutes.


Configuring Cache

You can configure the cache time for the logical models in a model store by using the **Cache Configuration** button.

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in using your credentials.
3. On the **Spectrum Discovery** home page, click **Modeling > Model Store**.
4. From the list of model stores displayed, select the model store for which you want to define the cache configuration, and click the **Edit Model Store** icon .

Note: You cannot define the cache configuration of already deployed model stores. In such cases, you need to undeploy the model store. The **Edit Model Store** icon will be enabled after deployment.

The **Edit Model Store** page is displayed, listing all the models in this model store.

5. Select the **Include** check box of the logical model and click the **Cache Configuration** button that appears.

Note: This button is displayed only for the logical models in the model store.

Once configuration is defined for a logical model in model store, the **Cache Configuration** button remains enabled for it until you uncheck the **Include** check box.

The **Cache configuration for model <name of the logical model>** pop-up is displayed showing a default cache time of **120** minutes (in the **Model's Time To Live** field) for all the entities in the model.

6. In the **Model's Time To Live** field, enter the duration (in minutes) for which you want to enable caching. The default duration is 120 minutes.

Note: The duration you enter in this field is automatically populated in the **Time To Live** fields for all the entities in the model. However, you can override the cache duration for any entity by turning the **Override TTL** selection **On**.

7. To enable caching of the entities in the model, select the respective **Enable Cache** check box. To enable caching of all the entities in the model, select the **Enable Cache** check box in the header.
8. To customize the cache duration for any of the entities in the model, click the **Override TTL** slider to **On**, and enter the required cache override duration in the respective **Time to Live** field.
9. To clear all the cache configuration overrides for the model, click the **Clear all overrides** button. The customizations done in the **Override TTL** and **Time To Live** fields are overridden and the **Time To Live** for all the entities is set to the value defined in the **Model's Time To Live** field.
10. Click the **OK** button.

The cache time is set, as specified for all the entities in the logical model and you return to the **Edit Model Store** page.

How to Video - Modeling

This video describes how to create a Logical model and map it with the physical data stores.

<https://www.youtube.com/embed/QeSi0zZwYzc>

6 - Query

Spectrum Discovery provides the **Query** application for creating, executing, and exporting queries that allow the technical users to create the 'queries' and the business users to execute and export the 'queries' where they can export the query results and use the data to generate various reports easily, with little technical know-how. It allows you to build complex SQL queries via an intuitive **Visual Query Builder** interface to help you construct complex database queries. Our **Query** application regulates access to both technical and business users via **Security** control in the **Management Console**. The technical users are required to have CRUD rights, whereas the business users are required to have the execute rights to export the queries.

The Query application also allows you to:

- Manage the list of queries on the home page. Based on your privileges, you can create, edit, view, delete, or run the queries.
- Define parameters for your query to act upon the resultset of the query dynamically.
- Export the complete set of data fetched from the query in .csv or .txt format to your desired location.
- Schedule the queries to run automatically at your scheduled time intervals.

In this section

Understanding the Home Page.....	134
Creating a Query.....	135
Exporting a Query.....	139
Scheduling a Query.....	141
How to Video - Query Business Entities.....	144






Understanding the Home Page


When you click **Query** in Spectrum Discovery, the **Query** page is displayed with these two tabs:

- **Query**
- **Schedule**

Query tab

This tab displays a list of created queries if any. The toolbar displayed on the **Query** tab allows you to perform these operations, based on your privileges:

Operation	Description
Create Query	Click  to create a new query. For more information, see Creating a Query on page 135.
Edit Query	Click  to edit a query from the list.
Configure Export	Click  to export a query from the list. For more information, see Exporting a Query on page 139.
Copy Query	Click  to copy a query from the list.
Delete Query	Click  to delete a query from the list.

You can also choose to sort your queries on the **Name** and **Last Modified On** fields. Moreover, if you have an extensive list of queries, you can filter your queries using the **Filter**  field.


The details displayed on this page are as follows:

Field	Description
Name	The name of the query.
Description	The description of the query.
Tags	The tags associated with a query.
Last Modified By	The user who last modified the query.
Last Modified On	The last modified date and time.


For information about the **Schedule** tab, see [Scheduling a Query](#) on page 141.

Creating a Query

To create a new query in Spectrum Discovery, perform these steps:

1. On the **Discovery** menu bar, click **Query**.
The **Query** page is displayed.
2. Click the Create Query button .
The **Create Query** page is displayed.
3. On the **Create Query** page, define the fields as follows:

Field Name	Description
Name	Enter a name for your query in the Name field.
Description	If you want to enter a description for your query as a reference for later use, enter it in the Description field.

Field Name	Description
Tags	<p>Add tags to your query in the Tags field to give them a relevant context and make them easily accessible later. Tags are useful in categorizing the query—for example, <i>Retail</i>, <i>Banking</i>, <i>Customer Data</i>, and more.</p> <p>The tags you add for your query appears on the corresponding query on the Query home page.</p>
Model store connection	<p>Select the type of Model Store connection you want to use to build the query.</p> <p>Note: You can only use the Model Store connections to build a query.</p> <p>To configure a new connection or edit any existing connections, click the Manage Connections link. It takes you to the Data Sources page in the Management Console. For details on configuring the Model Store connection, see Connecting to a Model Store on page 51.</p> <p>Note: Click the Refresh  button to update any modification, deletion, or addition to your list of connections.</p>
SQL	It shows the queries you build using the Visual Query Builder .
Build SQL	Create a complex query by selecting multiple columns, creating joins, and nested queries by clicking Build SQL , which opens the Visual Query Builder . For more information, see the <i>Visual Query Builder</i> section of the <i>Enterprise Data Integration Guide</i> .
Preview	To see a sample of the records fetched by the SQL query, click Preview .
Parameters	It allows you to define parameters to act upon the resultset of the query dynamically. By using different parameters, a parameterized query returns different results. For more information, see Creating a Query on page 135.

4. After you define the fields, click **Save**.

The query is displayed on the **Query** home page.

Creating parameterized query

A parameterized query allows you to create a query that can be updated easily to search upon a new term. To create a parameterized query, perform the following steps.

1. Create a query as you would normally do by using the **Visual Query Builder**.
2. Locate the field or fields where you want the parameter to appear, and place your cursor in the **Criteria** field.
3. In the **Criteria** field, enter a *question mark* ?.
As you enter a *question mark* ?, you see the {field} value being added with a suffix 1,2,3 and so on to the WHERE clause in the corresponding SQL statement, as shown in the example below.

Visual Query Builder

Select Columns Apply Operations

Output	Expression	Aggregate	Alias	Sort Type	Sort Order	Grouping	Criteria	Or...
<input checked="" type="checkbox"/>	"alidataPM"."All_DataTypes_withHeader"."b_d_...	None	None	None	None	<input type="checkbox"/>	?	None
<input checked="" type="checkbox"/>	"alidataPM"."All_DataTypes_withHeader"."d_st_...	None	None	None	None	<input type="checkbox"/>	None	None
<input checked="" type="checkbox"/>	"alidataPM"."All_DataTypes_withHeader"."e_b_...	None	None	None	None	<input type="checkbox"/>	None	None
<input checked="" type="checkbox"/>	"alidataPM"."All_DataTypes_withHeader"."f_bi_...	None	None	None	None	<input type="checkbox"/>	None	None
<input checked="" type="checkbox"/>	"alidataPM"."All_DataTypes_withHeader"."Jul_...	None	None	None	None	<input type="checkbox"/>	None	None

ADD

SQL Statement

```
Select "alidataPM"."All_DataTypes_withHeader"."b_date", "alidataPM"."All_DataTypes_withHeader"."d_string", "alidataPM"."All_DataTypes_withHeader"."e_boolean", "alidataPM"."All_DataTypes_withHeader"."f_bigdecimal", "alidataPM"."All_DataTypes_withHeader"."JulianDay" From "alidataPM"."All_DataTypes_withHeader" Where "alidataPM"."All_DataTypes_withHeader"."b_date" is #[FirstName1]
```

OK Cancel

4. Click **OK**
You see the **Create Query** page with the **Parameters** workspace being auto-populated with the field or fields you chose to parameterize in Steps 2 and 3.
5. On the **Parameters** workspace, do as follows:

Note: The parameters you define here auto-updates in the corresponding **SQL** field.

- a) Select an operator from any of these listed in the table below.

Operation	Description
equals	Checks if the value in the field matches the value specified.

Operation	Description
not equals	Checks if the value in the field does not match the value specified.
is greater than	Checks if the field has a numeric value that is greater than the value specified. This operator works on numeric data types as well as string fields that contain numbers.
is greater than or equals	Checks if the field has a numeric value that is greater than or equal to the value specified. This operator works on numeric data types as well as string fields that contain numbers.
less than	Checks if the field has a numeric value that is less than the value specified. This operator works on numeric data types as well as string fields that contain numbers.
is less than or equals	Checks if the field has a numeric value that is less than or equal to the value specified. This operator works on numeric data types as well as string fields that contain numbers.
is between	Selects values within a given range. The values can be numbers, text, or dates.
exists in	Selects multiple values, and specifies the list separated by new line. Only applicable for string or numeric fields.
like	Determines if a character string matches a specified pattern.

Note: The list of operators may vary based on the data type of your parameterized field.

- b) Enter a name for your parameter.
- c) Based on the data type of your parameterized field, you can enter or select the data in the input field.


The input field box varies based on the data type. For example, if your parameterized field is of **Date** type, the input field would be a date-picker; if the parameterized field is a **String** type, the input field would be a text box.

Note: It is mandatory to define valid values for the parameter; else, you will not be able to preview or save the query.


- d) Click **Save** to save the query.

Exporting a Query

You can export the queries on the fields you parameterized. To export a query, perform these steps:

1. On the **Query** home page, select a query you want to export and click the **Configure Export**  button.
The **Configure Export** page appears with the parameterized fields in the **Parameters** workspace.

Note: You can export a query only if you have the required privileges.

2. In the **Parameters** workspace, you have the option to either:
 - Export the query based on the default field values, or
 - Enter a new set of values on runtime to export the query.
3. After you choose to export your data, click **Preview**.
You see, the set of records fetched by the query in the **Preview** section. By default, a maximum of 50 records at a time can be shown.
4. To export the query in .csv or .txt format, do as follows:
 - a) On the **File Name** field, click the **Choose File** button.
The **Save File** pop-up is displayed.
 - b) Use the file path selection box  / [Server](#) / [E](#) / [Sample Files](#) to go to the required path on the server.
A list of files at the selected location is displayed below the file path selection box.
 - c) Select an existing file or name a different file in the **Name** field with an appropriate extension in the file name, to indicate the desired compression format to be used while exporting the file.
 - d) Click **OK**.
 - e) Select a **Field Separator** from the drop-down list to specify a character to separate fields in a delimited file. The characters available to define as field separators are:

Note: By default, the field separator is set to 'Comma'.

- Space
- Tab
- Comma
- Period

- Semicolon
 - Pipe
- f) Select a **Record Separator** from the drop-down list to specify the character to separate records in line in a sequential or delimited file. The record separator settings available are:

Note: By default, the record separator is set to 'Default'.

- **Unix (U+000A)** - A line feed character separates the records. This is the standard record separator for Unix systems.
 - **Macintosh (U+000D)** - A carriage return character separates the records. This is the standard record separator for Macintosh systems.
 - **Windows (U+000D U+000A)** - A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.
- g) Click the **Export** button to export the complete set of data fetched from the query. As you click, a spinner appears on the button to show the progress, which keeps on spinning until the file gets exported.
- The output file will be generated on the server or other available connections, such as FTP, SFTP, Cloud, HDFS, and more, as selected in the previous steps, and the system will retain the last exported path in the **File Name** field.

Note: You must have the permissions below on the entities to export a query, which can be accessed through **System > Security** in the management console:






Entity Type	Permission	
	View	Execute
Query (in Spectrum Discovery)	✓	✓
Model Store (in Spectrum Discovery)		✓
Jobs (in Platform)		✓
Resources - JDBC Drivers (in Resource Connection)	✓	
Data Source* (in Resource Connection)	✓	


* Whichever data source is being used underneath the model store connection.

Scheduling a Query

Schedule tab

This tab displays a list of scheduled queries if any. It lets you run the queries automatically at your scheduled time intervals. The toolbar displayed on the **Schedule** tab allows you to perform these operations, based on your privileges:

Operation	Description
Create Schedule	Click  to schedule a new query. For more information, see Scheduling a Query on page 141.
Edit Schedule	Click  to edit a scheduled query from the list.
View Schedule	Click  to view a scheduled query from the list. Note: Read-only information is displayed.
Copy Schedule	Click  to copy a scheduled query from the list.
Delete Schedule	Click  to delete a scheduled query from the list.

You can also choose to sort your scheduled queries on the **Name**, **Trigger**, and **Next Run** fields. Moreover, if you have an extensive list of scheduled queries, you can filter your queries using the **Filter**  field.

The details displayed on this page are as follows:


Field	Description
Name	The name of the scheduled job.

Field	Description
Query Name	The name of the query.
Scheduled By	The user who has scheduled the query.
Trigger	The status of the scheduled query, such as enabled or disabled. While scheduling a query, if the Schedule field is toggled to ON , the status is shown as Enabled , and if toggled to OFF , the status is shown as Disabled .
Next Run	It displays the next scheduled run date and time.
Last Run	It displays the last scheduled run date and time.


Scheduling Query

To schedule a query to run automatically at your scheduled time intervals, perform these steps:

Note: You must have the CRUD rights to create, view, edit, or delete the scheduled queries.

1. On the **Spectrum Discovery** menu bar, click **Query**.
The **Query** page is displayed.
2. On the **Schedule** tab, click the **Create Schedule**  button.
The **Create Schedule** page is displayed. On this page, define these fields:

Field Name	Description
Name	Enter a name for your scheduled job in the Name field.
Query Name	Select a query from the drop-down list for which you wish to schedule the job.

Field Name	Description
Schedule Section	<p>Select any of these time trigger from the drop-down list:</p> <ul style="list-style-type: none"> • Date/Time - You can schedule the date and time on which you wish to run the job by clicking this  button. • Recurring Date/Time - You can schedule a recurring date and time event for your job. <ul style="list-style-type: none"> Note: When you select this option, Pattern, Time, and Range sections appear below the trigger. Use these for setting up a recurring job, as described below. <p>a. Pattern (of days or week or month):</p> <ul style="list-style-type: none"> • Every weekday: To run the job on every day of the week. • Every: At a specified periodicity. For example, to run a job every alternate day, choose this option, and enter 2 in the selection box. • Day <> of every <> month(s): To run a job on a specified day and month. For example, to run a job on the third day of every alternate month, enter 3 in the Day selection box and 2 in the month(s) selection box. • Every <> month(s) <on a specified day> <of a week>: For example, to run the job every third month on the second Tuesday, enter 3 in the month(s) selection box, and from the dropdowns, select on the second and Tuesday options respectively. • Days of the week: Select the days of the week on which you want to run the job. <p>Note: You can select multiple days.</p> <p>b. Time: To run the job at a specified interval of time, select this option and, in the selection boxes, specify the number of hours or minutes after which you want to run the analysis. For example, every three hours.</p> <p>c. Range: To specify the range of analysis for the job (at a specified time between a range of dates), select the Start date and End date from the respective calendars and select the time.</p> <p>Note: Select the No end date checkbox if you do not want to specify an end date for the recurring job.</p>
Schedule	<p>Toggle the Schedule button to ON or OFF to enable or disable the schedule based on your requirements. If the Schedule button is toggled to ON, the status is shown as Enabled, and if toggled to OFF, the status is shown as Disabled, on the Schedule home page.</p>

Field Name	Description
Parameters	<p>It displays the default set of parameterized values based on the query you choose. You can choose to create different schedules on a different set of parameters. For example, different users can create schedules for the same query using a different set of parameters.</p> <p>By using different parameters, a parameterized query returns different results. For more information, see Creating Parameterized query section in Creating a Query on page 135.</p>
Export Details	<p>It allows you to export the query results to your selected location. For more information about exporting the query results, see Exporting a Query on page 139.</p> <p>Note: It is mandatory to select a file path to export a query. In case you copy the scheduled job, the file path will not be copied, and you will have to define a new path.</p>

- After you define the fields, click **Save**.
The scheduled query is displayed on the **Schedule** home page.

Note: When a query is scheduled, the query and its parameters are saved along with the schedule details. If this query is updated, it will not reflect in the schedule unless the schedule is updated, and the query is selected again to pull the updates.

In case the schedule is not updated after the query update, it will keep running with the initially saved query.

How to Video - Query Business Entities

This video explains how you can build and export report from business entities.

<https://www.youtube.com/embed/00oVSRUXRzM>

7 - Preparing Quality Rules

Data quality involves ensuring the accuracy, timeliness, completeness, and consistency of the data used by an organization so that the data is fit for use. We support the data quality initiatives by providing the capabilities to prepare matching and consolidation rules to fix data anomalies and add more context in your enterprise data assets. Matching is the process of identifying records related to each other in some way that is significant for your purposes. For example, if you are trying to eliminate redundant information from your customer data, you may want to identify duplicate records for the same customer; or, if you are trying to eliminate duplicate marketing pieces going to the same address, you may want to identify records of customers that live in the same household.

A match rule defines the criteria that are used to determine if one record matches another. Match rules specify the fields to compare, how to compare the fields and a hierarchy of comparisons for complex matching rules. A consolidation rule is a Best of Breed rule which consolidates duplicate records by selecting the best data in a duplicate record collection and creating a new consolidated record using the best data. You define the rules that select records to consolidate.

How Spectrum Discovery helps you prepare rules

The **Prepare** module on the **Discovery** page allows you to create, modify, delete, and publish the rules. It lists all the rules built for the sources in a dashboard view, a one-stop-shop for all your rules management. You can create the rules manually or by machine learning based solutions where the matching algorithms and thresholds are learned automatically based on the user's matching scenario. Spectrum Discovery helps you to build these rules and allows you to associate them with the *Business Glossary* entities.

In this section

Accessing Prepare Home Page.....	147
Selecting a source.....	151
Creating Rules.....	156
How to Video - Prepare Dashboard.....	191



How to Video - Preparing Manual Matching and Consolidation Rule.....	191
How to Video - Preparing Smart Consolidation Rule.....	191
How to Video - Preparing Smart Match Rule.....	191

Accessing Prepare Home Page

To access the **Prepare** home page, click **Prepare** on the **Discovery** page main menu, which displays the home page.

Home page features

The home page contains these features:

- Provides a unified list of your quality **Rules**, such as the matching and consolidation rules.
- Provides filtering capabilities to limit the data being displayed on the dashboard based upon specified parameters. For example, you can filter the data based upon **Source**, **Entity**, and **Status**.
- Provides individual statuses for the rule, focusing on items such as **Rule Type**, **Source Desc.**, **Source**, **Source Type**, **Linked Glossary Entities**, **Created By**, **Last Updated**, **Exceptions**, and **Status**.
- Provides the capabilities to manage your rules such as creating or deleting a rule, based upon your access privileges, and allows you to associate a rule to the Business Glossary entities. A business glossary is a list of business terms with a clear and consistent set of definitions, enhancing data governance across the organization. For more information, see [Defining Business Glossary](#). The match rules you associate with a business glossary keeps a business user aware of the associated match rules to gain better insights while working on a business entity.
- Provides the capability to refresh the data displayed on the dashboard manually.
- Provides an enhanced capability to drill down into the rule to further modify or enhance them based on your business needs.

Home page GUI

The screenshot displays the 'Prepare' section of the Spectrum Discovery GUI. It features a top navigation bar with various functional tabs and a user profile. A left-hand sidebar provides quick access to different views: Dashboard, Matching, and Consolidation. The main workspace is dedicated to managing rules, with a 'Filter By' section at the top allowing users to refine their view by source, entity, or status. Below the filters, a table lists existing rules with detailed information such as rule names, types, source descriptions, and creation dates. The table is paginated, showing the first of ten items.

The home page GUI consists of a selection pane on the left and a display pane on the right.

Selection pane Lets you choose from a list of selection modes, such as **Dashboard**, **Matching**, and **Consolidation**.

Display pane Displays the unified list and status of the rules and further allows you to create, modify, or delete them based upon your requirements.


The following table describes the meaning of different columns listed on the **Dashboard**, **Matching**, and **Consolidation** modes.

Column	Description
Rule Type	Displays the rule type, such as Match Rule or Consolidation Rule . The Matching mode categorizes the matching rules as Manual Rule or Smart Rule , whereas the Dashboard mode does not categorize the matching rules and instead displays them as a Match Rule .
Source Desc.	Describes the source from which the rule was derived or generated. For more information, see Using Source Description in Selecting a source on page 151.
Source	Displays the source type and the file or table name. For example, if the source type is Flat file and the file name is xyz , Flat File-xyz is displayed.

Column	Description
Source Type	Displays the type of configured data source for the rule, such as a Flat File or Connection . For more information, see Selecting a source on page 151.
Linked Glossary Entities	<p>Displays the glossary entities linked with the rule, such as Customer, Product, Retail, Banking, and more, which helps to categorize them. A business glossary is a list of business terms with a clear and consistent set of definitions, enhancing data governance across the organization. For more information, see Defining Business Glossary. The match rules you associate with a business glossary keeps a business user aware of the associated match rules to gain better insights while working on a business entity.</p> <p>The rules that you associate with the business glossaries are also displayed in the Data Quality tab on the Glossary Entity Wiki page, which further allows you to drill down into the rule to modify or enhance them. For more information, see Glossary Entity Wiki.</p>
Created By	Displays the user who created the rule.
Last Updated	Displays the date and time when the rule was last updated in DD/MM/YYYY HH:MM format.
Exceptions	<p>Displays the number of exceptions found in the rule and allows you to re-tune them.</p> <p>Note: It is available only in case of a smart match rule and appears on the Matching mode screen only.</p>
Status	<p>Displays the current status of the rule.</p> <p>Draft - Indicates that the rule is in a preliminary state and does not exist in the repository.</p> <p>Published - Indicates that the rule exists in the repository, and a new rule with the same name cannot be created.</p>





Dashboard mode

This option lets you view the complete **Prepare** dashboard with a unified list of matching and consolidation rules.

- You can create a new rule, such as a matching or consolidation rule, by clicking the  **Create Rule** button, which navigates you to the **Select Source** page. For more information, see [Selecting a source](#) on page 151 and see [Creating Rules](#) on page 156.
- You can filter the data, look upon the existing rules statuses, and further drill down to modify or enhance them by clicking on their names, navigating you to the rule.

Matching mode

This option gives you the same experience as the **Dashboard** mode, but with an enhanced capability to manage your match rules. The toolbar on this page allows you to perform these operations:

Operation	Description
Create Rule	Click  button to create a new rule, which navigates you to the Select Source page. For more information, see Creating Rules on page 156.
Delete Rule	Click  button to delete an existing rule from the repository.
Link rule to glossary entities	<p>Click  to link a glossary entity to the rule. As you click on this button, a Link Glossary Entities pop-up is displayed with a list of published glossary entities. You can select the required entities from the list and link them with the rule.</p> <p>Note: Only the published entities, displayed in the pop-up, can be linked to a rule. The rules linked with an entity are maintained for that particular version of the entity only. For example, if a published xyz entity is on 1.0 version, and you link the rule to this version, the rule will be maintained for 1.0 version only. In case the xyz entity gets updated to version 1.1, you will have to re-link the rule.</p> <p>Note: You must have the view rights to view the entities that are available to link and modify rights on both match rule and glossary entity to link an entity to a rule.</p> <p>The rules you associate with a business glossary keeps a business user aware of the associated rules to gain better insights while working on a business entity. For more information about business glossary, see Defining Business Glossary.</p> <p>The rules that you associate with the business glossaries are also displayed in the Data Quality tab on the Glossary Entity Wiki page, which further allows you to navigate to the rule page to modify or enhance them. For more information, see Glossary Entity Wiki.</p> <p>Note: You can link multiple glossary entities to a rule.</p>
Refresh	Click  button to refresh the list.

Consolidation mode

This option gives you the same experience as the **Matching** mode, but with an enhanced capability to manage your consolidation rules.

Selecting a source

When you click the **Create Rule** button on the **Discovery Prepare** dashboard, you are navigated to the **Select Source** page, which allows you to configure a source for your rule. It can be one of these:

- **Flat File** from your system.
- **Connection** configured in Spectrum Technology Platform. For details on configuring the connections, see [Defining Connections](#) on page 12.


Using Source Description field

The **Enter source description** field, by default, auto-populates your chosen file name or the table name you select from the connection, but you can also enter a description of your choice. The description you enter here appears in the corresponding **Source Desc.** column on the dashboard. This field attaches the matching and consolidation rules to the source configuration, which might help you understand the associated rules attached to a particular table later. You can select any one table at a time from the drop-down list to create the rules. By entering the name of your choice, this field helps to easily identify the tables associated with a connection type data source as the actual table names could be confusing to understand.

Note: If the source has already been configured for a table, an error is displayed, which helps you to go to the **Source Details** page where it already exists. The **Source Details** page lists all the rules for a particular source, where you can switch between different table descriptions from the drop-down list to view the associated rules. For more information, see [Creating Rules](#) on page 156.

Adding Data from Flat File

These steps describe creating a match rule from records in flat files on your workstation or on the server.

1. On the **Discovery** application page, click **Prepare**.
2. Click the Create Rule button .
This displays the **Select Source** page.
3. Next to **Select data source**, click the **Flat File** option.
4. Click the **Select File** button.
This opens the **Choose File** dialog box.
5. Click on **Server** and click to navigate to locate the data file on the Spectrum server.


A list of files at the selected location is displayed below the file path selection box.

6. Select the file you need, and click **OK**.

Note: You can enter the file name or part of the name in the **Filter** box to locate a specific file in the list.

7. To modify settings for the file, select it, and make changes as described in this table.

Settings	Description
Character encoding	The text file's encoding. Select one of these:
UTF-8	Supports all Unicode characters and is backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html .
UTF-16	Supports all Unicode characters but is not backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html .
US-ASCII	A character encoding based on the order of the English alphabet.
UTF-16BE	UTF-16 encoding with big endian byte serialization (most significant byte first).
UTF-16LE	UTF-16 encoding with little endian byte serialization (least significant byte first).
ISO-8859-1	An ASCII character encoding typically used for Western European languages. Also known as Latin-1.
ISO-8859-2	An ASCII character encoding typically used for Eastern European languages. Also known as Latin-2.
ISO-8859-3	An ASCII character encoding typically used for Southern European languages. Also known as Latin-3.
ISO-8859-9	An ASCII character encoding typically used for Turkish language. Also known as Latin-5.
CP850	An ASCII code page used to write Western European languages.
CP500	An EBCDIC code page used to write Western European languages.
Shift_JIS	A character encoding for the Japanese language.
MS932	A Microsoft's extension of Shift_JIS to include NEC special characters, NEC selection of IBM extensions, and IBM extensions.


Settings	Description						
Field Delimiter	<p>Specifies the character used to separate fields in a delimited file.</p> <p>For example, this record uses a pipe () as a field delimiter:</p> <pre>7200 13TH ST MIAMI FL 33144</pre> <p>The characters available as field delimiter are:</p> <ul style="list-style-type: none"> • Comma • Semicolon • Pipe • Tab • Space • Period <p>You can also add custom field delimiters. To add a custom field delimiter, follow these steps:</p> <ol style="list-style-type: none"> Click the Add button  placed adjacent to Field delimiter. Add Separator pop-up window is displayed. Enter your desired field delimiter in the character field, the corresponding Unicode is displayed automatically. Enter a suitable name for your delimiter in the Description field. Click Save, your delimiter appears in the Field delimiter drop down. 						
Text qualifier	<p>The character used to surround text values in a delimited file.</p> <p>For example, this record uses double quotes (") as a text qualifier.</p> <pre>"7200 13TH ST" "MIAMI" "FL" "33144"</pre> <p>The characters available to define as text qualifiers are:</p> <ul style="list-style-type: none"> • Single Quotes (') • Double Quotes (") 						
Line separator	<p>Specifies the character used to separate records in line in a sequential or delimited file.</p> <p>The record separator settings available are:</p> <table> <tr> <td>Unix (U+000A)</td><td>A line feed character separates the records. This is the standard record separator for Unix systems.</td></tr> <tr> <td>Macintosh (U+000D)</td><td>A carriage return character separates the records. This is the standard record separator for Macintosh systems.</td></tr> <tr> <td>Windows (U+000D U+000A)</td><td>A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.</td></tr> </table>	Unix (U+000A)	A line feed character separates the records. This is the standard record separator for Unix systems.	Macintosh (U+000D)	A carriage return character separates the records. This is the standard record separator for Macintosh systems.	Windows (U+000D U+000A)	A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.
Unix (U+000A)	A line feed character separates the records. This is the standard record separator for Unix systems.						
Macintosh (U+000D)	A carriage return character separates the records. This is the standard record separator for Macintosh systems.						
Windows (U+000D U+000A)	A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.						

Settings	Description
First row is header row	<p>Specifies if the first record in a delimited file contains header information. A <code>Yes</code> indicates it has header information.</p> <p>For example, this file snippet shows a header row in the first record.</p> <pre>"AddressLine1" "City" "StateProvince" "PostalCode" "7200 13TH ST" "MIAMI" "FL" "33144" "One Global View" "Troy" "NY" "12180"</pre>

- Click **Save and Continue** to save your changes.
You are now ready to create your match rule.

Adding Data from a Connection

To add data from any of the database connections configured in Spectrum Technology Platform, perform these steps:

- On the **Discovery** application page, click **Prepare**.
- Click the Create Rule button .
This displays the **Select Source** page.
- Next to **Select data source**, click the **Connection** option.
- In the **Select connection** drop-down list box, select the connection.
To narrow down your search by connection type or its name, type it in the field, and the related connections display below it. For example, if you type `mssql`, then all the MSSQL server connections display.

Note: The drop-down list displays all the connections that are configured in Spectrum Technology Platform. To configure a new connection, use the **Add Data Sources** page, which you can access through the **Connect** tab in Spectrum Discovery or the Management Console. For steps to configure a new data source, see the section [Connections](#) on page 12.
- Click the **Select tables** button.
The **Select tables** pop-up window is displayed showing all the tables in the selected connection.
- From the list of tables in the **Select tables** pop-up window, select the tables that you want to include, and click **OK**.
This displays the selected tables below the **Select tables** button.
- Select a column from the table.

Note: Use the type-ahead search text box to search for specific tables and columns.

8. Click **Save and Continue** to save your changes.
You are now ready to create your match rule.

Viewing Profile Statistics

When you select a data source to build a rule, the **Profile Statistics** panel extracts and displays the corresponding table's statistics to help you analyze and review the completeness and uniqueness of the data values to build a better rule.

Note: The statistics displayed on the panel are non-editable. It is intended for viewable purposes only.

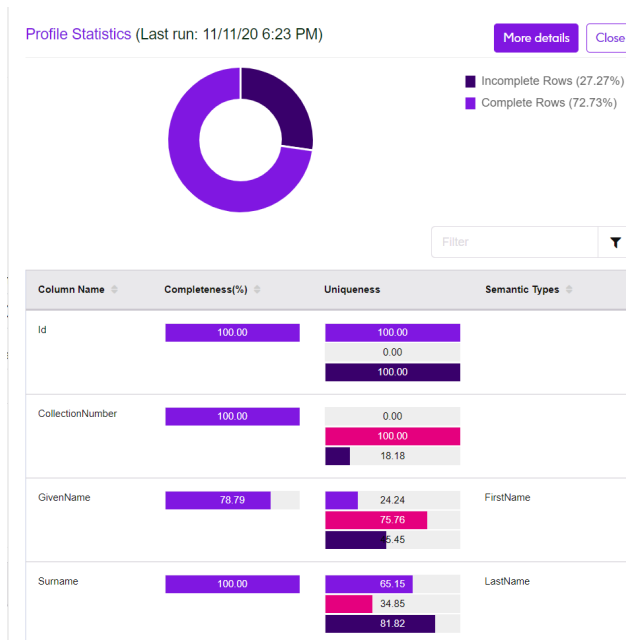
The **Profile Statistics** panel allows you to:

- View a table and its corresponding column's statistics.

Note: The statistics are based on when the profile last ran, which could be days, months, or years. If you need the latest statistics, assuming that the source's data could have changed from its previous run, re-run the profile by clicking the **Run Profile** button, which navigates you to the **Profile & Monitor** module.

- View the detailed profiling statistics of a table by clicking **More details** adjacent to **Profile Statistics**. If you click on **More details**, you are navigated to the **Data Profiling Results** page with the corresponding tables and columns' details. For more information, see [Viewing Data Profiling Results](#) on page 232.

Note: The profile statistics is available only if you had run the profile against the selected table. In case your selected table has never been profiled, you must run the profile against the table at least once to make the profile statistics available. To run a profile, click the **Run Profile** button adjacent to the table name on **Profile Status** column, or open the panel and click the **Run Profile** button. You are navigated to the **Add Profile** page. For more details on how to create and run a profile, see [Creating a Profile](#) on page 211.



The **Profile Statistics** panel displays these comprehensive profiling statistics of a column associated with the table:

- **Completeness:** Displays the completeness of records in the column.
- **Uniqueness:** Displays the uniqueness of the data contained in the column by showing the percentage of these statistics:
 - **Unique:** Records with no duplicates in the data source.
 - **NonUnique:** Records having duplicates in the data source.
 - **Distinct:** A list of all records present in your data source irrespective of those being unique or non-unique record.
- **Semantic Types:** Displays the associated semantic types.

Creating Rules

After you successfully configure the source, you are navigated to the **Source Details** page, which lists all the rules for a particular source and the rules for all the tables in case of a connection type data source. You can switch between different table descriptions from the drop-down list to view the associated rules. You can choose between the **Matching** and **Consolidation** process to create the relevant rules based upon your needs on this page. This page also lists the published and saved matching and consolidation rules, if any, which you can choose to enhance further based upon your needs.

Note: In case you edit any of the existing rule, the rule will be over-written.

To create a new rule, do as follows:

- Click the **Create matching rule** button to create a matching rule, which navigates you to the **Match Rule Type** page to choose between manual or smart match rule. For more information, see [Creating Manual Match Rule](#) on page 158 and [Creating Smart Match Rule](#) on page 170, or
- Click the **Create consolidation rule** button to create a consolidation rule, which navigates you to the **Rule Type** page to choose between manual or smart consolidation rule. For more information, see [Creating Manual Consolidation Rule](#) on page 174.

Viewing published and draft rules on source details page





When a rule is saved or published, it is available in the following states:

- **Draft** - Indicates that the rule is in a preliminary state and does not exist in the repository.
- **Published** - Indicates that the rule exists in the repository, and a new rule with the same name cannot be created.

The **Source Details** page lists the published and draft rules in a card format, which contains this information:




- **Created By:** Displays the user who created the rule.
- **Last Updated:** Displays the date and time when the rule or match key was last updated in DD/MM/YYYY HH:MM format.
- **Rule Type:** Displays if the rule is a smart rule or a manual rule.

Moreover, you can also perform these operations on the rule card:

Operation	Description
Edit rule	Click  to edit a rule. If you click, you are navigated to the rule to edit it further.
Delete rule	Click  to delete the rule.
Expand	Click  to expand the rule card to view the data.
Collapse	Click  to collapse the rule card to hide the data.

This page gives you the filtering capabilities to limit the published and draft rule cards. On the top-right corner of the page, use these **Filter By** options:



Note: By default, both the published and draft rules are displayed.

- To view the published rules only, click the **Published**  button.
- To view the draft rules only, click the **Draft**  button.
- To view both published and draft rules, click the **Clear**  button.

Creating Matching Rules

Creating Manual Match Rule


To create a manual match rule, perform these steps:

1. On the **Source Details** page, click the **Create matching rule** button.
The **Create Match Rule** page is displayed.
2. On the **Create Match Rule** page, go to the **Manual Match Process** section and click the **Create Rule** button.
The **Manual Match Rule** page is displayed. On the **Manual Match Rule** page, you can:
 - Configure one of the predefined rules from **Template Rules** located on the left corner of the page which you can use as-is, or
 - Configure a new match rule and publish it to the repository for re-use.
3. Enter a unique name for your match rule in the **Rule name** field.
4. Specify the dataflow fields, parent or child; you want to use in the match rule and match rule hierarchy.
 - Click  button and enter a name for the parent below **Match when not true**.
 - Click  button and select a field to add to the parent from the drop-down list below **Match when not true**.

Note: All children under a parent must use the same logical operator. If you want to use different logical operators between fields, you must first create intermediate parents.

5. Define these parent options as listed in the table below, which are displayed on the parent node:

Option	Description
Match when not true	<p>It changes the logical operator for the parent from and to and not. If you select this option, records will only match if they do not match the logic defined in this parent.</p> <p>Note: If you select the Match when not true option, it negates the Matching Method options. For more information, see the section Negative Match Conditions in the Spectrum Data Quality Guide.</p>
Matching Method	<p>Select one of these from the drop-down list to determine if a parent is a match or non-match:</p> <ul style="list-style-type: none"> • All true: A parent is considered a match if all children are determined to match. This method creates an "and" connector between children. • Any true: A parent is considered a match if at least one child is determined to match. This method creates an "or" connector between children. • Based on threshold: A parent is considered a match if at least one child is determined to match. This method creates an "or" connector between children. <p>If you select this option, the Threshold field enables you to specify a threshold value. The Scoring Method determines which logical connector to use. The thresholds at the parent cannot be higher than the threshold of the children. For more information, see the matching method-to-scoring method matrix below this table.</p>
Missing Data	<p>Select one of these from the drop-down list to specify how to score blank data in a field:</p> <ul style="list-style-type: none"> • Ignore blanks: Ignores the field if it contains blank data. • Count as 0: Scores the field as 0 if it contains blank data. • Count as 100: Scores the field as 100 if it contains blank data. • Compare blanks: Scores the suspect and candidate fields as 100 if they both contain blank data; otherwise, scores the suspect and candidate fields as 0.
Scoring Method	<p>Select one of these from the drop-down list to determine the matching score:</p> <ul style="list-style-type: none"> • Weighted Average: Uses the weight of each child to determine the average match score. • Average: Uses the average score of each child to determine the score of a parent. • Maximum: Uses the highest child score to determine the score of a parent. • Minimum: Uses the lowest child score to determine the score of a parent. • Vector Summation: Uses the vector summation of each child score to determine the score of the parent. The formula for calculation is: $\sqrt{a^2+b^2+c^2} / \sqrt{n}$ <p>where a, b, and c are the scores of three children, and n is the number of children.</p> <p>For more information, see the matching method-to-scoring method matrix below this table.</p>


Option	Description
Evaluate	Click the Evaluate  button to evaluate match rule. For more information, see Evaluating a match rule on page 168.
Copy settings to	<p>It allows you to copy the same settings for any number of elements.</p> <ol style="list-style-type: none"> Use the drop-down list to select or de-select the elements. Click Apply adjacent to the Copy Settings to field to copy and apply the same settings for the selected elements. <p>Note: You can copy the parent settings to a parent element and child settings to a child element only.</p>


Matching Method-to-Scoring Method Matrix

The table below shows the logical relationship between **Matching Method** and **Scoring Method** and how each combination changes the logic used during match processing.

Scoring Method	Matching Method			Comments
	Any true	All true	Based on threshold	
Weighted Average	NA	and	and	Only available when All true or Based on threshold are selected as the Matching Method .
Average	NA	and	and	
Vector Summation	NA	and	and	
Maximum	or	NA	or	Only available when All true or Based on threshold are selected as the Matching Method .
Minimum	or	NA	or	

- Define these child options as listed in the table below, which are displayed on the child node:

Option	Description
Match when not true	<p>It changes the logical operator from and to not. If you select this option, the match rule will only evaluate to true if the records do not match the logic defined in this child.</p> <p>For example, if you want to identify individuals who are associated with multiple accounts, you could create a match rule that matches the name but where the account number does not match. You would use the Match when not true option for the child that matches the account number.</p>
Candidate field	Select this to map the child record field you select from the drop-down list to a field in the input file.
Cross match against	Select this to choose one or more field names from the drop-down list to match different fields to one another between two records.
Threshold	Enter the threshold that must be met at the individual field level for that field to be determined a match.
Missing Data	<p>Select one of these from the drop-down list to specify how to score blank data in a field:</p> <ul style="list-style-type: none"> • Ignore blanks: Ignores the field if it contains blank data. • Count as 0: Scores the field as 0 if it contains blank data. • Count as 100: Scores the field as 100 if it contains blank data. • Compare blanks: Scores the suspect and candidate fields as 100 if they both contain blank data; otherwise, scores the suspect and candidate fields as 0.
Scoring Method	<p>Select one of these from the drop-down list to determine the matching score:</p> <ul style="list-style-type: none"> • Weighted Average: Uses the weight of each algorithm to determine the average match score. • Average: Uses the average score of each algorithm to determine the match score. • Maximum: Uses the highest algorithm score to determine the match score. • Minimum: Uses the lowest algorithm score to determine the match score. • Vector Summation: Uses vector summation of the score of each algorithm to determine the match score. This scoring method is useful if you want a higher vector summation match score in one or more algorithms to get proportionately represented in the final match score. The formula for calculating the final score is: $\sqrt{a^2+b^2+c^2} / \sqrt{n}$, where a, b, and c are the scores of three different algorithms, and n is the number of algorithms used.
Evaluate	Click the Evaluate  button to evaluate match rule. For more information, see Evaluating a match rule on page 168.

Option	Description
Profile Statistics	Click the Profile Statistics  button to see the column profile statistics, which is displayed in a side-panel. For more information, see Viewing Column Profile Statistics on page 169.
Copy Settings to	<p>It allows you to copy the same settings for any number of elements.</p> <ol style="list-style-type: none"> Use the drop-down list to select or de-select the elements. Click Apply adjacent to the Copy Settings to field to copy and apply the same settings for the selected elements. <p>Note: You can copy the parent settings to a parent element and child settings to a child element only.</p>

7. To configure algorithms for your child type, click **Configure Algorithms** on the child options node to add one or more of these algorithms to determine the match in the field values:

Note: Use **Search**  to selectively configure the algorithms.

String Matching Algorithms

Acronym	<p>It determines whether a business name matches its acronym by looking for acronym data; otherwise, it creates an acronym using the first character of every word.</p> <p>For example, Internal Revenue Service and its acronym IRS would be considered a match and return a match score of 100.</p>
Character Frequency	It determines the frequency of occurrence of each character in a string and compares the overall frequencies between two strings.
Exact Match	It determines if two strings are the same.
Initials	It matches the initials for parsed personal names.
Name Variant	<p>It determines whether two names are variants of each other. The algorithm returns a match score of 100 if two names are variations of each other, and a match score of 0 if two names are not variations of each other.</p> <p>For example, JOHN is a variation of JAKE and returns a match score of 100. JOHN is not a variant of HENRY and returns a match score of 0.</p> <p>Click Edit to specify the name variant options. For more information, see the section Name Variant Finder in the Spectrum Data Quality Guide.</p>

Numeric String It compares address lines by separating the numerical attributes of an address line from the characters. See the examples below.

- In the string address 1234 Main Street Apt 567, the numerical attributes of the string (1234567) are parsed and handled differently from the remaining string value (Main Street Apt). The algorithm first matches numeric data in the string with the numeric algorithm. If the numeric data match is 100, the alphabetic data is matched using Edit distance and Character Frequency. The final match score is calculated as follows:

$$(numericScore + (EditDistanceScore + CharacterFrequencyScore) / 2) / 2$$

- If you calculate the match score of these two addresses:

123 Main St Apt 567
123 Maon St Apt 567

the match score would be 95.5, calculated as follows:

Numeric Score = 100
Edit Distance = 91
Character Frequency = 91

$91 + 91 = 182$
 $182 / 2 = 91$
 $100 + 91 = 191$
 $191 / 2 = 95.5$

SubString It determines whether one string occurs within another.

Phonetic Algorithms

Daitch-Mokotoff Soundex A Phonetic algorithm that allows greater accuracy in matching of Slavic and Yiddish surnames with similar pronunciation but differences in spelling. Coded names are six digits long, and multiple possible encodings can be returned for a single name. This option was developed to respond to the limitations of Soundex in the processing of Germanic or Slavic surnames.

Double Metaphone It determines the similarity between two strings based on a phonetic representation of their characters. Double Metaphone is an improved version of the Metaphone algorithm and attempts to account for the many irregularities found in different languages.

Koeln Indexes names by sound as they are pronounced in German. Allows names with the same pronunciation to be encoded to the same representation so that they can be matched, despite minor differences in spelling. The result is always a sequence of numbers; special characters and white spaces are ignored. This option was developed to respond to the limitations of Soundex.

Metaphone	It determines the similarity between two English-language strings based on a phonetic representation of their characters. This option was developed to respond to the limitations of Soundex.
Metaphone (Spanish)	It determines the similarity between two strings based on a phonetic representation of their characters. This option was developed to respond to the limitations of Soundex.
Metaphone3	It improves upon the Metaphone and Double Metaphone algorithms with a more exact consonant and internal vowel settings that allow you to produce words or names more or less closely matched to search terms on a phonetic basis. Metaphone3 increases the accuracy of phonetic encoding to 98%. This option was developed to respond to the limitations of Soundex.
Nysiis	<p>It is a phonetic code algorithm that matches an approximate pronunciation to an exact spelling and indexes words that are pronounced similarly—part of the New York State Identification and Intelligence System.</p> <p>For example, consider that you are looking for someone's information in a database of people. You believe that the person's name sounds like "John Smith," but it is spelled "Jon Smath". If you conducted a search looking for an exact match for "John Smith," no results would be returned. However, if you index the database using the NYSIIS algorithm and search using the NYSIIS algorithm again, the correct match will be returned because both "John Smith" and "Jon Smath" are indexed as "JANSNATH" by the algorithm. This option was developed to respond to limitations of Soundex; it handles some multicharacter n-grams and maintains relative vowel positioning, whereas Soundex does not.</p> <p>Note: This algorithm does not process non-alpha characters; records containing them will fail during processing.</p>
Phonix	It preprocesses name strings by applying more than 100 transformation rules to single characters or sequences of several characters. Nineteen of those rules are applied only if the characters are at the beginning of the string, while 12 of the rules are applied only if they are at the middle of the string, and 28 of the rules are applied only if they are at the end of the string. The transformed name string is encoded into a code that is comprised of a starting letter followed by three digits (removing zeros and duplicate numbers). This option was developed to respond to the limitations of Soundex; it is more complex and, therefore, slower than Soundex.
Sonnex	It determines the similarity between two French-language strings based on the phonetic representation of their characters. It returns a Sonnex coded key of the selected fields.
Soundex	It determines the similarity between two strings based on a phonetic representation of their characters.

Syllable Alignment	It combines phonetic information with edit distance based calculations. Converts the strings to be compared into their corresponding sequences of syllables and calculates the number of edits required to convert one sequence of syllables to the other.
---------------------------	--

Similarity and Distance Measures

Edit Distance	It determines the similarity between two strings based on the number of deletions, insertions, or substitutions required to transform one string into another.
Euclidean Distance	<p>It provides a similarity measure between two strings using the vector space of combined terms as the dimensions. It also determines the greatest common divisor of two integers. It takes a pair of positive integers and forms a new pair that consists of the smaller number and the difference between the larger and smaller numbers. The process repeats until the numbers are equal. That number then is the greatest common divisor of the original pair.</p> <p>For example, 21 is the greatest common divisor of 252 and 105: $(252 = 12 \times 21; 105 = 5 \times 21)$; since $252 - 105 = (12 - 5) \times 21 = 147$, the GCD of 147 and 105 is also 21.</p>
Jaro-Winkler Distance	It determines the similarity between two strings based on the number of character replacements it takes to transform one string into another. This option was developed for short strings, such as personal names.
Keyboard Distance	<p>It determines the similarity between two strings based on the number of deletions, insertions, or substitutions required to transform one string to the other, weighted by the position of the keys on the keyboard.</p> <p>Click Edit to specify the type of keyboard you are using: QWERTY (U.S.), QWERTZ (Austria and Germany), or AZERTY (France).</p>
Kullback-Liebler Distance	It determines the similarity between two strings based on the differences between the distribution of words in the two strings.
NGram Distance	<p>It calculates in text or speech the probability of the next term based on the previous n terms, which can include phonemes, syllables, letters, words, or base pairs and can consist of any combination of letters.</p> <p>Click Edit to enter the size of the NGram; the default is 2.</p>
NGram Similarity	<p>It determines the similarity between two strings based on the length of the longest common subsequence of phonemes, syllables, letters, words, or base pairs.</p> <p>Click Edit to specify these options:</p> <ul style="list-style-type: none"> • Ngram size: Enter the size of the NGram. The default is 2. • Drop Noise Characters: Select to replace punctuation with space. • Drop Spaces: Select to merge words.

Date Algorithms

Date It compares date fields regardless of the date format in the input records. Click **Edit** to specify these:

- **General Options—Require Month:** It prevents a date that consists only of a year from matching.
- **General Options—Require Day:** It prevents a date that consists only of a month and year from matching.
- **General Options—Match Transposed MM/DD:** Where month and day are provided in numeric format, it compares suspect month to candidate day and suspect day to candidate month as well as the standard comparison of the suspect month to candidate month and suspect day to candidate day.
- **General Options—Prefer DD/MM/YYYY format over MM/DD/YYYY:** It contributes to date parsing in cases where both month and day are provided in numeric format, and their identification can not be determined by context.

For example, given the numbers 5 and 13, the parser will automatically assign 5 to the month and 13 to the day because there are only 12 months in a year. However, given the numbers 5 and 12 (or any two numbers 12 and under), the parser will assume whichever number is first to be the month.

If you select this option, it ensures that the parser reads the first number as the day rather than the month.

- **Range Options—Overall:** It allows you to set the maximum number of days between matching dates. See the examples below.
 - If you enter an overall range of 35 days and your candidate date is December 31, 2000, a suspect date of February 5, 2001, would be a match, but a suspect date of February 6 would not.
 - If you enter an overall range of 1 day and your candidate date is January 2000, a suspect date of 1999 would be a match (comparing December 31, 1999), but a suspect date of January 2001 would not.
- **Range Options—Year:** It allows you to set the number of years between matching dates, independent of month, and day. See the examples below.
 - If you enter a year range of 3 and your candidate date is January 31, 2000, a suspect date of January 31, 2003, would be a match, but a suspect date of February 2003 would not.
 - If your candidate date is 2000, a suspect date of March 2003 would be a match because months are not in conflict, and it's within the three-year range.
- **Range Options—Month:** It allows you to set the number of months between matching dates, independent of year and day.

For example, if you enter a month range of 4 and your candidate date is January 1, 2000, a suspect date of May 2000 is a match because there is no day conflict and it's within

the four-month range, but a suspect date of May 2, 2000, is not, because of the day's conflict.

- **Range Options—Day:** It allows you to set the number of days between matching dates, independent of year and month.

For example, if you enter a day range of 5 and your candidate date is January 1, 2000, a suspect date of January 2000 is a match because there is no day conflict but a suspect date of December 27, 1999, is not, because of the month's conflict.





Matching Algorithm-to-Scoring Method Matrix

The table below describes the logical relationship between the number of algorithms you can use based on the parent **Scoring Method** you select.

Scoring Method	Algorithms	
	Single	Multiple
Weighted Average	NA	Yes
Average	NA	Yes
Vector Summation	Yes	Yes
Maximum	NA	Yes
Minimum	NA	Yes

8. Click **Ok**.

Note:




- If you define n number of parent and child elements, use **Filter**  to selectively look for the elements.
- If you want to expand or collapse all the tree nodes, click the **Expand all** and **Collapse all**   buttons.
- To view the table's profile statistics, click the Profile Statistics  button, which opens a side-panel to view the statistics. For more information, see [Viewing Profile Statistics](#) on page 155.

You can now save or publish the rule. For more information, see [Saving and publishing the rule](#) on page 170.

Evaluating a match rule

The **Evaluate** option allows you to test the results for your match rule. You can test any node you want in the match rule hierarchy.

To evaluate, perform these steps:

1. In the match rule hierarchy, choose the node you want to test and click **Evaluate**.
You see the **Evaluate** page. On this page, you can also view the profile statistics by clicking the **Profile Statistics**  button located in the **Match Rule** panel. For more information, see [Viewing Profile Statistics](#) on page 155 and [Viewing Column Profile Statistics](#) on page 169.
2. In the **Input** panel, enter the test data in these ways:
 - To enter the test data manually:
 - a. Enter a suspect record under the **Suspect** column and up to ten candidates under **Candidate** columns.
 - b. Click **Export** to save the records to a file that you can import later instead of reentering the data manually.
 - To import the test data from a file:
 - a. Click **Import**.
 - b. Select the file that contains the sample records. Delimited files can be comma, pipe, or tab-delimited and should have a header record with header fields that match the field names shown under **Candidates**. For example, a sample header record for household input would be `Name,AddressLine1, City, StateProvince`.
3. Select any of these method to evaluate your match rule:
 - a. **Current Rule:** This runs the rule defined on the **Match Rule** panel. The results are displayed for one suspect and candidate pair at a time. To cycle through the results, select the arrows  . Scores for fields and algorithms are displayed in a tree format similar to the match rule control.

Note: If you make changes to the match rule and want to apply the changes to the stage's match rule, click **Save** on the top-right corner of the page.
 - b. **All Algorithms:** This ignores the match rule and instead runs all algorithms against each field for suspect and candidate pairs. Results are displayed for one suspect and candidate pair at a time and can be cycled through using the arrows.
 - c. Your results are displayed in the **Scores** panel on the right corner of the screen with these color codes:
 - **Green:** The rule resulted in a match
 - **Gray:** The rule was ignored
 - **Purple:** The results for individual algorithms within the rule

Viewing Column Profile Statistics

The column profile statistics let you analyze and review the completeness, uniqueness, frequency, and pattern of the data values to help you identify the potentially flawed data values, which could lead to poor analytical results. In case you observe any flaws with the values, you can investigate further to improve them and build a better match rule.

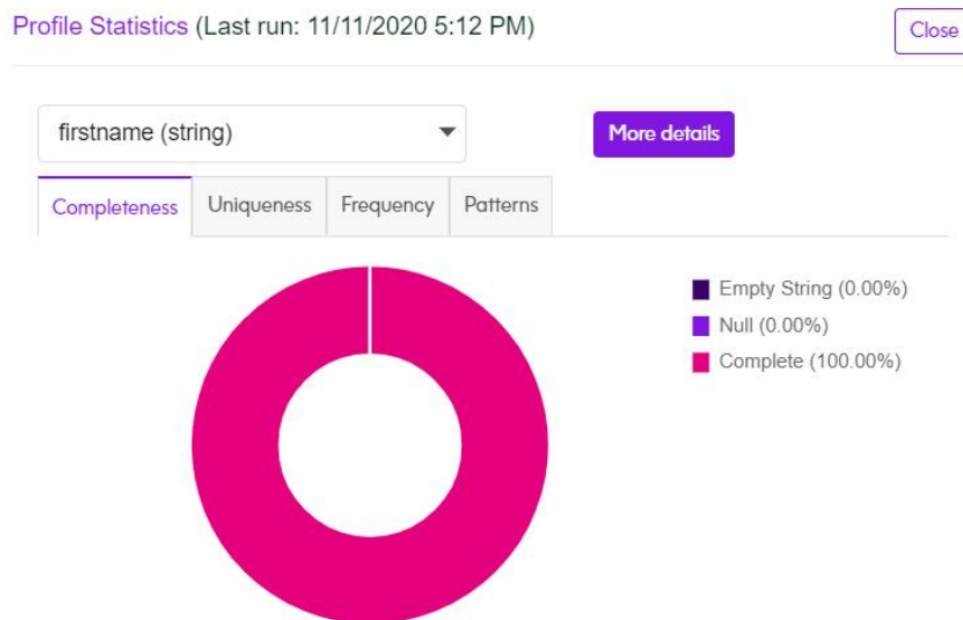
The **Profile Statistics** panel allows you to:

1. Select a column name from the **Column** drop-down to view the statistics of any particular column.

Note: The statistics are based on when the profile last ran, which could be days, months, or years. If you need the latest statistics, assuming that the source's data could have changed from its previous run, re-run the profile by clicking the **Run Profile** button.

2. View the detailed profiling statistics of the columns by clicking **More details** adjacent to **Profile Statistics**. If you click on **More details**, you are navigated to the **Data Profiling Results** page with the corresponding columns' details. For more information, see [Viewing Data Profiling Results](#) on page 232.

Note: The profile statistics is available only if you had run the profile against the selected column. In case your selected column has never been profiled, you must run the profile against the column at least once to make the profile statistics available. To run a profile, click the **Run Profile** button. You are navigated to the **Add Profile** page. For more details on how to create and run a profile, see [Creating a Profile](#) on page 211.



The **Profile Statistics** panel displays these comprehensive profiling statistics of a column:

1. **Completeness:** Displays the completeness of records in the column. The percentage of **Complete**, **Null**, and **Empty String** detected in the column is displayed in the legend.
2. **Uniqueness:** Displays the uniqueness of the data contained in the column by showing the percentage of these statistics in the legend:
 - **Unique:** Records with no duplicates in the data source.
 - **NonUnique:** Records having duplicates in the data source.
 - **Distinct:** A list of all records present in your data source irrespective of those being unique or non-unique record.
3. **Frequency:** Displays the frequency of the data contained in the column.

Note: It is displayed only if the column data type is string.
4. **Patterns:** Displays the pattern of the data contained in the column.

Note: It is displayed only if the column data type is string.

Saving and publishing the rule

After configuring and evaluating the match rule, you can save or publish the match rule according to your requirements.

- To save the rule, click the **Save** button.

The rule is saved as a **draft** and is displayed on the **Source Details** page and the **Home page**, which indicates that the rule is in a preliminary state and does not exist in the repository.

- To publish the rule in the repository, click the **Publish** button.

The rule is published to the repository and is displayed on the **Source Details** page and the **Home page**.

Note: In case you make changes to a published rule, you cannot save it as a draft copy. You will have to re-publish the rule to save the changes.

For more information, see *Viewing published and draft rules on source details page* in **Creating Rules** on page 156.

Creating Smart Match Rule

A smart match rule is prepared with a process that combines machine learning and human expertise to accelerate and automate the process of record matching. It enables business users to actively participate in the matching process, making it less reliant on the IT workforce.

To create a smart match rule, perform these steps:

1. On the **Source Details** page, click the **Create matching rule** button.
The **Create Match Rule** page is displayed.

2. On the **Create Match Rule** page, go to the **Smart Match Process** section and click the **Create Rule** button.
3. On the top-left corner of the page, enter a name for your match rule in the **Match rule name** field, and proceed with these three stages as shown below:



Selecting Columns

In this section, columns of your sample data are displayed in a tabular format. You must select the columns on which you would like to perform matching.

This procedure describes how to select columns for creating groups and generating match criteria:

1. Click the **Detect Semantic Type** button. The detected semantic types in the selected records is displayed in the **Semantic Type** column. By default, **NONE** is displayed.
If the desired semantic type is not displayed, you can select it from the drop-down after selecting the corresponding check-box of that column.

Note: This step is recommended for generating better match criteria. Based on the selected semantic type, relevant algorithms are used for generating match criteria. For example, *phonetic algorithms* are used for the semantic type *name* and not for *phone numbers* and *zip code*.



2. Slide the **Smart Sampling** to **On** to consider all the entire set of records for sampling. When **Off**, the first 20K records are taken for sampling.
3. Select the **Column Name** check-box for the columns to be selected for generating match criteria.
4. Use the **Handling Nulls** column to specify how to treat the null values in the respective columns. The options are:
 - **Null as match:** To treat the vacant fields equivalent to the corresponding field of a record pair
 - **Null as non-match:** To treat the vacant fields as non-equivalent to the corresponding field of a record pair

Note: This is the default value.

The selection made here reflect in the **Enterprise Designer** under the *missing data* option of the match rule. If you select **Null as match**, **Count as 100** is pre-selected, and if you choose **Null as non-match**, **Count as 0** is pre-selected.

Note: This option is applied globally to a field; it will remain uniform for various conditions of a field.

5. Rank your columns in the order you want those sampled. To rank, place your cursor at the extreme left of the column, and move it up or down when the cursor changes to a hand.

6. Click the  icon to save your changes and move to the next stage.
7. Click the  icon to cancel your current task.

Based on the selected columns and unsupervised machine learning algorithms, groups of records are automatically generated and these are displayed on the next page for tagging.

Tagging Records

This page is for reviewing the pairing and tagging done by the system. You need to specify if you consider the pairs a **Match**, **Non-Match** or you are **Unsure** of it. The system generates the **Match Rule** and **Match Key** based on your feedback.

Note: The pre-tags are suggestive only and should be reviewed thoroughly.

Use one of these options for reviewing the tags, as needed.

Perform a Bulk Action

After thoroughly reviewing, you can choose to tag multiple record pairs across all pages as **Match**, **Non-Match**, or **Unsure** in one go by using the **Bulk Action** option. Follow these steps to perform this action:

1. Select multiple record pairs using the respective check-boxes. To select all the records displayed on the page, click the check-box in the header row.
2. From the **Bulk Action** drop-down, select the required option (**Match**, **Non-Match**, or **Unsure**).
3. Click the **Apply** button.

Note: Uncheck all the record pairs which were a part of bulk action to continue manual tagging.

Use Filters

Use any of the required filters on the top of the table to select the required set of data.

- **All** - This displays all the record pairs.
- **Match** - This displays the record pairs which are tagged as a match.
- **Non- Match** - This displays the record pairs which are tagged as non- match.
- **Unsure** - This displays the record pairs which are tagged as unsure.

Save and Hide Tagged Records

- To save the tags specified by you, click the **Save Tagging** button on the top right of the page just above the table. Your specifications will remain saved across multiple sessions.
- To reset the tagging done so far, click the **Reset Tagging** button.

- To hide the tagged records so that you can clearly review the ones you were unsure of, click the **Hide Tagged Records** check-box on the right side in the header row.

Note: You must tag all record pairs to generate an accurate match rule. If adequate records are not tagged, the generated match criteria might be incorrect.

Analyzing Results

The **Analyze Results** page displays the generated nested Boolean match rule, and potential match key components learnt from the information provided by you. The match rule can be reviewed and exported to the *match rules* repository of **Match Rules Management** option in the **Enterprise Designer**; this can further be consumed in your batch jobs. The potential match key components can be used in the **Match Key Generator** stage of the **Enterprise Designer** after reviewing.

Match Rule Tab

This tab is divided into two sections:

- The left pane displays the match rule. On expanding the **Rule**, you can view all the conditions and sub-conditions.
- A preview of these conditions is displayed in a tabular format on the right pane of the screen. It shows these details:
 - An **Attribute** such as **Threshold**, **Scoring Method**, **Algorithms**, and **Missing Data Method**.
 - **Value** for each of these attributes.

Note: The Spectrum Smart Data Quality (SDQ) is integrated with Data Stewardship, which helps you improve the match rules based on the exception handling done in Data Stewardship. When you save the manual updates to the records in Data Stewardship, it reflects as a notification on the **Projects** page in SDQ, corresponding to the project you made modifications to.

Note: Finally, the Data Stewardship Data Quality page provides information regarding trends across data flows and stages.

Match Key Tab

This tab displays potential match key components in a tabular format. It also displays the **Column** in which the match key component was detected along with the **Algorithm** to be used. The **Average Group Size** helps you determine the average size of the group for your match key, which is generated for the complete dataset. You get the average of each value based on each match key combination to avoid any loss of accuracy. You can review and choose to consume any of the potential match key components based on your scenario by adding these in the **Match Key Generator** stage of **Enterprise Designer**.


Note: As of now, only the *Substring* algorithm is supported.

Example: This table displays a potential match key- **Match Key 1** detected in the **phone** column with an average group size of 2. The algorithm to be used is **SUBSTRING (1, 7)**, where **1** is the starting index, and **7** is the last index to be specified in the options of the **Match Key Generator** stage. The starting index is fixed to **1** for all potential match key components.

Match Key	Column	Algorithm	Average Group Size
Match Key 1	phone	SUBSTRING (1, 7)	2

Based on the actions performed by you : **Variations present in the sample data uploaded**, **Columns selected for matching** , and **Records Tagged**, the system has unlocked patterns present in your data to provide you with a match rule and potential match key components. It is suggested to test the generated results on your dataset.

Evaluating and publishing the smart rule

For evaluating a smart match rule, click the **Evaluate** button, which navigates you to the **Evaluate** page. For more information, see [Evaluating a match rule](#) on page 168. You can also view the profile statistics by clicking the **Profile Statistics**  button located in the **Match Rule** panel on this page. For more information, see [Viewing Profile Statistics](#) on page 155 and [Viewing Column Profile Statistics](#) on page 169.

After reviewing the generated match rule, you can publish it to the repository by giving a unique rule name and clicking the **Publish Rule** button. The rule is displayed on the **Source Details** page and the **Home page**. For more information, see *Viewing published and draft rules on source details page* in [Creating Rules](#) on page 156.

Creating Consolidation Rules

Creating Manual Consolidation Rule

A manual consolidation rule is a Best of Breed rule which consolidates duplicate records by selecting the best data in a duplicate record collection and creating a new consolidated record using the best data. This consolidated record is known as the best of breed record. You define the rules that select records to consolidate. When processing completes, the best of breed record is retained by the system.

To create a manual consolidation rule, perform these steps:

1. On the **Source Details** page, click the **Create consolidation rule** button.
The **Create Consolidation Rule** page is displayed.
2. On the **Create Consolidation Rule** page, go to the **Manual Consolidation Process** section and click the **Create Rule** button.

The **Manual Consolidation Rule** page is displayed. On this page, you can:

- **Create and publish a rule:** Define your rule, and publish it to the repository. You can also use the **Create new rule** button at the left corner of the page to remove the current rule and define a new rule.
 - Configure the options for the rule in the **Template Record Settings** and **Best of Breed Settings** panel. For more information, see [Defining Template Record Settings](#) on page 175 and [Defining Best of Breed Settings](#) on page 178.
3. Enter a unique name for your rule in the **Rule name** field.
You must now proceed to configure the options.

Configuring the Options

Defining Template Record Settings

In Best of Breed processing, the **Template Record** is the record in a collection that is used to create the best of breed record.

The template record is used as the starting point for constructing the best of breed record and is modified based on the **Best of Breed Settings** you define. The Best of Breed stage can select the template record automatically, or you can define rules for selecting the template record.

Example

Template record rules are written by specifying these options - **Field name**, **Field type**, **Operator**, **Value type**, and **Value**.

If you define the template record options as below:

Field name: MatchScore

Field type: Numeric


Operator: Equal

Value type: String

Value: 100

The template rule will select the record in the collection where the Match Score is equal to the value of 100.

To define a template record rule, go to **Template Record Settings** in the **Rule Configuration** panel and perform these steps:

1. Select the option **Define template record**.
2. Click the **Add rule**  button.
3. Define these options as listed in the table below, which are displayed on the **Rules** node:

Option	Description
Field name	It specifies the name of the dataflow field whose value you want to evaluate to determine if the record should be the template record.
Field type	<p>Select one of these from the drop-down list to specify the type of data in the field:</p> <ul style="list-style-type: none">• Non-Numeric: Select this option if the field contains non-numeric data, for example, string data.• Numeric: Choose this option if the field contains numeric data, for example, double or float.

Option	Description
Operator	<p>Select one of these from the drop-down list to specify the type of comparison you want to use to evaluate the field:</p> <ul style="list-style-type: none"> • Contains: Ignores the field if it contains blank data. • Equal: Scores the field as 0 if it contains blank data. • Greater Than: Scores the field as 100 if it contains blank data. • Greater Than Or Equal To: Scores the suspect and candidate fields as 100 if they both contain blank data; otherwise, scores the suspect and candidate fields as 0. • Highest: Compares the field's value for all the records group and determines which record has the highest value in the field. <p>For example, if the fields in the group contain 10, 20, 30, or 100, the record with the field value 100 would be selected.</p> <p>This operation only works on numeric fields. If multiple records are tied for the longest value, one record is selected.</p> <ul style="list-style-type: none"> • Is Empty: Determines if the field contains no value. • Is Not Empty: Determines if the field contains any value. • Less Than: Determines if the field value is less than the value specified. This operation only works on numeric fields. • Less Than Or Equal To: Determines if the field value is less than or equal to the value specified. This operation only works on numeric fields. • Longest: Compares the field's value for all the records group and determines which record has the longest (in bytes) value in the field. <p>For example, if the group contains the values "Mike" and "Michael", the record with the value "Michael" would be selected. If multiple records are tied for the longest value, one record is selected.</p> <ul style="list-style-type: none"> • Shortest: Compares the field's value for all the records group and determines which record has the shortest (in bytes) value in the field. <p>For example, if the group contains the values "Mike" and "Michael", the record with the value "Mike" would be selected. If multiple records are tied for the shortest value, one record is selected.</p> <ul style="list-style-type: none"> • Lowest: Compares the field's value for all the records group and determines which record has the lowest value in the field. <p>For example, if the fields in the group contain 10, 20, 30, or 100, the record with the field value 10 would be selected. This operation only works on numeric fields. If multiple records are tied for the longest value, one record is selected.</p> <ul style="list-style-type: none"> • Most Common: Determines if the field value contains the value that occurs most frequently in this field among the group's records. If you select this operator, you see the Multiple results option, which allows you to choose one of these: <ul style="list-style-type: none"> a. Consider None: If the field has the same number of occurrences for more than one value, no value is returned as most common. b. Consider All: If the field has the same number of occurrences for more than one value, all most common values are returned. • Not Equal: Determines if the field value is not the same as the value specified.

Option	Description
Value type	<p>Select one of these from the drop-down list to specify the value you want to compare to the field's value:</p> <p>Note: This option is not available if you select one of these Operators - Highest, Lowest, Longest, Shortest, Is Empty, Is Not Empty, or Most Common.</p> <ul style="list-style-type: none"> • Field: Select this option if you want to compare another dataflow field's value to the field. • String: Select this option if you want to compare the field to a specific value.
Value	<p>It specifies the value to compare to the field's value.</p> <ul style="list-style-type: none"> • If you selected Field in the Value type field, select a dataflow field. • If you selected String in the Value type field, enter the value you want to use in the comparison. <p>Note: This option is not available if you select one of these Operators - Highest, Lowest, Longest, Shortest, Is Empty, Is Not Empty, or Most Common.</p>

4. (Optional) If you want to define additional rules, click the **Add rule**  button.

Note: If you add additional rules, you will have to select a logical operator to use between each rule.

- If you want the new rule and the previous rule to both pass in order for it to be selected as the template record, select **And**.
- If you want either the previous rule or the new rule to pass in order for the record to be selected as the template record, select **Or**.

You have now configured rules to use to select the template record. You must now configure the best of breed settings to complete the configuration.

Defining Best of Breed Settings


The Best of Breed rules and actions work together to determine which fields from duplicate records in a collection to copy to the Best of Breed record. Rules test values in a record, and if the record passes the rules, the data is copied from the record to the template record. Actions define which data to copy, and which field in the template record should receive the data. After all the rules and actions are run, the template record will be the best of breed record.

Note: In **Best of Breed Settings**, **Rules** and **Actions** are grouped together into conditions, and you can have multiple conditions.

To define Best of Breed rules and actions, go to **Best of Breed Settings** in the **Rule Configuration** panel and perform these steps:

1. Click the **Add condition**  button.

A condition node is displayed to define the rules and actions.

2. Click the **Add rule**  button.
3. Define these options as listed in the table below, which are displayed on the **Rules** node:

Option	Description
Field name	It specifies the name of the dataflow field whose value you want to evaluate to determine if the condition is met and the associated actions should be taken.
Field type	Select one of these from the drop-down list to specify the type of data in the field: <ul style="list-style-type: none">• Non-Numeric: Select this option if the field contains non-numeric data, for example, string data.• Numeric: Choose this option if the field contains numeric data, for example, double or float.

Option	Description
--------	-------------

Operator	
-----------------	--


Option	Description
	<p>Select one of these from the drop-down list to specify the type of comparison you want to use to evaluate the field:</p> <ul style="list-style-type: none"> • Contains: Determines if the field contains the value specified, for example, "sailboat" contains the value "boat". • Equal: Determines if the field contains the exact value specified. • Greater Than: Determines if the field value is greater than the value specified. This operation only works on numeric fields. • Greater Than Or Equal To: Determines if the field value is greater than or equal to the value specified. <p>Note: This operation only works on numeric fields.</p> <ul style="list-style-type: none"> • Highest: Compares the field's value for all the records group and determines which record has the highest value in the field. <p>For example, if the fields in the group contain 10, 20, 30, or 100, the record with the field value 100 would be selected.</p> <p>This operation only works on numeric fields. If multiple records are tied for the longest value, one record is selected.</p> <ul style="list-style-type: none"> • Is Empty: Determines if the field contains no value. • Is Not Empty: Determines if the field contains any value. • Less Than: Determines if the field value is less than the value specified. This operation only works on numeric fields. • Less Than Or Equal To: Determines if the field value is less than or equal to the value specified. This operation only works on numeric fields. • Longest: Compares the field's value for all the records group and determines which record has the longest (in bytes) value in the field. <p>For example, if the group contains the values "Mike" and "Michael", the record with the value "Michael" would be selected. If multiple records are tied for the longest value, one record is selected.</p> <ul style="list-style-type: none"> • Shortest: Compares the field's value for all the records group and determines which record has the shortest (in bytes) value in the field. <p>For example, if the group contains the values "Mike" and "Michael", the record with the value "Mike" would be selected. If multiple records are tied for the shortest value, one record is selected.</p> <ul style="list-style-type: none"> • Lowest: Compares the field's value for all the records group and determines which record has the lowest value in the field. <p>For example, if the fields in the group contain 10, 20, 30, or 100, the record with the field value 10 would be selected. This operation only works on numeric fields. If multiple records are tied for the longest value, one record is selected.</p> <ul style="list-style-type: none"> • Most Common: Determines if the field value contains the value that occurs most frequently in this field among the group's records. If you select this operator, you see the Multiple results option, which allows you to choose one of these: <ul style="list-style-type: none"> a. Consider None: If the field has the same number of occurrences for more than one value, no value is returned as most common. b. Consider All: If the field has the same number of occurrences for more than

Option	Description
	<p>one value, all most common values are returned.</p> <ul style="list-style-type: none"> • Not Equal: Determines if the field value is not the same as the value specified.
Value type	<p>Select one of these from the drop-down list to specify the value you want to compare to the field's value:</p> <p>Note: This option is not available if you select one of these Operators - Highest, Lowest, Longest, Shortest, Is Empty, Is Not Empty, or Most Common.</p> <ul style="list-style-type: none"> • Field: Select this option if you want to compare another dataflow field's value to the field. • String: Select this option if you want to compare the field to a specific value.
Value	<p>It specifies the value to compare to the field's value.</p> <ul style="list-style-type: none"> • If you selected Field in the Value type field, select a dataflow field. • If you selected String in the Value type field, enter the value you want to use in the comparison. <p>Note: This option is not available if you select one of these Operators - Highest, Lowest, Longest, Shortest, Is Empty, Is Not Empty, or Most Common.</p>


4. (Optional) If you want to define additional rules, click the **Add rule**  button.

Note: If you add additional rules, you will have to select a logical operator to use between each rule.

- If you want the new rule and the previous rule to both pass in order for the condition to be met and the associated actions taken, select **And**.
- If you want either the previous rule or the new rule to pass in order for the condition to be met. select **Or**.

5. Click the **Add action**  button.
6. Define these options as listed in the table below, which are displayed on the **Actions** node:

Option	Description
Source type	<p>Select one of these from the drop-down list to specify the type of data to copy to the Best of Breed record:</p> <ul style="list-style-type: none"> • Field: Select this option if you want to copy a value from a field to the Best of Breed record. • String: Select this option if you want to copy a constant value to the Best of Breed record.
Source data	<p>It specifies the data to copy to the best of breed record.</p> <ul style="list-style-type: none"> • If the Source type is Field, select the field whose value you want to copy to the destination field. • If the Source type is String, specify a constant value to copy to the destination field.
Destination	<p>It specifies the field in the Best of Breed record to which you want to copy the data defined in the Source data field.</p>
Accumulate source data	<p>If the data in the Source data field is numeric, you can enable this option to combine the source data for all duplicate records and put the total value in the Best of Breed record.</p> <p>For example, if there were three duplicate records in the group and they contained these values in the Deposits field:</p> <p>100.00 20.00 5.00</p> <p>Then all three values would be combined, and the total value, 125.00, would be put in the Best of Breed record's Deposits field.</p>

7. (Optional) If you want to define additional actions for the same condition, repeat steps 5 and 6.
8. (Optional) If you want to add another condition, click the **Add condition**  button.

Example for Best of Breed Rule and Action

This Best of Breed rule selects the record where the Match Score is equal to the value of 100. The Account Number data that corresponds to the selected field is then copied to the AccountNumber field on the Best of Breed record.

Rule

Field name: MatchScore

Field type: Numeric

Operator: Equal

Value type: String

Value: 100

Action Source type: Field Source data: AccountNumber Destination: AccountNumber
--

Saving and publishing the rule

After configuring the options for the Best of Breed rule, you can save or publish the rule according to your requirements.

- To save the rule, click the **Save** button.

The rule is saved as a **draft** and is displayed on the **Source Details** page and the **Home page**, which indicates that the rule is in a preliminary state and does not exist in the repository.

- To publish the rule in the repository, click the **Publish** button. Make sure that you enter a unique name while publishing a rule.

The rule is published to the repository and is displayed on the **Source Details** page and the **Home page**.

Note: In case you make changes to a published rule, you cannot save it as a draft copy. You will have to be re-publish the rule to save the changes.

For more information, see *Viewing published and draft rules on source details page* in **Creating Rules** on page 156.

Creating Smart Consolidation Rule

A smart consolidation rule is prepared with a process that combines machine learning and human expertise to accelerate and automate the process of record merging. It works in tandem with the smart match process, so you have an option to use the matching rules to resolve the duplicates to get a Best of Breed record, or you can use any other file source as input after resolving the duplicates.

To create a smart consolidation rule, perform these steps:


1. On the **Source Details** page, click the **Create consolidation rule** button.
The **Create Consolidation Rule** page is displayed.
2. On the **Create Consolidation Rule** page, go to the **Smart Consolidation Process** section and click the **Create Rule** button.
3. On the top-left corner of the page, enter a name for your rule in the **BOB rule name** field, and proceed with the flow as shown below:



Selecting Source

This page allows you to select a source file that contains the group of records, such as the collection number, where the records with the same collection number form a group. On this page, you can either select the same source by toggling the **Same as source** button to **Yes** or choose to select a different source file, as mentioned below.


Note: It is recommended to select a file with duplicate records and ensure that the unique records are filtered out. You can also use the output of your matching as an input source here.

1. Click the **File path** button.
The **Choose File** pop-up window is displayed.
2. Use the file path selection box  / [Server](#) / [E:](#) / [Sample Files](#) to go to the required path on the server.
A list of files at the selected location is displayed below the file path selection box.
3. Select the file you need, and click **OK**.

Note: Use the **Filter** to go to specific file in the list.

4. To modify settings for the file, make changes as described in this table.

Settings	Description
Character encoding	The text file's encoding. Select one of these:
UTF-8	Supports all Unicode characters and is backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html .
UTF-16	Supports all Unicode characters but is not backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html .
US-ASCII	A character encoding based on the order of the English alphabet.
UTF-16BE	UTF-16 encoding with big endian byte serialization (most significant byte first).
UTF-16LE	UTF-16 encoding with little endian byte serialization (least significant byte first).
ISO-8859-1	An ASCII character encoding typically used for Western European languages. Also known as Latin-1.
ISO-8859-2	An ASCII character encoding typically used for Eastern European languages. Also known as Latin-2.
ISO-8859-3	An ASCII character encoding typically used for Southern European languages. Also known as Latin-3.
ISO-8859-9	An ASCII character encoding typically used for Turkish language. Also known as Latin-5.
CP850	An ASCII code page used to write Western European languages.
CP500	An EBCDIC code page used to write Western European languages.
Shift_JIS	A character encoding for the Japanese language.
MS932	A Microsoft's extension of Shift_JIS to include NEC special characters, NEC selection of IBM extensions, and IBM extensions.

Settings	Description						
Field Delimiter	<p>Specifies the character used to separate fields in a delimited file.</p> <p>For example, this record uses a pipe () as a field delimiter:</p> <pre>7200 13TH ST MIAMI FL 33144</pre> <p>The characters available as field delimiter are:</p> <ul style="list-style-type: none"> • Comma • Semicolon • Pipe • Tab • Space • Period <p>You can also add custom field delimiters. To add a custom field delimiter, follow these steps:</p> <ol style="list-style-type: none"> Click the Add button  placed adjacent to Field delimiter. Add Delimiter pop-up window is displayed. Enter your desired field delimiter in the Character field, the corresponding Unicode is displayed automatically. Enter a suitable name for your delimiter in the Name field. Click Save, your delimiter appears in the Field delimiter drop down. 						
Text qualifier	<p>The character used to surround text values in a delimited file.</p> <p>For example, this record uses double quotes (") as a text qualifier.</p> <pre>"7200 13TH ST" "MIAMI" "FL" "33144"</pre> <p>The characters available to define as text qualifiers are:</p> <ul style="list-style-type: none"> • Single Quotes (') • Double Quotes (") 						
Line separator	<p>Specifies the character used to separate records in line in a sequential or delimited file.</p> <p>The record separator settings available are:</p> <table> <tr> <td>Unix (U+000A)</td><td>A line feed character separates the records. This is the standard record separator for Unix systems.</td></tr> <tr> <td>Macintosh (U+000D)</td><td>A carriage return character separates the records. This is the standard record separator for Macintosh systems.</td></tr> <tr> <td>Windows (U+000D U+000A)</td><td>A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.</td></tr> </table>	Unix (U+000A)	A line feed character separates the records. This is the standard record separator for Unix systems.	Macintosh (U+000D)	A carriage return character separates the records. This is the standard record separator for Macintosh systems.	Windows (U+000D U+000A)	A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.
Unix (U+000A)	A line feed character separates the records. This is the standard record separator for Unix systems.						
Macintosh (U+000D)	A carriage return character separates the records. This is the standard record separator for Macintosh systems.						
Windows (U+000D U+000A)	A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.						

Settings	Description
First row is header row	<p>Specifies if the first record in a delimited file contains header information. A Yes indicates it has header information.</p> <p>For example, this file snippet shows a header row in the first record.</p> <pre>"AddressLine1" "City" "StateProvince" "PostalCode" "7200 13TH ST" "MIAMI" "FL" "33144" "One Global View" "Troy" "NY" "12180"</pre>

- Click **Save and Continue** to save your changes.
You are now ready to select columns for your rule.

Selecting Columns

On this page, columns of your data are displayed in a tabular format. You must select the columns from your data that must participate in the model training and best of breed rule creation and select the fields for which you want the data to be merged.

- Select the **Column Name** check-box for the columns that must participate in the best of breed rule creation.
- If you want to merge a field, toggle **Merge** to **YES**.

Note: Make sure that you always select the corresponding check-box of the field you wish to merge.

- Select a desired semantic type from the drop-down after selecting the corresponding check-box of that column. By default, **NONE** is displayed.

Based on the selected columns, groups of records are automatically generated, and these are displayed on the next page for tagging. The generated groups cover all the variations in the data using advanced and smart algorithms and techniques. For example, say if your original input file contained 5000 groups, the system might show only 50 groups covering all the variations.

Note: By default, the maximum collection size limit is 10, and the groups larger than that are excluded from the consolidation process.

The purpose of generating variations is to identify a small subset of collections for tagging, which covers most of the unique variations in source data. It's like picking up few collections from a large set of collections representing the complete set so that tagging on this subset will provide the best of breed rule close to one we would have got by tagging the entire collection set.

The variations are generated based on operations that we have in the **Best Of Breed** stage.

BOB Operator	Based on Feature
Most Common	Frequency
Longest/Shortest	Length
Highest/Lowest	Rank
Greater/Less Than	Absolute values
Equals/Not Equals	It is based on finding the values which are category-specific and using the obtained values as a feature.
Empty/Not Empty	Frequency

Note: By default, the field **Collection number**, which is a mandatory field, is auto-selected and disabled. The collection number identifies each duplicate record in a match queue, and if the candidate is a duplicate, it is assigned a collection number.

Tagging Records

This page lets you review the groups and taggings done by the system. You need to select the fields from the groups generated by the system to propose a master record. The fields identified as the master record within a group are collectively displayed on the first row by the system and highlighted within the table. The system then generates the **Best of Breed Rule** on the next page based on your feedback.

Note: The pre-tags are suggestive only based on the internal algorithms and should be reviewed thoroughly.

In case you wish to select any other field for a given group, place your cursor over the required field and click on it. The selected field is then dynamically updated in the first row of the table. You can also choose to select the entire row as a master record by clicking the **Move to Master** icon, which appears on the row's extreme left-side.

Note: Each column's header is re-sizeable and can be dragged-dropped to any place based on your needs.

Click the **Save and Continue** button to see the best of breed rule generated by the system.

CustomerMasterRecord

Cancel < Previous Save and Continue >

Select Source Select Columns Tag Records Analyze Results

Select the fields from below groups to update the master record. (Total groups: 9)

Save Tagging Reset Tagging

	GivenName	Surname	EmailAddress	Datasource
▼	Darren	keller	elewis@gmail.com	
⋈ Promote as Master		Kellerson	elewisNew@gmail.com	hr
	derran	keller	elewis@gmail.com	web
	Darren	kelle	timothyjohnson@yahoo.com	

Save and Reset Tagged Records

- To save the tags specified by you, click the **Save Tagging** button on the page's top right just above the table. Your specifications will remain saved across multiple sessions.
- To reset the tagging done so far, click the **Reset Tagging** button.

Analyzing and Publishing Results

The **Analyze Results** page displays the generated consolidation rule from the information provided by you. The rule can be reviewed and published to the repository, which can be further consumed in your batch jobs.

Note: The consolidation rule displayed on this page is non-editable. It is intended for viewable purposes only.

Publishing the rule

On the top-right corner of the page, click the **Publish** button to publish the rule, which can then be used while creating jobs in the flow designer. The rule is published to the repository and is displayed on the **Source Details** page and the **Home page**.

For more information, see *Viewing published and draft rules on source details page* in **Creating Rules** on page 156.

Note: If you do not publish the rule, it is shown in a draft state on the dashboard, which indicates that the rule is in a preliminary state and does not exist in the repository.

How to Video - Prepare Dashboard

This video explains how you can use the Prepare dashboard to build DQ Rules and tag them to business entities.

<https://www.youtube.com/embed/R6i-kb3FF70>

How to Video - Preparing Manual Matching and Consolidation Rule

This video explains how you can build manual matching and consolidation rules.

<https://www.youtube.com/embed/kRN1sCTbNTE>

How to Video - Preparing Smart Consolidation Rule

This video explains how you can build smart consolidation rule.

<https://www.youtube.com/embed/IGDg5DpzTaw>

How to Video - Preparing Smart Match Rule

This video explains how you can build smart match rule.

<https://www.youtube.com/embed/tJmMFkB5cnA>

8 - Profiling and Scorecarding

Data Profiling

Successful decision making is heavily dependent on reliable, correct, complete, and valid data. Incomplete records, malformed fields, and a lack of context can result in misleading or inaccurate data being delivered to your business users, which can result in flawed decisions.

Data profiling scans your data records from all the data sources - irrespective of its volume and complexity. It identifies problems related to correctness, completeness, and validity in the data, and suggests actions to fix the issues. Thus, it improves the quality and utility of your data with very little manual effort.

Benefits of data profiling

- It is the first step in analyzing your data and predicting how much effort is needed to make it usable
- It improves your trust on the data set you have
- It is one of the mandatory steps for taking control of your organizational data and using it across the enterprise.

Scorecard

Monitoring is one of the indispensable aspects of data governance and good decision making. Constant monitoring of data quality helps you to take measures to improve the consistency, reliability, and accuracy of your data. Scorecards help you measure the quality of your data and assign score cards to it by tracking key matrices defined by you. It represents your data health in a graphical form, making assessment all the more easy and swift for you.

Trends

Trends help you measure the improvement in the quality of your data over a period of time. For example, in a merger between two companies (Company A and Company B), company A needs to migrate customer data in its system from company B. It will run a scorecard on the data at the beginning of the process to see the quality and may thereafter do this scoring after every cleansing. Trends plot a chart based on your recent runs or for a specified date range to give you a clearer picture on the improvement index.

Note: You need to have the **Data Integration** module installed in order to use the profiling feature of **Spectrum Discovery**.

In this section

Profile & Monitor Home page.....	194
Generating Score cards.....	195
Editing a scorecard.....	208
Scheduling Scorecard Analysis.....	209
Viewing scorecard trends.....	211
Creating a Profile.....	211
Specifying Profiling Default Settings.....	226
Analyzing a Profile.....	227
Viewing Data Profiling Results.....	232
Collaborating on Data Profiling Results.....	239
How to Video - Profiling.....	240
How to Video - Applying Suggestions to Profiled Data.....	240
How to Video - Scorecard.....	240
How to Video - Consuming Spectrum Services in Profiling and Scorecard.....	240

Profile & Monitor Home page

To access the **Profile & Monitor** Home page, click **Profile & Monitor** on the **Spectrum Discovery** main menu.

Profile & Monitor Defaults Start

Profiles Scorecards Schedules History

+ 📄 ✎ 🗑 🔍 ⏮ ⏭ ⏹ 🖨

<input type="checkbox"/>	Name	Data Source	User	Last Run Start Time	Last Run End Time	Duration	Status	Tables	Records	Suggestions	Reports
<input type="checkbox"/>	party-Profilenew	party-Profilenew	admin	21/09/2020 1:18 PM	21/09/2020 1:18 PM	00:10	✓	1	100		
<input type="checkbox"/>	branch-Profile	branch-Profile	admin	18/09/2020 12:43 PM	18/09/2020 12:43 PM	00:08	✓	1	13		
<input type="checkbox"/>	GAV_Input_Custom-Profile	GAV_Input_Custom-Profile	admin	17/09/2020 7:07 PM	17/09/2020 7:07 PM	00:08	✓	1	670		

Profiles

Use this tab to create a profile from various data sources, models, and connections, configure profiling rules, and analyze the profiling results.

Scorecards

Use this tab to create a scorecard from various data sources, models and connections, configure parameters and rules, and generate a preview of the score card.

Schedules

Use this tab to create a schedule of profile and score card analysis. The schedule can be a combinations of pattern, time, and date range, as suited for your requirement.

History

Use this tab to view the history of profile and score card runs.

Generating Score cards

Score cards are a graphical representation of your data health. It helps you measure and keep track of the improvement in quality of your data.

To create a score card, you need to:

1. Add data sources
2. Define the data quality metric to be used
3. Build rules for each of the data quality metric
4. Define the threshold to indicate if the data needs attention, is marginal, or acceptable
5. Save the scorecard (The **Save** button also runs the score card.)

Spectrum Discovery gives you the capability of creating and assigning score cards to your data on parameters, such as accuracy, consistency, and completeness. It also gives you the flexibility of defining your own parameters and accordingly script rules.

The scores are assigned based on the rules you configure and threshold limits you set for the data.

The three stages in creating a score card are:




Adding data sources


Selection of data source is the first step of creating a score card. Your sources can be one of these.

- Flat files from your system or server
- Logical or physical models created using the modeling feature of Spectrum Discovery
- Connections configured in Spectrum Technology Platform

Adding Data from Flat Files

These steps describe the process of creating a scorecard from records in flat files on your machine or on server.

1. On the **Spectrum Discovery** Home page, click **Profile**.
2. Click the  icon on the icon bar just below the tabs.
3. On the page that is displayed, select the **Flat File** option from the **Select data source** field.
4. Click the **Add Files** button.
The **Choose File** pop-up window is displayed.

5. Use the file path selection box  / [Server](#) / [E:](#) / [Sample Files](#) to go to the required path on your local machine or server.

A list of files at the selected location is displayed below the file path selection box.

6. Select all the files you need, and click **OK**.

Note: Use the **Filter** to go to specific files in the list.

The selected files are displayed on the left side of the page, below the **Add Files** button.

7. To add more files, click the **Add Files** button again and repeat steps 6 and 7.


Note: You can add files from different folders on your local system, on server, and FTP server in any combination you want. For using FTP server, you need to configure it first. For more information, see the Administration Guide.

8. Click any file in the list to view these details below the **Add Files** button.:

- The number of files selected. For example *7 File(s) selected*
- Color coding for file's metadata
 - **Green:** Metadata detected automatically
 - **Blue:** User configured metadata
 - **Red:** Metadata not detected. Such files are skipped during profiling and scorecarding. However, you can configure the metadata of these files using the **File Setting** fields as described in this task below. After manual configuration, these files turn blue.
- Path of the file. For example, `E:/Sample Files/profile/fileOne.txt`
- File settings
- Preview of the records in the file

9. To modify settings for any of the files, select it and make changes as described in this table.


Settings	Description
Character encoding	<p>The text file's encoding. Select one of these:</p> <p>CP1252 This encoding is also known as the Windows-1252 or simply Windows character set. It is a super set of ISO-8859-1 and uses the 128-159 code range to display additional characters not included in the ISO-8859-1 character set.</p> <p>UTF-8 Supports all Unicode characters and is backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html.</p> <p>UTF-16 Supports all Unicode characters but is not backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html.</p> <p>US-ASCII A character encoding based on the order of the English alphabet.</p> <p>UTF-16BE UTF-16 encoding with big endian byte serialization (most significant byte first).</p> <p>UTF-16LE UTF-16 encoding with little endian byte serialization (least significant byte first).</p> <p>ISO-8859-1 An ASCII character encoding typically used for Western European languages. Also known as Latin-1.</p> <p>ISO-8859-3 An ASCII character encoding typically used for Southern European languages. Also known as Latin-3.</p> <p>ISO-8859-9 An ASCII character encoding typically used for Turkish language. Also known as Latin-5.</p> <p>CP850 An ASCII code page used to write Western European languages.</p> <p>CP500 An EBCDIC code page used to write Western European languages.</p> <p>Shift_JIS A character encoding for the Japanese language.</p> <p>MS932 A Microsoft's extension of Shift_JIS to include NEC special characters, NEC selection of IBM extensions, and IBM extensions.</p> <p>CP1047 An EBCDIC code page with the full Latin-1 character set.</p>

Settings	Description						
Field delimiter	<p>Specifies the character used to separate fields in a delimited file.</p> <p>For example, this record uses a pipe () as a field delimiter:</p> <pre>7200 13TH ST MIAMI FL 33144</pre> <p>The characters available as field delimiter are:</p> <ul style="list-style-type: none"> • Comma • Semicolon • Pipe • Tab • Space • Period <p>You can also add custom field delimiters, to add a custom field delimiter follow these steps:</p> <ol style="list-style-type: none"> Click the Add button  placed adjacent to Field delimiter. Add Separator pop-up window is displayed. Enter your desired field delimiter in the character field, the corresponding Unicode is displayed automatically. Enter a suitable name for your delimiter in the Description field. Click Save, your delimiter appears in the Field delimiter drop down. 						
Text qualifier	<p>The character used to surround text values in a delimited file.</p> <p>For example, this record uses double quotes (") as a text qualifier.</p> <pre>"7200 13TH ST" "MIAMI" "FL" "33144"</pre> <p>The characters available to define as text qualifiers are:</p> <ul style="list-style-type: none"> • Single quote (') • Double quote (") 						
Line separator	<p>Specifies the character used to separate records in line in a sequential or delimited file.</p> <p>The record separator settings available are:</p> <table> <tr> <td>Unix (U+000A)</td><td>A line feed character separates the records. This is the standard record separator for Unix systems.</td></tr> <tr> <td>Macintosh (U+000D)</td><td>A carriage return character separates the records. This is the standard record separator for Macintosh systems.</td></tr> <tr> <td>Windows (U+000D U+000A)</td><td>A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.</td></tr> </table>	Unix (U+000A)	A line feed character separates the records. This is the standard record separator for Unix systems.	Macintosh (U+000D)	A carriage return character separates the records. This is the standard record separator for Macintosh systems.	Windows (U+000D U+000A)	A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.
Unix (U+000A)	A line feed character separates the records. This is the standard record separator for Unix systems.						
Macintosh (U+000D)	A carriage return character separates the records. This is the standard record separator for Macintosh systems.						
Windows (U+000D U+000A)	A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.						

Settings	Description
First row is header row	<p>Specifies if the first record in a delimited file contains header information. A Yes indicates it has header information.</p> <p>For example, this file snippet shows a header row in the first record.</p> <pre>"AddressLine1" "City" "StateProvince" "PostalCode" "7200 13TH ST" "MIAMI" "FL" "33144" "One Global View" "Troy" "NY" 12180</pre>


10. To apply settings of one file to another:
 - a. Select the file from which you want to copy the settings, and click the **Copy File Settings** button.
 - b. In the **Apply file settings** pop-up window that is displayed, select the files to which you want to apply the settings, and click **OK**.

Note: This option is applicable only in cases where you have selected multiple files.

11. To see preview of the selected file, see the **Preview** section below **File Settings**.
12. To go to the configuration stage, click  on the top right corner of the page.

Adding Data from Models

To add data from physical or logical models, perform these steps:

1. On the **Spectrum Discovery** Home page, click **Profile**.
2. Click the  icon on the icon bar just below the tabs.
3. From the **Select data source** options, select the **Model** option.
4. From the **Select model** drop-down list, select the physical or logical model, as needed. To narrow down your search by model type (physical or logical) or model name, type it in the field and the related models start displaying below it. For example, if you type "logical", all the logical models are displayed.

Note: To create a new model or to modify an existing one, click on the **Manage models** link. It navigates you to the **Modeling** page, where you can create and save a new physical or logical model. This model can then be selected here using the **Select model** drop-down list.

The tables and columns in the selected model are displayed below the field.


5. From the list of tables, select tables that you want to include. In each of the tables, you can also select the columns you require to be added. To include all the tables and columns, select the check box in the header row.

Note: Use the type-ahead search text box to search for specific tables or columns.


6. To go to the configuration stage, click  on the top right corner of the page.

Adding Data from a Connection

To add data from the any of the database connections configured in Spectrum Technology Platform, perform these steps:

1. On the **Spectrum Discovery** Home page, click **Profile**.
2. Click the  icon on the icon bar just below the tabs.
3. From the **Select data source** options, select the **Connection** option.
4. From the **Select connection** drop-down list, select the connection you want to use to create the new profile. To narrow down your search by connection type or its name, type it in the field and the related connections display below it. For example, if you type `mssql` then all the MSSQL server connections display.

Note: The drop-down list displays all the connections that are configured in Spectrum Technology Platform. To configure a new connection, use the **Add Data Sources** page, which you can access through the **Connection** tab in Spectrum Discovery or with the Management Console. For steps to configure a new data source, see the section [#unique_240](#).

5. Click the **Select tables** button.
The **Select tables** pop-up window is displayed showing all the tables in the selected connection.
6. From the list of tables in the **Select tables** pop-up window, select the tables that you want to include, and click **OK**.
The selected tables are displayed below the **Select tables** button.
7. To select columns from a table, click , and select the columns you require. To include all the columns, select the check box in the table header.

Note: Use the type-ahead search text box to search for specific tables and columns.

8. To go to the configuration stage, click  on the top right corner of the page.

Configuring Service

Use the **Service Configuration** feature to consume the Spectrum services in building rules for score carding. You can use both the default Spectrum services as well as the services you have configured using the different Spectrum stages you are licensed for. This configuration helps you reuse the transformations designed with various Spectrum stages on your data.


Note: This is an optional feature.


1. On the **Service Configuration** stage of the **Create Scorecard** page, use the **Services** drop down list to select the required service.
2. The **Service input fields** displays the list of the fields used as input in the selected service. Use the corresponding **Source input fields** drop down to select field from your input data.

Note: The data types of input fields and source fields with which those are mapped should be the same.

3. **Service output fields** Shows the output fields in the selected service. Select the fields that you want in your output. You can use the type-ahead search box (above the column) to find out the required fields.


Note: Data in the selected field will be used as input when you configure the rules. The field name will have this format: *<Service Name> ~ <Column Output Field>*.

4. Click the forward arrow  on the top right of the page to move to the next stage.

Note: At any point, to go back and edit details in the previous stage, click the backward arrow  on the top right of the page.

Configuring scorecard

After selecting the source files of data to be scored, you need to configure the score card. Configuration involves:

- Adding data quality dimensions on the basis of which scoring is to be performed
 - Adding and configuring rules for every DQ metric
 - Defining the threshold percentages for categorizing the data as bad, acceptable, or good
1. On the **Create Scorecard** page, from the **Entities** list, select the entity for which you want to define the DQ dimensions and perform one of these steps, based on whether you want to select a DQ metric from the list or define a new one.
 - a. Click the **Add DQ Metric**  icon, and select the required parameter from *Accuracy*, *Consistency*, and *Completeness*.
 - b. To define a new DQ metric, click **Add DQ Metric**, write your parameter where the cursor is blinking, and press the **Enter** button on your system.

The selected DQ metric is displayed as a tab adjacent to the **Add DQ Metric** tab.

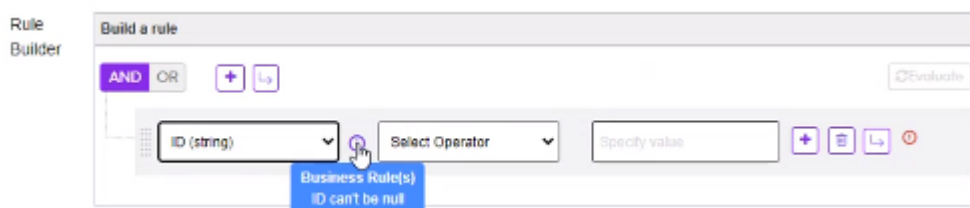
2. Click the **Add rule** button to create a rule for the DQ metric.
The **Rule Configuration** page is displayed.

Note: On the **Rule Configuration** page, you can view the profile statistics for your entity by clicking on the **PROFILE STATISTICS** palette on the right side of the page. For more information, see [Viewing Profile Statistics](#) on page 203.

3. Enter the **Name** of the rule.
4. Give a **Description** to the rule.
5. In the **Rule Builder**, build your rule by using the **Select Column** and **Select Operator** drop down lists.
 - a. The **Select Column** drop-down lists all the columns in the selected data source along with the data type. For example, PartyID (string).

Note: For models built upon the Business Glossary entities, the properties of the entities are displayed here. For example, in case of a *Party* Business Glossary entity, the property can be Party ID (string).

Note: If you have specified a business rule for the properties of the business glossary entities, it will be displayed here as suggestion. See the example of **ID (string)** property in the image below.





- b. The **Select Operator** drop-down lists the operators. These are the operators that display based on the data type of the column you selected: exists in, equal, greater than or equal, equal, less than, is between, ends with, contains, is not null, not equal, less than or equal, is null, greater than, is not empty, starts with, matches, is null, is empty, matches semantic type, Does not contain, Does not end with, and Does not start with.
 - To make use of the custom semantic types defined in the **Glossary**, use the matches semantic type operator. When you select it, the custom semantic types defined for the selected data type are displayed in the adjoining drop-down list. For example, if you have defined BusinessParty and PassportNumber as two custom semantic types for the string data type, these two options will be shown to you when you select PartyID (string) as the column and matches semantic type as the operator.
 - Use the Is between operator to fetch the values lying in between two primitive integer and primitive date type data.
 - c. Use the text box after **Select Operator** to enter values for the expression.

Note: Avoid using anchor operators, such as ^ (beginning of line) and \$ (end of line) in the regular expression, as these are not supported by Lucene's regular expression engine.


- d. To build multiple conditions, click the **Condition** button, and use the **AND OR** slider to join the multiple conditions.
- e. To create another group of conditions, click the **Group** button.
- f. Drag the conditions set up and down and across the groups to build the needed expression.


The expression you built is displayed as **Rule Expression**.

6. Click **OK**.
You are navigated back to the **Create Scorecard** page, with the rule details displayed in a grid below the **Add Rule** button.
7. To edit rule details, click the **Edit rule**  icon in the **Actions** column.
8. To delete the rule, click the **Delete rule**  icon in the **Actions** column.
9. Adjust the **Threshold** slider to define your score card as **Needs Attention**, **Marginal**, or **Acceptable**.

Note: Color coding for the slider:

- Needs Attention: Red
- Marginal: Yellow
- Acceptable: Green

10. Click the forward arrow  on the top right of the page to move to the next stage in creating scorecard.

Note: At any point, to go back and edit details in the previous stage, click the backward arrow  on the top right of the page.

Viewing Profile Statistics

When you build a rule to configure the scorecard, the **Profile Statistics** panel extracts and displays the corresponding entity's column statistics. You can analyze the statistics for an entity's column rule and make decisions to build a better rule for your entity. The statistics are auto-refreshed based on your column selection in the **Rule Builder**.

Note: The statistics displayed on the panel are non-editable. It is intended for viewable purposes only.

The **Profile Statistics** panel allows you to:

1. Select a column name from the **Column** drop-down to view the statistics of any particular entity's column.

Note: The profile statistics for **Service Configuration Output Fields** will not be displayed.

2. View the detailed profiling statistics of an entity by clicking on **Details** adjacent to the **Column** drop-down. If you click on **Details**, you are navigated to the **Data Profiling Results** page with the details of the corresponding column. For more information, see [Viewing Data Profiling Results](#) on page 232.

Note: The **Profile Statistics** panel is available only if you had run the profile against the selected entity. In case your selected entity has never been profiled, you must run the profile against the entity at least once to make the **Profile Statistics** panel visible, or configure and run the scorecard once to make the statistics available during the rule update, but in this case, you will not see the **Details** link to view the detailed profiling statistics.




The **Profile Statistics** panel displays these comprehensive profiling statistics of an entity generated at the column level:

1. **Completeness:** Displays the completeness of records in the column. The percentage of **Complete**, **Null**, and **Empty String** detected in the column are displayed in the legend.
2. **Uniqueness:** Displays the uniqueness of the data contained in the column by showing the percentage of these statistics in the legend:
 - **Unique:** Records with no duplicates in the data source.

- **NonUnique:** Records having duplicates in the data source.
 - **Distinct:** A list of all records present in your data source irrespective of those being unique or non-unique record.
3. **Frequency:** Displays the frequency of the data contained in the column.
Note: It is displayed only if the column data type is string.
 4. **Patterns:** Displays the pattern of the data contained in the column.
Note: It is displayed only if the column data type is string.

Accessing profile statistics

You can access **Profile Statistics** by these two ways:

1. On the **Profile & Monitor** page:
 - a. Go to the **Profiles** tab and select a profile.
 - b. Click the **Create Scorecard**  button below the **Profiles** tab.
 The **Create Scorecard** page is displayed with a list of entities associated with the profile you select.
Note: You do not have to reselect the source as your profile already has the selected source.
 - c. Configure the scorecard for your entity. For more information, see [Configuring scorecard](#) on page 201.
 You see the **Profile Statistics** panel on the **Rule Configuration** page to analyze the statistics for your entity's columns.
2. On the **Profile & Monitor** page:
 - a. Go to the **Scorecards** tab.
 The **Create Scorecard** page is displayed.
 - b. Add the data source. For more information, see [Adding data sources](#) on page 195.
 - c. Configure the scorecard for your entity. For more information, see [Configuring scorecard](#) on page 201.
Note: The **Profile Statistics** panel will be visible only if you run and evaluate the scorecard against the entity at least once while modifying any rules. In this case, you will not see the **Details** link to view the detailed profiling statistics as the entity has only been evaluated by the scorecard and has not been profiled.

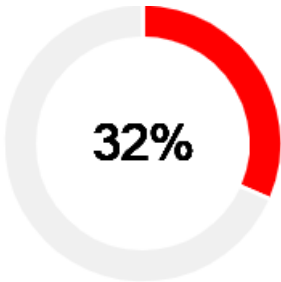
Generating and viewing scorecard

Use the **Create Scorecard** page to generate and view the scorecard for the selected entity, based on the DQ metrics you selected and rules you configured for those metrics.

1. On the **Create Scorecard** page, give a name to this scorecard in the **Scorecard Name** field.

Note: This is a mandatory field.


2. Enter a **Description** for the score card.
3. Click the **Save** button on the top right of the page.
The scorecard is generated for each of the defined DQ metric. It displays these details:

Field	Details
Doughnut chart	<p>A doughnut chart is displayed for each of the defined DQ metric. The chart shows percentage score of the entity based on the rules defined for the DQ metric.</p> 
KPI	<p>Lists all the defined DQ metrics.</p> <p>Note: To edit any of the KPI parameters, click the respective link. You will be navigated to the Edit Scorecard page.</p>

Field	Details
Rules	<p>Displays these details for each of the defined rules:</p> <ol style="list-style-type: none"> Rule name: Displays the name of the rule. <p>Note: Click the hyperlink to go to the Edit Rule page, where you can modify the rule configuration, as required. See Editing a scorecard on page 208 for details.</p> Rule Expression Valid Records: The number of records found valid for the defined DQ metric and rule. Total Records: The total number of records in the data source Score: The percentage score of the records in the selected entity Threshold: The threshold value for the entity, based on the percentage score defined for each category in the Configure Scorecard page. <p>Note: The Current Score is displayed if the scorecard has already been run at least once. The font color of the score matches colors of the Acceptable, Marginal, and Needs Attention as defined.</p>

Creating scorecard from an existing scorecard

You can use any existing scorecard to create a new one.


- Click the **Scorecard** tab on the **Profile & Monitor** page.
The left pane below the icons displays a list of defined score cards. Clicking on any of the score cards displays its details on the right pane of the page.
- Select the score card that you want to copy, and click the **Copy Scorecard**  icon.
The selected score card opens in the copy mode.
- Make the required modifications and give a name to this score card in the **Finish** stage.

Creating scorecard on profiled data

Create scorecard for your profiled data using the steps given in this task. The scorecard thus created allows you to monitor the trend of your profile without having to go to the **Profile Results** page. It also helps you to monitor your profile's performance on KPIs you value or needed for your business case.



Note: You can create scorecard only on those profiles that have been run at least once.

- On **Spectrum Discovery** Home page, click **Profile & Monitor > Profiles**.

2. From the list of profiles, select the profile on which you want to create a scorecard and click the **Create Scorecard**  icon.
The **Create Scorecard** page is displayed.
3. See the current column-level statistics of the selected profile by clicking the **Profile Statistics** ribbon on the right side of the page. This detail is the same as displayed on the **Profile Results** page and it will help you in creating better KPI parameters and rules over the data.
4. Use the **Configure Services (Optional)** and **Configure Scorecard** steps to define relevant KPIs and rules. For information about how to configure the scorecard and see the trend, refer to [Configuring Service](#) on page 200 and [Configuring scorecard](#) on page 201.
5. Generate the scorecard and view trends using the **Trends** page. For more information, see [Generating and viewing scorecard](#) on page 206 and [Viewing scorecard trends](#) on page 211.

Editing a scorecard

To edit details of any of the score cards, perform these steps:


1. Click the **Scorecard** tab on the **Profile & Monitor** page.
The left pane below the icons displays a list of defined score cards. Clicking on any of the score cards displays its details on the right pane of the page.
2. On the left pane, click the score card that you want to edit, and click the **Edit Scorecard**  icon.
The selected score card opens in the edit mode.
3. To edit any of the **KPI** parameters, click the respective **KPI** link.
The **Edit Scorecard** page is displayed that allows you to modify the rules and threshold values for the selected KPI. The **Current Score** is displayed if the scorecard has already been run at least once. The font color of the score matches colors of the **Acceptable**, **Marginal**, and **Needs Attention** as defined.
4. To edit the rule configuration and preview the valid and invalid data for the expression built, click the **Rule name** link.
The **Edit Rule** page is displayed.
 - a. Make the required modifications.
 - b. Click the **Evaluate** button to preview the valid and invalid data for the expression you built.
The preview mentions the expression and lists down the **Valid** and **Invalid** values at the bottom of the page.
 - c. Click the **Valid** tab to view the list of valid data and **Invalid** tab to view the list of invalid data.
 - d. Use the type-ahead **Filter** to narrow down your search for the required record.
 - e. To download the result in a CSV file, click the **Download CSV** icon on the page and save the file in your system at preferred location.
5. Click the forward arrow  to go the **Configure Scorecard** page.

Note: To be able to send mail notifications the mail server needs to be configured. You can do that using this path: **Management Console > System > Mail**.

- a. Click the **Send notification when score threshold changes** check-box, and enter the mail ID of the users to whom you want to send notification for the changed category of the score. For example, if the score crosses the threshold set for the bad score and enters into the good category, a notification will be sent to each of the user IDs you specify.
 - b. To add multiple users, click **+**, and enter the ID in the adjacent text box.
6. Save your modifications.

Scheduling Scorecard Analysis

To schedule a scorecard analysis, use the **Add Schedule** page. This page gives you various combinations of pattern, time, and date range to schedule the profiling in the way that suits your need. You can also have it as a one-time activity.

1. On the **Profile & Monitor** page, click the **Schedules** tab, and click the **Add Schedules** icon . The **Add Schedule** page is displayed.
2. Enter a **Name** for the schedule.
3. Select the **Scorecard** option, and from the drop down list, select the profile that you want to schedule for analysis.

Note: It displays all the defined score cards.

4. In the Schedule section, select one of the triggers as needed from the **Trigger** drop-down list.
 - Select **Date/Time** to schedule a one-time analysis. Use the **Start date** and **Start time** from the respective calendars below the drop-down list.
 - Select the **Recurring Date/Time** option for a recurring analysis.

Note: When you select this option, **Pattern**, **Time** and **Range** sections appear below Trigger. Use these for setting up a recurring analysis, as described below.

5. To set up a recurring analysis, use the one of these options or a combination of all of these:
 - **Pattern** (of days or week or month)
 - **Every weekday:** To run the analysis on every day of the week
 - **Every:** At a specified periodicity. For example, to run analysis every alternate day, choose this option, and enter 2 in the selection box.

- **Day <> of every <> month(s):** To run analysis on a specified day and month. For example, to run analysis on the third day of every alternate month, enter 3 in the **Day** selection box and 2 in the **month(s)** selection box.
- **Every <> month(s) <on a specified day> <of a week>:** For example, to run the analysis every third month on the second Tuesday, enter 3 in the **month(s)** selection box, and from the drop downs, select **on the second** and **Tuesday** options respectively.
- **Days of the week:** Select the days of the week on which you want to run the analysis.

Note: You can select multiple days.

- **Time** To run the analysis at a specified interval of time, select this option and in the selection boxes specify the number of hours or minutes after which you want to run the analysis. For example, every three hours.
- **Range** To specify the range of analysis (at a specified time between a range of dates), select the **Start date** and **End date** from the respective calendars and select the time and period (AM/PM) from the **Start time** option.

Note: Select the **No end date** check box if you do not want to specify an end date for the recurring analysis.

6. To activate this schedule, ensure the **Schedule** slider (on the top right of the page) is **On**.
7. Click the **Save** button.

The schedule is saved and gets displayed on the **Schedules** tab of the **Profile** page. This page lists all the profiles that are scheduled for analysis and displays these details for each of those:



- **Schedule Name:** Name of the schedule
- **Schedule On:** Name of the Profile or Scorecard that is scheduled for analysis
- **Type:** Whether the schedule is for a **Profile** or a **Scorecard**.
- **User:** The user who created the schedule
- **Trigger:** It specifies **SCHEDULE** to indicate that the profile or scorecard is scheduled for analysis
- **Next Run:** Specifies the date, time, and period when the analysis will run again.
- **Last Run:** Specifies the date, time, and period when the analysis was last run.

Note: When analysis begins, the in-progress profile also gets displayed on the **Profiles** tab of the **Profile** page in case of recurring scheduling. If the scheduling is non-recurring, the in-progress profile gets moved from the **Schedules** tab to the **Profiles** tab the moment the processing begins.

Viewing scorecard trends

Trends help you measure the improvement in the quality of your data over a period of time. For example, in a merger between two companies (Company A and Company B), company A needs to migrate customer data in its system from company B. It will run a scorecard on the data at the beginning of the process to see the quality and may thereafter do this scoring after every cleansing. Trends plot a chart based on your recent runs or for a specified date range to give you a clearer picture on the improvement index.

To view the trend:

1. On the **Spectrum Discovery** Home page, click **Profile**.
The **Scorecards** tab is displayed.
2. On the left pane of the page, select the scorecard for which you want to see the trend, and click the **View Trends** icon  on the right side of the page.
3. The **Trends** graph is displayed.
4. You can view the trends on the basis of the `Recent runs` and `Date range`. Select the required option from the **View by** field.
5. Use the arrows on the right of the **View by** field to restrict the number of runs you want to view.
6. Click the **View Dashboard**  button to go back to the Scorecard dashboard.

Creating a Profile

You need to create a profile to run the data quality analysis rules. A profile can include data set from any of these sources:

- Flat files
- Logical or physical models created using the modeling feature of Spectrum Discovery
- Connections configured in Spectrum Technology Platform

In **Spectrum Discovery**, creating a new profile involves these three stages:

1. Adding data sources: Your data source can be any of these:
 - Flat files from your system or server
 - Logical or physical models created using the modeling feature of Spectrum Discovery
 - Connections configured in Spectrum Technology Platform
2. Configuring data analysis rules, defining the sample size, and setting up notification

3. Defining the profile



Adding data sources

Selection of data source is the first step of creating a profile. Your sources can be one of these.

- Flat files from your system or server
- Logical or physical models created using the modeling feature of Spectrum Discovery
- Connections configured in Spectrum Technology Platform



Adding Data from Flat Files

You can create your profile from records in flat files on your machine or on server. To do this, perform these steps:

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in with your credentials.
3. On the **Home** page, click **Profiling**.
4. Click the **Add Profile** button . The **Add Profile** page is displayed with a wizard showing three stages of profile creation: **Select Source** > **Configure Profile** > **Finish**.
5. From the **Select data source** options, select the **Flat File** option, and click the **Add Files** button. The **Choose File** pop-up window is displayed.
6. Use the file path selection box  / **Server** / **E:** / **Sample Files** to go to the path on your local machine or server where the files are located. A list of files is displayed below the file path selection box. You can filter required files from the list by using the **Filter** type ahead text box.
7. Select all the files you need to create the profile and click **OK**. The selected files are displayed on the left side of the page, below the **Add Files** button.
8. To add more files, click the **Add Files** button again and repeat steps 6 and 7.

Note: You can add files from different folders on your local system, on server, and FTP server in any combination you want. For using FTP server, you need to configure it first. For more information, see the Administration Guide.

9. Click any file in the list to view these details:
 - The number of files selected. For example *7 File(s) selected*
 - Color coding for file's metadata
 - **Green**: Metadata detected automatically
 - **Blue**: User configured metadata
 - **Red**: Metadata not detected. Such files are skipped during profiling. However, you can configure the metadata of these files using the **File Setting** fields as described in this task below. After manual configuration, these files turn blue.
 - Path of the file. For example, `E:/Sample Files/profile/fileOne.txt`
 - File settings
 - Preview of the records in the file
10. To modify settings for any of the files, select it and make changes as described in this table.

Settings	Description
Character encoding	<p>The text file's encoding. Select one of these:</p> <p>CP1252 This encoding is also known as the Windows-1252 or simply Windows character set. It is a super set of ISO-8859-1 and uses the 128-159 code range to display additional characters not included in the ISO-8859-1 character set.</p> <p>UTF-8 Supports all Unicode characters and is backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html.</p> <p>UTF-16 Supports all Unicode characters but is not backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html.</p> <p>US-ASCII A character encoding based on the order of the English alphabet.</p> <p>UTF-16BE UTF-16 encoding with big endian byte serialization (most significant byte first).</p> <p>UTF-16LE UTF-16 encoding with little endian byte serialization (least significant byte first).</p> <p>ISO-8859-1 An ASCII character encoding typically used for Western European languages. Also known as Latin-1.</p> <p>ISO-8859-3 An ASCII character encoding typically used for Southern European languages. Also known as Latin-3.</p> <p>ISO-8859-9 An ASCII character encoding typically used for Turkish language. Also known as Latin-5.</p> <p>CP850 An ASCII code page used to write Western European languages.</p> <p>CP500 An EBCDIC code page used to write Western European languages.</p> <p>Shift_JIS A character encoding for the Japanese language.</p> <p>MS932 A Microsoft's extension of Shift_JIS to include NEC special characters, NEC selection of IBM extensions, and IBM extensions.</p> <p>CP1047 An EBCDIC code page with the full Latin-1 character set.</p>

Settings

Description

Field delimiter

Specifies the character used to separate fields in a delimited file.


For example, this record uses a pipe (|) as a field delimiter:

```
7200 13TH ST|MIAMI|FL|33144
```

The characters available as field delimiter are:

- Comma
- Semicolon
- Pipe
- Tab
- Space
- Period

You can also add custom field delimiters, to add a custom field delimiter follow these steps:

- a. Click the Add button  placed adjacent to **Field delimiter**. **Add Separator** pop-up window is displayed.
- b. Enter your desired field delimiter in the **character** field, the corresponding **Unicode** is displayed automatically.
- c. Enter a suitable name for your delimiter in the **Description** field.
- d. Click **Save**, your delimiter appears in the **Field delimiter** drop down.

Text qualifier

The character used to surround text values in a delimited file.

For example, this record uses double quotes (") as a text qualifier.

```
"7200 13TH ST"|"MIAMI"|"FL"|"33144"
```

The characters available to define as text qualifiers are:


- Single quote (')
- Double quote (")

If the file uses a different text qualifier, click the ellipses button to select another character as a text qualifier.

Settings	Description
Line separator	<p>Specifies the character used to separate records in line in a sequential or delimited file.</p> <p>The record separator settings available are:</p> <p>Unix (U+000A) A line feed character separates the records. This is the standard record separator for Unix systems.</p> <p>Macintosh (U+000D) A carriage return character separates the records. This is the standard record separator for Macintosh systems.</p> <p>Windows (U+000D U+000A) A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.</p>
First row is header row	<p>Specifies if the first record in a delimited file contains header information. A Yes indicates it has header information.</p> <p>For example, this file snippet shows a header row in the first record.</p> <pre>"AddressLine1" "City" "StateProvince" "PostalCode" "7200 13TH ST" "MIAMI" "FL" "33144" "One Global View" "Troy" "NY" 12180</pre>

11. To apply settings of one file to another:
 - a. Select the file from which you want to copy the settings, and click the **Copy File Settings** button.
 - b. In the **Apply file settings** pop-up window that is displayed, select the files to which you want to apply the settings, and click **OK**.

Note: The file settings are applied to the selected files.

12. To see preview of the selected file, see the **Preview** section below **File Settings**.
13. To configure the profile you created, click  on the top right corner of the page.


Adding Data from Models

To create a new profile from physical or logical models, perform these steps:

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.


2. Log in with your credentials.
3. On the **Home** page, click **Profiling**.
4. Click the **Add Profile** button .
The **Add Profile** page is displayed with a wizard showing three stages of profile creation: **Select Source > Configure Profile > Finish**.
5. From the **Select data source** options, select the **Model** option.
6. From the **Select model** drop-down list, select the physical or logical model you want to use to create your profile. To narrow down your search by model type (physical or logical) or model name, type it in the field and the related models will start displaying below the field. For example, if you type "logical" all the logical models will be displayed.

Note: To create a new model or to modify an existing one, click on the **Manage models** link. It navigates you to the **Modeling** page, where you can create and save a new physical/logical model. This model can then be selected to create profile using the **Select model** drop-down list.

The tables and columns in the selected model are displayed below the field.

7. From the list of tables, select tables that you want to include in the profile. In each of the tables, you can also select the columns you need for the new profile. To include all the tables and columns, select the check box in the header row.

Note: Use the type-ahead search text box to search for specific tables/columns.

8. To configure the profile you created, click  on the top right corner of the page.


Adding Data from a Connection

To create a new profile from the any of the database connections configured in Spectrum Technology Platform, perform these steps:

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in using your credentials.
3. On the **Home** page, click **Profiling**.
4. Click the **Add Profile** button .
The **Add Profile** page is displayed with a wizard showing three stages of profile creation: **Select Source > Configure Profile > Finish**.
5. From the **Select data source** options, select the **Connection** option.
6. From the **Select connection** drop-down list, select the connection you want to use to create the new profile. To narrow down your search by connection type or its name, type it in the field and

the related connections will display below the field. For example, if you type `mssql` then all the MSSQL server connections will display.

Note: The drop-down list displays all the connections that are configured in Spectrum Technology Platform. To configure a new connection, use the **Add Data Sources** page, which you can access through the **Data Sources** tab in Spectrum Discovery or with the Management Console. For steps to configure a new data source, see the section [Connections](#) on page 12.

7. Click the **Select tables** button.
The **Select tables** pop-up window is displayed showing all the tables in the selected connection.
8. From the list of tables in the **Select tables** pop-up window, select the tables that you want to include in the profile and click **OK**.
The selected tables are displayed below the **Select tables** button on the **Add Profile** page.
9. To select columns from a table, click **>**, and select the columns needed for creating the profile. To include all the columns, select the check box in the table header.

Note: Use the type-ahead search text box to search for specific tables/columns.

10. To configure the profile you created, click **>** on the top right corner of the page.

Configuring rules for profiling

This task describes the process of configuring the rules based on which you want to profile your data set. These settings include:

- Rules of data quality analysis
- Sample size for each table in the result
- List of users to whom the profiling results are to be sent automatically

Note: To be able to send out automated mails, you need to configure a mail server using the **Management Console**. For more information, see the *Administration Guide*.


Besides, you can also specify the number of tables to be run concurrently in profiling and the number of runs that are saved for data preview. For more information, see [Specifying Profiling Default Settings](#) on page 226.

You can configure your profile using the **Configure Profile** stage of the **Add Profile** page. To do this, perform these tasks.



1. In the **Configure Profile** stage of the **Edit Profile** page, select the required **Profiling rules**. For the list of profiling rules and their description, see [Profiling Rules](#) on page 219.

- Specify the sample size of each table using the **Sample size per table** field. For example, if you specify 10000, the analysis will take into consideration 10000 rows of data in every analysis sample and generate report for it.

Note: You can fetch random records while profiling your models or connections. To enable this feature, select the **Fetch Random Records** check box present below the **Sample size per table** field.

- In the **Notification** section, select the option **When the data quality analysis is completed send an email to** if you want to email the analysis report to users. Specify the email addresses of the users. To add multiple email addresses, click the **Add** button .

Note: To be able to send out automated mails, you need to configure a mail server using the **Management Console**. For more information, see the *Administration Guide*.


- For any kind of correction or update, you can go to the previous stage by clicking  on the top right corner of the page.
- Click  on the top right corner of the page to move to the next stage of profile creation.

Profiling Rules

Profiling rules perform different types of analysis on your data. When setting up a profile, choose the profiling rules that perform the types of data analysis you are interested in.


This section describes the profiling rules supported in Spectrum Discovery.

Character Analysis

This rule identifies patterns, scripts, and character types in string fields. To configure this rule when creating a profile, click the Configure icon  and add, edit, or delete special characters from the prepopulated list. When you enable this rule, you will get this information for the string fields in your data:

- **Frequency:** The most-used phrases in the selected string column.
- **Scripts Distribution:** The different scripts identified in the selected string column and their count.
- **String Lengths:** The distribution of string lengths in the selected string column.
- **Character Categories:** The types of characters in the selected column, such as letter, punctuation, and number.
- **Text Patterns:** Converts data in the string column to pattern and displays the pattern, its count, and percentage occurrence. The pattern is determined using this rule:
 - Latin upper-case characters are replaced with "A"
 - Latin lower-case characters are replaced with "a"
 - Digits are replaced with "9"
 - Control characters are replaced with "C"

Custom Pattern Analysis

This rule identifies any pattern in the string column. You can configure as many regular expressions as you want to match your data against. To configure this rule when creating a profile, click the Configure icon , and enter these details:

1. **Key:** Name of the pattern to be identified
2. **Value:** The regular expression for the pattern

For example, if you want to identify email addresses in string columns, enter the regular expression `^(.+)\@(.+)\$` in the **Value** field and email in the **Key** field.

3. To add another expression, click the Add icon and add details of the next key value pair. You can add as many expressions as you would like to match the data against.

When you enable this rule, you will get this information:

- **Validity:** The values that matched at least one of the regular expression patterns in the rule.
- **Pattern Match Distribution:** The distribution of records that matched the regular expressions.

Internal Rule

Determines statistics, such as completeness, uniqueness, frequency and outliers in the data set.

Semantic Analysis

1. **User Defined:** This rule detects the user-defined semantic types in the **Define Glossary** and uses it to profile the data set. The user-defined semantic types can find out data based on these parameters:
 - a. If you defined an expression in the semantic type, it finds out its match in the data set and gets the result on the **Data Profiling Results** page.
 - b. If you specified the data to be looked for, the user-defined semantic type looks for it in the data set and presents it on the **Data Profiling Results** page. For example, if *Single, Married* is specified in the **Exists In** option of the semantic type, it will fetch all the records that have these value.
2. **Advanced Transformer Look-up:** This rule determines the user-defined tables in the Advanced Transformer stage and uses it to profile the data set. To run this rule, you need to install Spectrum Data Normalization and load the Advanced Transformer reference tables. The tables you configure appears in the **Semantic Type** tab on the **Define Glossary** page, which are non-editable.

Note: You can edit the tables in Enterprise Designer or Flow Designer.


If you select this rule, the **Data Profiling Results** page displays the detected semantic types if the data matches with the table defined in the stage.

3. **Open Parser Look-up:** This rule determines the user-defined tables in the Open Parser stage and uses it to profile the data set. To run this rule, you need to install Spectrum Data Normalization and load the Open Parser reference tables. The tables you configure appears in the **Semantic Type** tab on the **Define Glossary** page, which are non-editable.

Note: You can edit the tables in Enterprise Designer or Flow Designer.

If you select this rule, the **Data Profiling Results** page displays the detected semantic types if the data matches with the table defined in the stage.

4. **Credit Card Validation Analysis:** Select this rule to detect and validate credit card numbers and identify credit card numbers as JCB, VISA, Diners Club (DINERS), MasterCard, Discover, or American Express (AMEX). If you select this rule, the **Data Profiling Results** page displays an additional Credit Card Summary tab showing these details:
 - a. **Validity:** The valid and invalid credit card numbers.
 - b. **Credit Card Distribution:** Category-wise distribution of the detected credit cards
5. **Date Analysis:** This rule detects and validates dates in string columns. It also identifies date patterns in the columns and their distribution. This analysis can be useful in detecting date entries in erroneous columns, for example in email data. If you select this rule, Data Profiling Results page displays an additional Date Summary tab for the string columns that have dates. This tab shows these details:
 - a. **Validity:** The valid and invalid values.
 - b. **Date Patterns:** The date patterns detected in the selected columns, their total count, and percentage of that pattern in the data set.
6. **Email Analysis:** This rule detects and validates the email addresses and determines the distribution of email domains in the selected data column. If you select this rule, Data Profiling Results page displays an additional Email Summary tab showing these details:
 - a. **Validity:** The valid and invalid values.
 - b. **Domain Distribution:** The top ten email domains in the selected data column.
7. **Phone Number Analysis:** Select this rule to detect and validate phone numbers and identify phone numbers as fixed line numbers, mobile numbers, or any other type of number. This rule also gives the distribution of the phone numbers by country and region. You need to configure this rule to define the default country to use when a phone number does not have a country code. If you select this rule, **Data Profiling Results** page displays an additional **Phone Number Summary** tab showing these details:
 - a. **Validity:** The valid and invalid phone numbers.
 - b. **Phone Number Types:** The types of phone numbers, such as mobile, land line, fixed line, VOIP, Pager, voice mail, or toll-free.
 - c. **Phone Numbers by Country:** The country-wise distribution of the detected phone numbers.
 - d. **Phone Numbers by Region:** The region-wise distribution of the detected phone numbers


8. **Vehicle Identification Number (VIN) Analysis:** Select this rule to detect and validate vehicle identification numbers. This rule also gives the distribution of Vehicle Identification Numbers by country. If you select this rule, the **Data Profiling Results** page displays an additional **VIN Summary** tab showing these details:
 - a. **Validity:** The valid and invalid vehicle identification numbers.
 - b. **VIN Country Distribution:** Country-wise distribution of the detected vehicle identification numbers.
9. **Social Security Number (SSN) Analysis:** Select this rule to detect and validate social security numbers. If you select this rule, the **Data Profiling Results** page displays an additional **SSN Summary** tab showing the valid and invalid social security numbers.
10. **International Bank Account Number (IBAN) Analysis:** Select this rule to detect and validate international bank account numbers. This rule also gives the distribution of International Bank Account Numbers by country. If you select this rule, the **Data Profiling Results** page displays an additional **IBAN Summary** tab showing these details:
 - a. **Validity:** The valid and invalid international bank account numbers.
 - b. **IBAN Country Distribution:** Country-wise distribution of the detected international bank account numbers.
11. **Semantic Analysis:** Select this rule to detect semantic types, such as first name, city, country, ISO country code 2 and 3, last name (family name), and states. This rule can help find values in incorrect columns, such as city names in a Country column. If you select this rule, the **Data Profiling Results** page displays an additional **Semantic Type** tab showing the detected semantic types and their frequency.
12. **U.S. Address Analysis:** This rule determines the quality of your address data using the U.S. database of Spectrum Universal Addressing. To run this rule, you need to:
 - a. Install the Spectrum Universal Addressing U.S. database and define it as a resource in Management Console. For more information about adding this database resource, the *Administration Guide*.
 - b. Configure the U.S. Address Analysis rule by clicking the Configure button  and entering this information:
 - **US Address Coder database:** Select the Spectrum Universal Addressing database resource configured in Management Console.
 - **AddressLine1 field to AddressLine5 field:** Map these fields to the columns of the table you are analyzing. You do not necessarily need to enter column names in all the fields. However, the more specific you are the better the matching score will be.
 - Map the columns from your table to the **City**, **Country**, **USUrbanName**, **FirmName** field, **PostalCode**, and **StateProvince** fields.

If you select this rule, the **Data Profiling Results** page shows the **Address Summary** tab.

 - The legend below the chart shows the match score for the data along with the color coding.

- Point anywhere in the graph area to view the match score. The scores are in ranges (0, 1-25, 26-50, 51-80, 81-99, and 100), with zero representing no match of the data to the database. The graph also shows the percentage of matching records detected (color coded).
- Click the area in the graph to view the data that matched or did not match to the database.

13. **International Address Analysis:** This rule determines the quality of your address data using the Spectrum Global Address Validation database. To run this rule, you need to:

- Install the Spectrum Global Address Validation database and define it as a resource in Management Console. For more information, see the *Administration Guide*.
- Configure the International Address Analysis rule by clicking the Configure button , and entering this information:
 - **Addressing Engine Database:** Select the Global Address Validation database resource configured in the **Management Console**.
 - **Confidence Threshold:** Enter the value of confidence threshold to detect fields falling below the specified value. The default value of this field is **80**.
 - **Table List:** Select the table on which you would like to run this rule.
 - **AddressLine1 field** and **Country** field: Map these fields to the columns in the table you are analyzing.
 - Map the columns from your table to the **LastLine**, **City**, **CitySubdivision**, **PostalCode**, **State**, **StateSubdivision**, and **FirmName** fields. You do not necessarily need to enter column names in all the fields. However, the more specific you are the better the matching score will be.


If you select this rule, the **Data Profiling Results** page shows and **Address Summary** tab, which displays:

- **International Address Confidence Distribution:** The match score for the data. The score is color coded. Point anywhere in the donut chart to view the score range (0, 1-25, 26-50, 51-80, 81-99, and 100), with zero representing no match of the data in the database. Click the area in the chart to preview the matching or non-matching data.
- **International Address Precision:** This distribution of validation levels of addresses, such as state, house, postal code, city, city sub-division, and street.
- **International Address Mismatched Fields Distribution:** The distribution of mismatched fields of addresses, such as City Subdivision, State Province, Street Name, and Postal Code are displayed here. The confidence threshold of this address is less than the value defined by you while configuring the **International Address Analysis** rule.

Service Configuration

Use this to consume the default Spectrum services and the services you have configured using the different Spectrum stages you licensed for. This configuration helps you reuse the transformations designed with various Spectrum stages on your data. Let us understand it with this example.

Example: You have used the **Validate Address** stage and the **Conditional Router** stage to filter out the validated addresses. You can use this transformation to profile your input data, so that profiling runs on the validated addresses. All you need to do is expose this service in **Enterprise Designer** and consume it here (in **Spectrum Discovery**) as **Service Configuration** rule. Let us see, how to do this.

1. Select the **Service Configuration** rule, and click the corresponding gear icon .
2. In the **Service Configuration** pop-up, specify these details.

Field	Description
Services	<p>From the drop down list, select the required service. It lists all the exposed services.</p> <p>Note: You can add multiple services by clicking the Add Services link.</p>
Service input fields and Source input fields	<p>Displays the list of the fields used as input in the selected service. Every Service input field has a corresponding Source input fields. Use the Source input fields drop-down list to select the required field from your input data.</p> <p>Data in the selected field will be used as input when you run profile.</p> <p>Note: The data types of input fields and source fields with which those are mapped should be the same.</p> <p>Example:</p> <ul style="list-style-type: none"> • The Service input fields are First_Name and Last_Name. • In the corresponding Source input fields, you select FirstName and LastName. <p>Now, when you run profiling, the data in the FirstName and LastName columns in your input sources will be used for profiling. The transformations defined in the selected service will run on this input data and those will be profiled to give you the reports.</p>
Service output fields	<p>Shows the output fields in the selected service. Select the fields that you want in your profile output. You can use the type-ahead search box to find out the required fields.</p> <p>Note: If the service has <i>List</i> data type as output, you can't use that in profiling.</p>

3. Click **OK**.
4. When you run profiling, the output will be displayed on the Data Profiling Results page under **Name_Service**. Click on any of the output fields to view the related statistics.

Note: You can use the services and obtain profiling outputs even if you do not have a license for the respective module or your license has expired. The Profile Summary page will have an additional tab to indicate the license exception.

Related concepts

[Viewing Data Profiling Results](#) on page 232

Defining Profile

The third and final stage of profile creation involves defining the profile, in terms of assigning it a name and description. To do this, on the Finish stage of the **Add Profile** page, perform these tasks:


1. Enter a name for the profile in the **Profile name** field.
2. Describe the profile in the **Description** field.

Note: The description can include the purpose of profile creation.

3. Set the **Runtime Settings** to improve the performance of the profiling process along with the default profiling options:

Enter the value for **Column threads per table**, this option divides column processing into threads. These threads run concurrently in a single job. The default value is 1.

For a **single table or file**, this value can be *equal to the number of logical cores of the system*. Whereas, for **multiple tables or files**, this value should be such that *on multiplying it with number of tables, the result should be less than or equal to the number of cores available*.

4. For any kind of correction or update, you can go to the previous stage by clicking  on the top right corner of the page.
5. To save the profile and run is later, click **Save**.
The profile is created and added to the **Profiles** page.
6. To save and run the profile, click **Save and Run**.
The profile is created and added to the **Profiles** page and the process of analysis begins on it. You can view details, such as status of the profile run on the **Profiles** page.

Note: If you have limited access to server directories, your profile will not run. For the profile to run, give **Directory Access** to the path where you have installed spectrum.

Specifying Profiling Default Settings

For the profiles you have configured, you can modify these two settings:

- How many tables should run concurrently during profile run
- For how many profile runs should data in files be available for preview

These rules help you optimize performance based on your requirements and file size. To specify these settings, perform these steps:

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in using your credentials.
3. On the **Home** page, click **Profile & Monitor**.
The **Profile & Monitor** page is displayed.
4. Click the **Defaults** link on the top right of the page.
The **Profiling Defaults** page is displayed.
5. In the **No. of tables for concurrent profiling** box, specify the number of tables that are to be run concurrently for analysis.

Note: This setting is applicable for all the profiles that you set to be run. The valid range of tables is 1-20, with default value being 5. Ideally, this value should be $[(\text{Number of logical cores of the system}/2) - 1]$

Example: Number of tables for concurrent profiling is specified as 5 and two profiles, each having six table are set to run:

Spectrum Discovery runs five tables from the profile scheduled to run first and keeps the second profile in queue (the status is displayed in the **Status** field on the **Profiles** tab of the **Profile & Monitor** page). Other tables are picked one after another as profile run completes on the picked-up tables.

6. In the **Keep source data for [N] runs** box, specify the number of runs for which the tables picked up in the profile analysis are saved in data preview. Default is 2.

Example: Keep source data for [N] runs is set to 5

Spectrum Discovery does not show the table data in preview after five runs are completed for those. If you click the graphs, bars, or donut charts corresponding to these tables on the **Data Profiling Results** page, you will get the message: *Preview is not available*. However, the graphs, bars and donut charts are displayed irrespective of the runs specified in this field.

Analyzing a Profile

After a profile is created and listed, you can send it for analysis. The profile is analyzed based on the rules selected for it and a report gets displayed on the **Data Profiling Results** page. The report can also be sent directly to the targeted users if you specify this on the **Add Profile** page and you can also download it as a PDF or Excel document.

Note: The report can be downloaded as an Excel document only at profile and table level.

The profile analysis can be run as a one time activity or a recurring event by scheduling the recurrence in the **Add Schedule** page. You can also start analysis directly from the **Profiles** tab. In all these cases the analysis is run and the status displayed on the **Profiles** and the **History** tab.

When a profile starts getting analyzed, a unique ID is assigned to it, which is displayed on the **History** tab. For more information, see [Viewing Analysis History](#) on page 230.


Note: If the same profile is sent for analysis multiple times, every analysis is assigned a different ID.

To run profile analysis from the list of profiles displayed on the **Profiles** tab, perform these steps:

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in with your credentials.
3. On the **Home** page, click **Profile & Monitor**.
4. From the list of profiles displayed, select the check box corresponding to the one you want to analyze, and click the **Run Profile** icon .

Note: You can search for the required profile by using the type-ahead **Filter** text box. It helps you search on the basis of the Name, User, or Data Source of the profile.


The selected data set of the profile starts getting analyzed, with the **Status** column showing the in-progress icon. The icon turns to a tick mark when the profiling is complete. The other details displayed for the profile are:

- **Name:** The name of the profile. Clicking on the profile name takes you to the **Data Profiling Results** page. For more information, see [Data Source Details in Profiling Results](#) on page 233.
- **Data Source:** The source of the data in the profile being analyzed. It can be a physical or logical model or a direct data source connection.

- **User:** The user who created the profile
- **Last Run Start Time:** The date (in *mm/dd/yy* format) and time (in *hh:mm* and AM/PM format) when the most recent profile analysis started
- **Last Run End Time:** The date (in *mm/dd/yy* format) and time (in *hh:mm* and AM/PM format) when the most recent profile analysis completed
- **Duration:** The duration (in *mm:ss*) of the analysis
- **Status:** If the analysis is in progress, complete, or failed
- **Tables** analyzed
- **Records** analyzed
- **Suggestions:** Click the suggestions icon to view the suggestions. You can copy the **Recommendations** provided in the form of groovy scripts and further use those in the **Enterprise Designer**.
- **Reports:** Click the PDF or Excel icon to download the analysis report. In case the profiling hits some roadblock, you can view the error in this field.

Scheduling Profile Analysis

To schedule a profile analysis, use the **Add Schedule** page. This page gives you various combinations of pattern, time, and date range to schedule the profiling in the way that suits your need. You can also have it as a one-time activity.

1. On the **Profile & Monitor** page, click the **Schedules** tab, and click the **Add Schedules** icon . The **Add Schedule** page is displayed.
2. Enter a **Name** for the schedule.
3. Select the **Profile** option, and from the drop down list, select the profile that you want to schedule for analysis.

Note: It displays all the defined profiles.

4. In the Schedule section, select one of the triggers as needed from the **Trigger** drop-down list.
 - Select **Date/Time** to schedule a one-time analysis. Use the **Start date** and **Start time** from the respective calendars below the drop-down list.
 - Select the **Recurring Date/Time** option for a recurring analysis.

Note: When you select this option, **Pattern**, **Time** and **Range** sections appear below Trigger. Use these for setting up a recurring analysis, as described below.

5. To set up a recurring analysis, use the one of these options or a combination of all of these:
 - **Pattern** (of days or week or month)
 - **Every weekday:** To run the analysis on every day of the week

- **Every:** At a specified periodicity. For example, to run analysis every alternate day, choose this option, and enter 2 in the selection box.
- **Day <> of every <> month(s):** To run analysis on a specified day and month. For example, to run analysis on the third day of every alternate month, enter 3 in the **Day** selection box and 2 in the **month(s)** selection box.
- **Every <> month(s) <on a specified day> <of a week>:** For example, to run the analysis every third month on the second Tuesday, enter 3 in the **month(s)** selection box, and from the drop downs, select **on the second** and **Tuesday** options respectively.
- **Days of the week:** Select the days of the week on which you want to run the analysis.

Note: You can select multiple days.

- **Time** To run the analysis at a specified interval of time, select this option and in the selection boxes specify the number of hours or minutes after which you want to run the analysis. For example, every three hours.
- **Range** To specify the range of analysis (at a specified time between a range of dates), select the **Start date** and **End date** from the respective calendars and select the time and period (AM/PM) from the **Start time** option.

Note: Select the **No end date** check box if you do not want to specify an end date for the recurring analysis.

6. To activate this schedule, ensure the **Schedule** slider (on the top right of the page) is **On**.
7. Click the **Save** button.

The schedule is saved and gets displayed on the **Schedules** tab of the **Profile** page. This page lists all the profiles that are scheduled for analysis and displays these details for each of those:

- **Schedule Name:** Name of the schedule
- **Schedule On:** Name of the Profile or Scorecard that is scheduled for analysis
- **Type:** Whether the schedule is for a **Profile** or a **Scorecard**.
- **User:** The user who created the schedule
- **Trigger:** It specifies **SCHEDULE** to indicate that the profile or scorecard is scheduled for analysis
- **Next Run:** Specifies the date, time, and period when the analysis will run again.
- **Last Run:** Specifies the date, time, and period when the analysis was last run.

Note: When analysis begins, the in-progress profile also gets displayed on the **Profiles** tab of the **Profile** page in case of recurring scheduling. If the scheduling is non-recurring, the in-progress profile gets moved from the **Schedules** tab to the **Profiles** tab the moment the processing begins.

Scheduling Recurring Analysis

You can set the recurrence of profile analysis by using the required combination of **Pattern**, **Time** and **Range** in the **Schedule** section of the **Add Schedule** page.

Pattern

- **Every weekday**: To run the analysis on every day of the week
- **Every**: At a specified periodicity. For example, to run analysis every alternate day, choose this option, and enter 2 in the selection box.
- **Day <> of every <> month(s)**: To run analysis on a specified day and month. For example, to run analysis on the third day of every alternate month, enter 3 in the **Day** selection box and 2 in the **month(s)** selection box.
- **Every <> month(s) <on a specified day> <of a week>**: For example, to run the analysis every third month on the second Tuesday, enter 3 in the **month(s)** selection box, and from the drop downs, select **on the second** and **Tuesday** options respectively.
- **Days of the week**: Select the days of the week on which you want to run the analysis.

Note: You can select multiple days.

Time

To run the analysis at a specified interval of time, select this option and in the selection boxes specify the number of hours or minutes after which you want to run the analysis.

Range

To specify the range of analysis, select the **Start date** and **End date** (in *dd/mm/yy* format) from the respective calendars and select the time (in *hh:mm* format) and period (AM/PM) from the **Start time** option.

Note: Select the **No end date** check box if you do not want to specify an end date for the recurring analysis.

Viewing Analysis History

The **History** tab of the **Profile & Monitor** page displays a list of all the analysis run on various profiles along with details, such as the start and completion time of the analysis, whether it was completed

successfully, the unique ID assigned by the system to the analysis, the number of tables and records analyzed in each case, and the data source of the profiles analyzed.

Note: It is recommended to keep deleting the obsolete profile runs from history.

To view the history of all the analysis run by the application, perform these steps:

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in with your credentials.
3. On the **Home** page, click **Profile & Monitor**.
4. Click the **History** tab.

A list of all the analysis run on various profiles is displayed, along with these details:

- **ID:** The unique identification assigned by the application to this profile analysis.



Note: In case the same profile is analyzed multiple times, each run will have a unique ID.

- **Name:** The name of the profile. Clicking on the profile name takes you to the **Data Profiling Results** page. For more information, see [Data Source Details in Profiling Results](#) on page 233.
- **Data Source:** The source of the data in the profile being analyzed. It can be a physical or logical model or a direct data source connection.
- **User:** The user who created the profile
- **Last Run Start Time:** The date (in *mm/dd/yy* format) and time (in *hh:mm* and AM/PM format) when the most recent profile analysis started
- **Last Run End Time:** The date (in *mm/dd/yy* format) and time (in *hh:mm* and AM/PM format) when the most recent profile analysis completed
- **Duration:** The duration (in *mm:ss*) of the analysis
- **Status:** If the analysis is in progress, complete, or failed

5. To view profiling report, select the required profile, and click the **View History**  icon.

Note: Use the type-ahead **Filter** text box to search the profile faster. It helps you search on the basis of the Name, Data Source, and User field details.

The **Data Profiling Results** page is displayed. For more information, see [Data Source Details in Profiling Results](#) on page 233.

6. To refresh the page, click the **Refresh History** icon .
7. To delete an analysis, select it and click the **Delete History**  icon.

Viewing Data Profiling Results

The **Data Profiling Results** page displays a comprehensive analysis of a profile, including the completeness and uniqueness of the contained data. The report is generated at the profile level as well as table and column level. At the column level the result displayed varies depending upon the profiling rules selected by you. For example, if you select **Semantic Analysis** rule on the **Add Profile** page, the **Data Profiling Results** will show the detected semantic types in the selected string columns.

Note: In all the charts, legends are hyper-linked and those lead you to individual records in the category.

You can download the results in the PDF or Excel format.

Note: If the Java Virtual Machine does not have the required fonts, the downloaded reports will be empty files and the error logs will indicate that the fonts are not available to the JVM. To install the missing font files (.tff), follow the instructions here:

<https://support.azul.com/hc/en-us/articles/360034030692-Using-Fonts-with-OpenJDK-Zulu>.

To access the **Data Profiling Results** perform these steps:

1. In a web browser, navigate to: `http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080

2. Log in using your credentials
3. On the **Home** page, click **Profile & Monitor**.
4. Perform one of these actions:
 - On the **Profiles** tab, click the **Name** of the profile for which you want to view the analysis report.

Note: The profile **Name** becomes hyper-linked only if the analysis has been run for it.

- On the **Profiles** tab, click the PDF or Excel icon in the **More** column corresponding to the required profile.
- On the **History** tab, click the ID of the profile for which you want to view the analysis report.

The **Data Profiling Results** page is displayed showing the results of profile analysis. This page is divided into 3 parts:

- A header showing the profile details. These details are displayed for the analyzed profile:
 - **Profile name:** Name of the profile
 - **Description:** A brief description of the profile

- **Started:** The date (yyyy/mm/dd), time (hh:mm:ss), and period (AM or PM) when the profile analysis was initialized.
- **Data Source:** The source of the data in the profile being analyzed. It can be a physical or logical model or a direct data source connection.
- **User:** The user who created the profile
- **Tables:** Number of tables in the selected model
- **Data source type:** Whether the data source for the profile is a model or a database
- **Records:** Total number of records analyzed
- **Suggestions Button:** Navigates you to the **Suggestions** page. For more information, see [Viewing the Suggestions Page](#) on page 236.
- The left pane that displays the profile name and all the tables and column included in it
- The report pane gives a pictorial view of the analyzed data

The report displayed in the right pane is based on what you select in the left pane (profile, table, or column). Details, such as Standard deviation (**Stdev**), Maximum (**Max**), Minimum (**Min**), and **Average** value are displayed at the column level for numerical data, while length frequency and text pattern frequency is displayed for string columns. Besides, based on the rules selected for profiling, additional tabs are displayed in the report, such as **Semantic Type** for columns having semantic type values if **Semantic Analysis** rule is selected. For more information, see [Profiling Rules](#) on page 219.

Additionally, you can collaborate on your profiling results using the comments feature available at the model, table, and column level. For more information, see [Collaborating on Data Profiling Results](#) on page 239.

Data Source Details in Profiling Results

Click the data source name in the left pane to display these details:

- **Completeness:** The percentage of complete and incomplete rows detected in your profiled data.
Note: Click the **Incomplete Rows** and **Complete Rows** hyperlink to view the records in both categories.
- **Model Summary:** Displays these details for every table in the profile
 - **Table Name:** Names of all the tables in the profile
 - **Number of Columns:** Number of columns in the table
 - **Number of Records:** Number of records in the table
 - **Completeness (%):** Completeness of records in the table

Table Details in Profiling Results

Click any of the table names in the left pane to display these details:

- **Completeness:** The percentage of complete and incomplete rows detected in your profiled data.

Note: Click the **Incomplete Rows** and **Complete Rows** hyperlink to view the records in both categories.

- **Table Summary:** Displays these details for every column in the table
 - **Column Name:** Names of all the columns in the table
 - **Data Type:** The data type in the column
 - **Completeness (%):** Completeness of records in the column
 - **Uniqueness :** Uniqueness of the data contained in the column
 - **Detected Type:** Displays the semantic types, such as email, phone, city, first name, and last name detected in the string in this column.
 - **Other Stats:** Displays various other statistics such as **Min Length**, **Max Length**, and **Text Patterns**
- **Null Count Frequency:** Displays the number of null values in every row of the table

Viewing Outliers

You can view a summary of the outliers detected for each column in your table by clicking the **Outlier Analysis** tab. This tab displays category-wise occurrences of any pattern, value, length, or frequency for a column that falls outside the range of other observations.

For example, The permissible length of **Country Names** is up to **14** characters and strings having **15** characters are detected in your data; this string will be categorized under **Length Outlier**. The supported categories are **Length Outliers**, **Frequency Outliers**, **Text Pattern Outliers**, **Numeric Outliers**, **Semantic Type Outliers**, and **Data Type Outliers**.

Viewing Malformed Records

In **Flat File** profiling you can view malformed records in your table by clicking the **Malformed Records** tab. This tab displays the **Category** and **Count** of the malformed records. A record is treated as malformed for these categories:

- Rows with lesser number of fields than the number of defined columns
- Rows with more number of fields than the number of defined columns

You can also display a preview of the malformed records by clicking the Category. The preview displays the **Record Number**, **Record**, and the **Reason** for categorizing a record as malformed.

Column Details in Profiling Results

Based on what column you click in the left pane of the **Data Profiling Results** page, these details are displayed:

Note: The categories in the doughnut charts and values on X and Y axes of graphs are hyper-linked. Clicking the links navigates you to the associated list of records.

- **Completeness:** The percentage of **Complete**, **Null**, and **Empty Strings** detected in the column. Click the hyperlink to view the records in each categories.
- **Uniqueness:** These statistics are displayed here:
 - **Unique:** Records with no duplicates in the data source.
 - **Non- unique:** Records having duplicates in the data source.
 - **Distinct:** A list of all records present in your data source irrespective of those being unique or non-unique record.

For example, your column contains these names:

```
Roger
Gigi
Gigi
Gigi
Garey
Elena
Brad
Brad
```

Here:

- *Roger, Garey, and Elena* are **unique** records.
- *Gigi and Brad* are **non-unique** records.
- *Roger, Gigi, Garey, Brad, and Elena* are **distinct** records.
- **Min Length** and **Max Length** of characters in case the column data is a string
- Standard deviation (**Stdev**), Maximum (**Max**), Minimum (**Min**), **Variance**, and **Average** value in case the column has numerical data.
- **Histogram** representing the distribution of data in case the column has numerical data
- **Percentile** in case the column has numerical data
- **Frequency** of data in the column. Click the hyper-linked value on the X axis to go the list of associated records.
- **String Lengths** in case the column data type is string
- **Scripts Distribution** in case the column data type is string
- **Text Pattern Frequency** in case the column data type is string

- **Character Categories:** Graphically displays the frequencies of Latin character types detected in the selected string column. The various categories are:
 - **Casing:** Upper Case, Lower Case, and Mixed Case
 - **Character Data Types:** Alphabetic, Numeric, and Alphanumeric
 - **Contains Spaces:** Single Space, Multiple Spaces, and Trailing or Leading Spaces
 - **Special Character:** Contains or Does not Contain.

Note: Only the special characters defined during configuration of **Character Analysis** rule will be considered here.


Viewing Outliers

You can view the outliers detected in a column by clicking the **Outliers** tab. This tab displays a tabular structure of category-wise detected outliers in your column. Details such as **Pattern**, **Length**, **Value** or **Frequency**, **Computed Range**, and **Occurrence** are also displayed.

Viewing the Suggestions Page


This page displays a comprehensive view of various suggestions generated for your data. You can choose to select these suggestions to cleanse your data.

This page is divided into two parts:

- A header showing the profile details - These details are displayed for the analyzed profile:
 - **Profile name:** Name of the profile
 - **Description:** A brief description of the profile
 - **Started:** The date (yyyy/mm/dd), time (hh:mm:ss), and period (AM or PM) when the profile analysis was initialized.
 - **Data Source:** The source of the data in the profile being analyzed. It can be a physical or logical model or a direct data source connection.
 - **User:** The user who created the profile
 - **Tables:** Number of tables in the selected model
 - **Data source type:** Whether the data source for the profile is a model or a database
 - **Records:** Total number of records analyzed
 - **Statistics** Button: Navigates you to the **Data Profiling Results** page. For more information, see [Viewing Suggestions on the Data Profiling Results Page](#) on page 237.
- The second section displays the name of your table along with a **Recommendation** button. On clicking this button, suggestion-wise **Groovy Scripts** for your table are displayed. On expanding the table through the  icon, you can view column-wise suggestions for your table.

Details such as **Suggestions** and their corresponding **Descriptions** are displayed in a tabular format for various columns on which suggestive actions were generated. The **View Details** button displays


a preview of statistics and options for your data, you can choose to select any suggestion here by clicking the **Select Suggestion** button or remove an already applied suggestion by clicking the


Remove Suggestion button.  icon signifies successful selection of a suggestion. You can choose to view and copy the groovy script for any particular suggestion by clicking the corresponding **Recommendation** button.

Note: To download and copy the **Groovy Script** for all the columns of a particular table for which suggestions are selected, click the **Recommendation** button placed adjacent to the table name. On clicking this button, a **Recommendation** window is displayed, this window displays details such as the table name, column name, and suggestion generated followed by the groovy script.

You can view and select suggestions for independent statistics from the **Data Profiling Results** page. Examine the statistics of your data and select any suggestion according to your requirements. For more information, see [Viewing Suggestions on the Data Profiling Results Page](#) on page 237.

Viewing Suggestions on the Data Profiling Results Page

You can view the suggestions provided to cleanse your data. These suggestions are generated based on the statistics of your data. You can choose to **select** these. Click the **Suggestions** button, a new window will pop-up showing suggestions. You can move to the next suggestion by clicking the  icon.

Note: To view the transformed data after selecting an option for any suggestion, click the  icon. If you wish to select the suggestion, click the **Select Suggestion** button, a message will be displayed on successful selection of the suggestion.

Suggestions for Summary Tab

For **Character Categories**, you will view any of these **suggestions** based on your data:

- **Character Data Type:** Character data types such as *numeric*, *alphanumeric*, and *alphabetic* detected in your data are displayed in a tabular format with their frequencies. From the list of options, you can select to **Trim** or **Eliminate** these values, or **set to a default value**. You have to specify this value in the text box which appears on selecting this option.
- **Remove Spaces:** Redundant spaces such as a **Single Space**, **Multiple Spaces**, and **Trailing or leading Spaces** present in your data are displayed in a tabular format with their frequencies. You can choose any of these options according to your requirement:
 - **Trim Spaces** - This option will remove any trailing or leading spaces present in your data.
 - **Minimize white spaces** - This option will remove redundant spaces between any two words and also remove any leading or trailing spaces
 - **Remove spaces** - This option will remove all spaces present in your data.
- **Standardize Casing:** You can regularize your data by standardizing the casing of your data such as **Mixed Case**, **Upper Case**, and **Lower Case** through this option. You can choose to change

the casing of your data to **Lower case** or **Upper case**, this will cleanse your data, improve its uniformity, and readability.

- **Remove Special Characters:** You can choose to remove any redundant special characters present in your data through this option. The detected special characters are displayed under the **Options** section in a text box. To cleanse your data, you can select the special characters to be removed and delete the characters which you want to keep using the *backspace* key from the text box.
- **Remove Control Characters:** You can remove any redundant non-printable control characters present in your data through this option. The control characters detected in your data are displayed under the **Statistics** section, select the **Remove Control Characters** option and apply this suggestion to further cleanse your data.

For **Uniqueness**, you will see this suggestion:

Create Semantic Type: You can categorize your distinct data into a semantic type such as *Gender*, *First Name*, *Last Name*, *Country*, and *Phone*. Click on the statistics graph of the distinct data, a preview of your data is shown. Download this data and create semantic types through *Table Management*. For more information about Table Management, see [Introduction to Lookup Tables](#).

For **Completeness**, you will see this suggestion:

Handling Nulls: You can set the *null* and *empty* strings present in your data to a default value. This increases the uniformity and completeness of your data. The frequencies of **Complete**, **Null**, and **Empty** strings are displayed in a tabular format under the **Statistics** section. Enter the desired default value in the **Options** text-box and select this suggestion.

Suggestions for Date Summary Tab

For **Date Patterns**, you will see this suggestion:

Standardize Date: You can regulate the formats of dates present in your data to improve the consistency of your data. Frequencies of various date formats detected in your data are displayed in a tabular format. Select the desired format from the *date format* drop-down placed under the **Options** section and select this suggestion.

Suggestions for Custom Pattern Tab

For **Matched Patterns**, you will see this suggestion:

Pattern Cleansing: You can eliminate values from your data not following the custom pattern specified by you while configuring a profile. For more information about custom patterns, see [Custom Pattern Analysis](#) on page 220. The category with the highest frequency is treated as reference, other statistics are matched to it. You can select any of these options for non-matching statistics:

- Set them to a default value. Enter the default value in the text-box provided.
- Set them to null

After selecting the desired option, select this suggestion to cleanse your data.

Suggestions for Outlier Tab

For **semantic type outliers**, you will see this suggestion:

Segregating Semantic Types: You can move any semantic type outliers present in a column to a separate column and improve the uniformity and clarity of your data. The frequency of various outliers detected in your data are displayed in a tabular format, you can choose to move any of the detected outliers to a separate custom column and segregate the semantic types.


Select the outliers which you wish to move to a separate column, enter the custom name of the column in the text box and select this suggestion.

Note: On closing the *suggestions* window after the successful selection of any suggestion, a **Show Recommendations** prompt is displayed. On clicking **Yes**, you will be navigated to the **Suggestions** page. You can copy the *groovy script* from this page and use in the **Enterprise Designer**. For more information, see [Viewing the Suggestions Page](#) on page 236.



Collaborating on Data Profiling Results

You can do real-time collaboration on your **Data Profiling Results** by using the **Comments** button on the **Data Profiling Results** page. This feature is available at the Model, Table, and Column level and multiple users can see and build on each other's analysis of the profile results.

To add comments and collaborate on your **Data Profiling Results**, follow these steps:

1. On the **Data Profiling Results** page, click the **Comments** icon  on the right panel. A **Comments** text box is displayed.
2. Add your findings on the displayed results to this text-box and click the **Save** button. Alternatively, you can press **Alt + Enter** keys.
The entered comment is saved and becomes visible to other users.

Note: Click the **Cancel** button if you do not want to save your comment.

3. To tag your comment to any section of the report, use the # symbol followed by the section name. For example, at table level display, to tag your findings to the **Completeness** and **Null Count Frequency** of records, comment it as: *#Completeness <your finding> #Null Count Frequency <your finding>*
4. To edit your comment, click the edit  icon.
5. To delete your comment, click the delete  icon.

How to Video - Profiling

This video describes how to profile your data and identify data quality issues in it.

<https://www.youtube.com/embed/n65csBzT1kQ>

How to Video - Applying Suggestions to Profiled Data

This video describes how to apply the automatically generated suggestions to your profiled data to make the data fit for purpose.

<https://www.youtube.com/embed/l2Q6TB5OP9c>

How to Video - Scorecard

This video explains how you can Measure and Monitor Data Quality.

<https://www.youtube.com/embed/65Q7Kp8QGxw>

How to Video - Consuming Spectrum Services in Profiling and Scorecard

This video explains how you can consume Spectrum Services in Profiling and Scorecarding to evaluate Data Quality Rules.

<https://www.youtube.com/embed/XLXtJedTbNU>

9 - Lineage and Impact Analysis

In this section

Viewing Lineage and Impact Analysis.....	242
Icons in Lineage and Impact Analysis.....	243
Uses.....	248
How to Video - Lineage and Impact Analysis.....	255



Viewing Lineage and Impact Analysis

This procedure shows how to trace the flow of data through lineage and impact analysis.


The Lineage and Impact Analysis view shows how data flows from data sources to data destinations and through Spectrum Technology Platform flows. Lineage and impact analysis are similar concepts that describe different ways of tracing the flow of data.



Lineage shows where data comes from. You can use it to trace the path of data back to its source, showing all the systems that process and store the data along the way, such as Spectrum Technology Platform flows, databases, and files.

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Log in.
3. Click **Lineage & Impact Analysis**.
4. Click the Select Entity button .
5. Select the type of entity you want to analyze. For example, if you want to view lineage or impact analysis for a dataflow, select **Dataflows**.
6. Select the specific entity whose data you want to analyze.
7. Click **OK**.

The resulting diagram shows the flow of data into and out of the entity you selected. The entity's lineage is to the left of the entity. The entity's impact is to the right. You can change the diagram to show only the entity's lineage  or only the entity's impact .

Some entities are expandable, allowing you to view the flow of data within the entity, such as the specific tables that the data flows into in a database, or the stages in a Spectrum Technology Platform flow. To expand an entity, click the tab in the lower-right corner of the entity's icon:



Once expanded, you can see the details of the flow of data through the entity. For example, here we have expanded the database entity named Prod SQL Server. You can see that data in the table Customer Table comes from the Spectrum Technology Platform flow named Add to Customer DB. In addition, you can see that data from Customer Table is used by two flows: Query Customer Job

and Geocode Job. Note as well that the flow Geocode Job writes data back to Customer Table, so in this example Customer Table is both an input to and an output from Geocode Job. This example could represent a situation where Geocode Job reads customer addresses from Customer Table, determines the latitude and longitude coordinates for the addresses, then adds the latitude and longitude fields to the customer records in Customer Table.







Related concepts


[A First Look at Spectrum Discovery](#) on page 6

Icons in Lineage and Impact Analysis


The Lineage & Impact Analysis canvas in Spectrum Discovery contains a variety of icons that represent different parts of Spectrum Technology Platform as well as the data resources connected to Spectrum Technology Platform.

Dataflows and Stages


Icon	Entity	Lineage	Impact Analysis
	Dataflow type Expand this icon to see the exposed and last saved versions of the dataflow.	The lineage of a dataflow type shows the data sources that feed into any version of the dataflow (either exposed or last saved).	The impact analysis of a dataflow type shows the destinations where any version of the dataflow writes data. If any version of the dataflow is an exposed job, the impact analysis also shows an activity type icon which represents the fact that the dataflow may be used as an activity in a process flow.
	Dataflow version (exposed or last saved) Expand this icon to see the stages in the dataflow.	The lineage of a dataflow version shows the data sources that feed into the version. It also shows subflows contained in the dataflow.	The impact analysis of a dataflow version shows the destinations where the selected version of the dataflow writes data. If the dataflow is an exposed job, the impact analysis also shows an activity type icon which represents the fact that the dataflow may be used as an activity in a process flow.
	Stage type	Stage types have lineage if the stage represents a subflow. In this case the subflow shows as lineage of the stage type. For other stage types, there is no lineage.	The impact analysis of a stage type shows all the flows that use the stage. For example, if you are looking at the impact analysis of the Read from File stage type, the impact analysis shows all the flows that contain a Read from File stage.
	Stage instance	The lineage of a stage instance shows data used by the stage, and also the stage type.	The impact analysis of a stage instance shows where the data goes to after being processed by the stage. This could be another stage in a dataflow, a database, a file, or any other place where the data is sent.

Icon	Entity	Lineage	Impact Analysis
	<p>Default service options</p> <p>This entity represents the default options for the selected service. Default options are those configured in Management Console.</p>	<p>The lineage of default service options shows entities that the service depends on by default. For example, if a service has a default option that specifies a Spectrum database such as a postal database or a geocoding database, that Spectrum database is shown in the default service option's lineage.</p>	<p>The impact analysis of default service options shows the stages that are configured to use the default service options rather than overriding them in the stage.</p>



Data Sources and Destinations

Icon	Entity	Lineage	Impact Analysis
	<p>Connection</p>	<p>The lineage of a connection shows the entities that write data to the connection.</p>	<p>The impact of a connection shows the entities that read data from the connection.</p>



Spectrum Databases



Icon	Entity	Lineage	Impact Analysis
	<p>Spectrum database</p>	<p>Spectrum databases have no lineage.</p>	<p>The impact analysis of Spectrum databases shows the stages that use the database. It also shows when the database is configured as a default service option.</p>

Reports





Icon	Entity	Lineage	Impact Analysis
	Report type	Report types have no lineage.	The impact analysis of a report type shows the flows where the report type is used. For example, if you are looking at the impact analysis of the ValidateAddressSummary report type, the impact analysis shows all the flows that generate a ValidateAddressSummary report.
	Report instance	The lineage of a report instance shows the report type.	Report instances have no impact.

Process Flows and Activities

Icon	Entity	Lineage	Impact Analysis
	Process flow type Expand this icon to see the exposed and last saved versions of the process flow.	The lineage of a process flow type shows the activities, such as dataflows, that are called by any version of the process flow, and the data sources that feed into those activities. The output from the activities is also shown in the lineage of the process flow, not in the impact.	Process flow types have no impact since any data produced by a result of running a process flow is shown in the lineage, where the called activities are shown along with the output from those activities.
	Process flow version Expand this icon to see the activities in the process flow.	The lineage of a process flow version shows the activities, such as dataflows, called by the selected version of the process flow, and the data sources that feed into the version.	Process flow versions have no impact since any data produced by a result of running a process flow is shown in the lineage, where the called activities are shown along with the output from those activities.

Icon	Entity	Lineage	Impact Analysis
	Activity type An activity type represents a flow that can be used inside a process flow.	The lineage of an activity type shows the dataflow that the activity represents, and the data sources for the dataflow.	The impact of an activity type shows the process flows that contain the activity. If an activity type has no impact, it is not used in any process flow.
	Activity instance	The lineage of an activity instance is the activity type and the dataflow represented by the activity type. If the activity is preceded in a process flow by another activity, that activity is also shown in the activity's lineage.	The impact of an activity is the activity that follows it in a process flow.

Modeling

Icon	Entity	Lineage	Impact Analysis
	Model store	The lineage of a model store shows the physical and logical models that comprise the model store.	The impact of a model store shows the connections to the model store and the flows that use the model store.
	Physical model	The lineage of a physical model shows the connections that supply data to the physical model, such as database connections, file server connections, or cloud connections.	The impact of a physical model shows the logical models that are mapped to the physical model. The impact also shows the model stores that contain the model.
	Logical model	The lineage of a logical model shows the physical model that is mapped to entities in the logical model.	The impact of a logical model shows the model stores that contain the model.
	Entity in a physical or logical model, or relationship in a logical model	Inside a physical model, entities do not have lineage. Inside a logical model, the lineage of an entity is the relationship between the entity and other entities	Inside a physical model, entities do not have impact. Inside a logical model, relationships have impact on the entities that they connect.

Uses

Modifying a Database Table


When you make a change to a database, that change can have unintended consequences for systems that rely on its data. Modifications to schemas, keys, columns, or tables can affect downstream processes and ultimately reports in ways that are sometimes obvious and sometimes subtle. So when planning a database modification, you want to be confident that you understand the downstream impact of your changes so that you can modify other systems if needed to ensure your business users continue to have accurate and reliable data on which to base business decisions.

Spectrum Technology Platform Spectrum Discovery provides a visual representation of the flow of data throughout your flows, databases, and files. You can select the database you are modifying and see the flows, databases and files that use the data. With this information you can determine the impact of database modifications. For more information, see [Viewing Lineage and Impact Analysis](#) on page 242.

1. In a web browser, go to:

```
http://server:port/discovery
```

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Click the Select Entity button .
3. Click **Database Tables**.
4. Select the table that you are considering modifying then click **OK**.

The resulting diagram shows the connection that contains the table. To the left of the connection (the table's lineage) is any entity that writes data to the table. To the right of the table are any entities that use data from the table, such as flows.

Example

You are working on a project to standardize column names across several databases. You have found a column the table `Customer_Data` that you want to rename. Before modifying the column name, you want to find out which entities are using data from this table so that you can analyze them to find out if they must be updated to accommodate the new column name.

To accomplish this, you open Spectrum Discovery, go to Lineage & Impact Analysis, and open the entity browser. You click **Connections**, then the name of the

connection for the database containing the table. Then you select the table and click **OK**. In the resulting diagram, you expand the connection icon containing the table, which shows you which flows are connected to the table:



The diagram reveals that the HouseholdRelationships flow writes data to the Customer_Data table, and also reads data from it. In addition, the Promotions flow reads data from the Customer_Data table. As a result of reviewing this diagram, you can open the HouseholdRelationships and Promotions flows in Enterprise Designer to find out if there are any changes you need to make to accommodate the renaming of columns in the Customer_Data table.

Related tasks

[Updating a Spectrum Database](#) on page 249

[Changing a Service's Default Options](#) on page 252

[Modifying a Subflow](#) on page 254

Updating a Spectrum Database

Spectrum databases contain reference data that some modules use to validate or enhance your data. Examples of Spectrum databases include the postal database used by Spectrum Universal Addressing and the geocoding databases used by Spectrum Geocoding modules. Spectrum databases need to be updated on a regular basis to ensure that you have the latest reference data. When it comes time to update a Spectrum database, you want to be confident that the update will go smoothly and not disrupt your operations.

To accomplish a smooth update, you need to know which processes are using the old database. If you know which processes are using the old database, you can:

- Update them to use the new database
- Test them with the new database to ensure you continue to get the expected results


The Lineage & Impact Analysis tool provides the visibility you need to achieve a smooth update of a Spectrum database. It shows the services and flows that use a Spectrum database so that you

can update them to point to the new database and test them to ensure that they continue to produce the expected results after the update.

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Click **Lineage and Impact Analysis**.
3. Click the Select Entity button .
4. Click **Spectrum Databases**.
5. Click the button next to the module name to select the module then click **OK**. This will display all the Spectrum databases for the selected module.

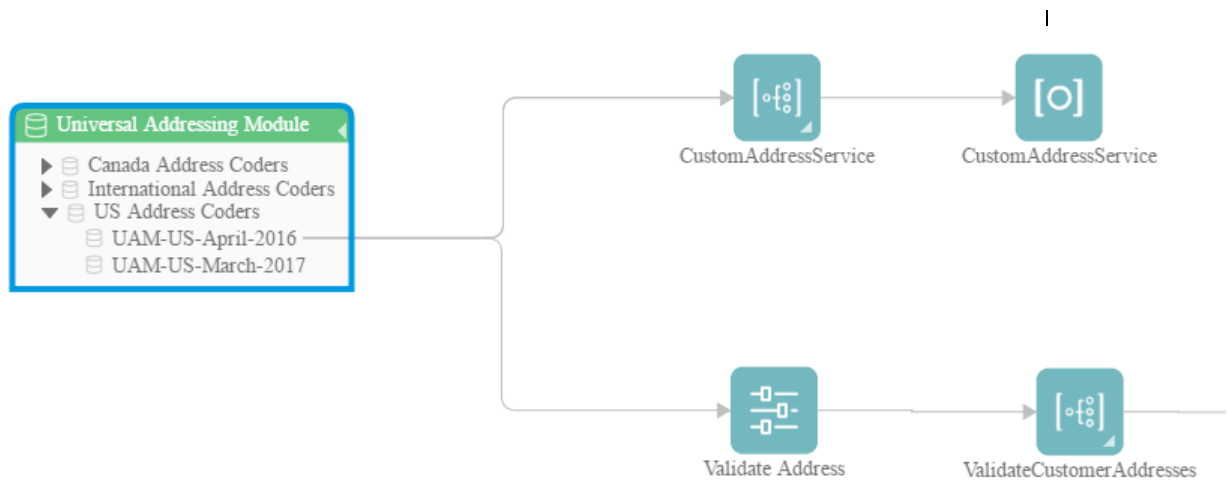
Tip: Even though you can click the module name itself to view the individual databases that are used by the module, we recommend viewing all the module's databases first so that you have a complete picture of all the Spectrum databases in use for the module.

The resulting diagram shows the entities that rely on each Spectrum database for the module you selected.

Example

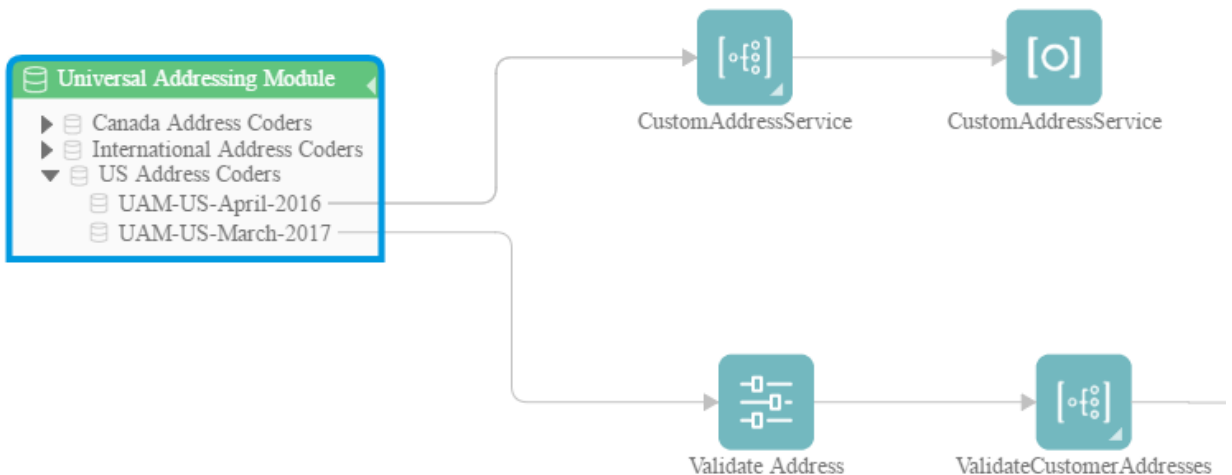
You use Spectrum Universal Addressing to validate addresses in your customer database. You are currently using outdated postal database and want to update your system to use the latest database. You have installed the database and defined a Spectrum database for it in Management Console. Now you need to find out where the old database is used so that you can update everything to use the database, and test all affected processes to ensure they are producing the expected results using the new database.

To do this, open Spectrum Discovery, navigate to **Lineage & Impact Analysis**, and open the entity browser. Click **Spectrum Databases**. Since you want to view all database resources for Spectrum Universal Addressing, you select the button next to **Universal Addressing Module** and click **OK**. The resulting diagram is:

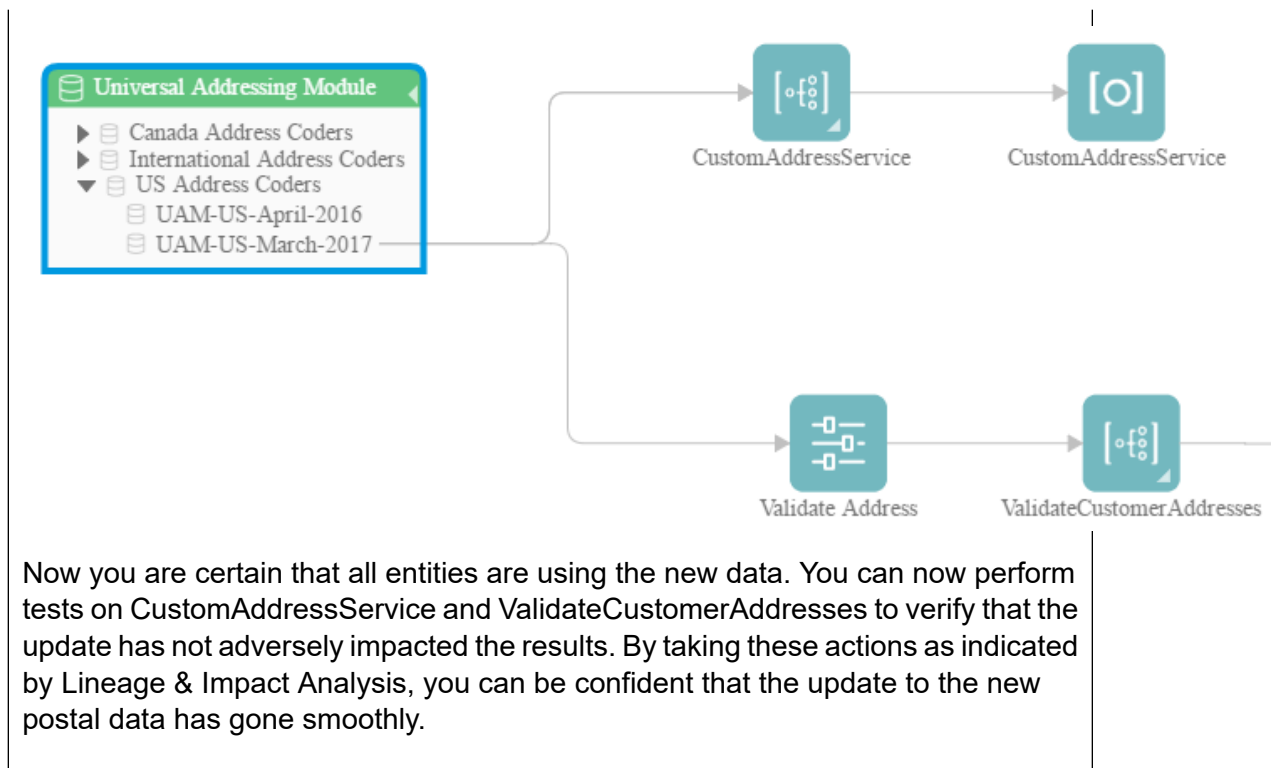


The diagram shows that there are two Spectrum databases on the system, the old one (UAM-US-April-2016) and the new one that you just created in Management Console (UAM-US-March-2017). There are two entities using the old database: a flow named CustomAddressService and a flow named ValidateCustomerAddresses. There are no entities using the new database.

You update the default service options for Validate Address to use the new database. After refreshing the diagram, you now see this:



As you can see, the default service options for Validate Address are now using the new database (UAM-US-March-2017) but the flow CustomAddressService is still using the old database (UAM-US-April-2016). You open the flow CustomAddressService in Enterprise Designer and change the Validate Address stage to use the new database. When you refresh the diagram you now see this:



Related tasks

[Modifying a Database Table](#) on page 248

[Changing a Service's Default Options](#) on page 252

[Modifying a Subflow](#) on page 254

Changing a Service's Default Options

Spectrum Technology Platform services have default options which you configure in Management Console. The default options are in effect when a service is used as a stage in a flow and the stage is configured to use the system's default options. So, before you make changes to service options in Management Console, you need to know which flows use the service's default options so that you can ensure that the change does not have undesirable consequences for those flows, or for any business processes that rely on the flows.

To find the flows that use a service's default options:

1. In a web browser, go to:

`http://server:port/discovery`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

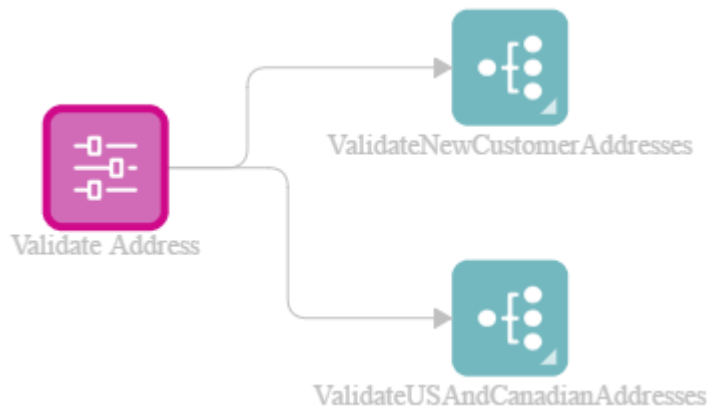
2. Click the Select Entity button .

3. Click **Default Service Options**.
4. Select the service whose default option you are considering modifying then click **OK**.

The resulting diagram shows all the flows that use the service's default options. In other words, these are the flows where the stage option **Override system default options with the following values** is *not* checked. If a flow contains the service, but has this option checked, it is not included in the diagram because it is not using the service's default options, and so would not be impacted by changes to the service's default options in Management Console.

Example

You have configured the service Validate Address to return addresses in upper case and you want to change the casing option to mixed case in Management Console, which means that all Validate Address stages that use the service's default options will start returning addresses in mixed case. To ensure that this change does not have undesirable consequences for these flows, you use the Lineage & Impact Analysis view in Metadata Insight to find out which flows use the Validate Address default options. It shows this diagram:



The diagram reveals that the flows ValidateNewCustomerAddresses and ValidateUSAndCanadianAddresses are both using the default options for the Validate Address service. Since any changes you make to Validate Address's options in Management Console will affect these two flows, you open them in Enterprise Designer to examine the impact of the change on how the flow handles addresses. If you find that the change to mixed case causes problems, you could adjust the flow to accommodate the change.

Related tasks

[Modifying a Database Table](#) on page 248

[Updating a Spectrum Database](#) on page 249

[Modifying a Subflow](#) on page 254

Modifying a Subflow


A subflow is a flow that can be reused within other flows. Subflows are useful if you want to create a reusable process that can be easily incorporated into multiple flows. When subflows are used extensively, it can be difficult to keep track of which flows use each subflow. This can be a problem if you want to modify a subflow, because you need to make sure that the modification does not cause undesirable results in the flows that use it. Using the Lineage & Impact Analysis feature in Spectrum Discovery, you can find all the flows that use a particular subflow so that you can be fully aware of the impact of changes you make to a subflow.

To view the flows that use a subflow:

1. In a web browser, go to:

`http://server:port/discovery`

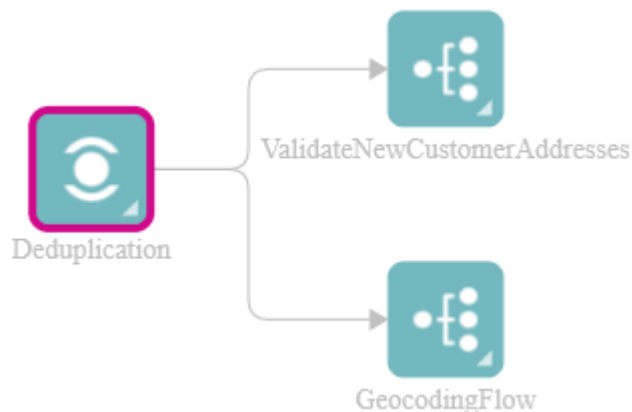
Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Click the Select Entity button .
3. Click **Stages**.
4. Select the subflow you want then click **OK**.

The resulting diagram shows all the flows that use the subflow.

Example

You have a subflow named Deduplication. It contains logic for identifying duplicate records in your customer data. You use this subflow in multiple flows. You want to make a change to the deduplication logic and need to make sure that you test the effects of the change in all the flows that use the subflow. So, you use Lineage & Impact Analysis to view the impact of the Deduplication subflow:



The Lineage & Impact Analysis diagram shows that the Deduplication subflow is used in two flows: ValidateNewCustomerAddresses and GeocodingFlow. When you make your change you can check these flows in Enterprise Designer to confirm that you are getting the results you want.

Related tasks

Modifying a Database Table on page 248

Updating a Spectrum Database on page 249

Changing a Service's Default Options on page 252

How to Video - Lineage and Impact Analysis

This video discusses the process of tracking the journey of data from its origin to destinations and how it has been flowing through the Spectrum Technology Platform.

<https://www.youtube.com/embed/AZLqscD9hE4>

10 - Accessing Services through API

In this section

API calls for Glossary and Profiling services	257
How to Video - Discovery APIs.....	258



API calls for Glossary and Profiling services

The **Business Glossary** and **Profiling** modules of **Spectrum Discovery** are also available as APIs , enabling you to use these features independent of Spectrum license.

Use this URL to access the APIs:

`http://server:port/discoveryapi`

Where *server* is the server name or IP address of your Spectrum Technology Platform server and *port* is the HTTP port. By default, the HTTP port is 8080.

The features available to you are:

1. Business Glossary
 - addGlossaryEntity
 - deleteGlossaryEntity
 - getGlossaryEntity
 - listGlossaryEntities
 - updateGlossaryEntity
2. Profiling
 - getColumnStatisticsPreview
 - getProfileColumnStatistics
 - getProfileConfiguration
 - getProfileModelStatistics
 - getProfileTableStatistics
 - getProfiles
 - getScorecardConfiguration
 - getScorecardPreview
 - getScorecardStatistics
 - getScorecards
 - getTableStatisticsPreview

For information about how to use the features, click the respective **Documentation** link.

API details for: glossary

- OpenAPI specification:
 - [openapi.json](#)
 - [openapi.yaml](#)
- [Documentation](#)
- [Swagger UI](#)

API details for: profiling

- OpenAPI specification:
 - [openapi.json](#)
 - [openapi.yaml](#)
- [Documentation](#)
- [Swagger UI](#)

How to Video - Discovery APIs

This video explains how you can access Discovery API's and consume them within your technological system.

<https://www.youtube.com/embed/KhNzRxeILP0>



2 Blue Hill Plaza, #1563
Pearl River, NY 10965
USA

www.precisely.com

© 2007, 2021 Precisely. All rights reserved.