
metatron-doc-user Documentation

Release 0.4.3

metatron team

May 12, 2020

METATRON DISCOVERY

I Metatron Discovery	1
1 Discovery Quick Guide	3
2 Introduction of Metatron Discovery	37
3 Data Management	77
4 Workspace	117
5 Workbook	131
6 Notebook	211
7 Workbench	225
8 Data Preparation	239
9 Account management	353
10 Data Exploration	365
11 Engine Monitoring	377
II EX-pack for Workflow Integrator	387
12 Introduction of Integrator Expansion Pack	389
13 Workflow list	391
14 Workflow editor	393
15 Monitoring	401
16 Use cases	403

III EX-pack for Anomaly Detection	405
17 Introduction of Metatron Anomaly	407
18 Statistics	411
19 Alarm	413
20 Alarm Rule	421
21 Algorithm	445
22 Dashboard	449
23 Search	459

Part I

Metatron Discovery

CHAPTER
ONE

DISCOVERY QUICK GUIDE

Metatron Discovery is an all-in-one solution that enables rapid loading, pre-processing, and analysis of large amounts all together. With Metatron Discovery, business users without technical knowledge can directly work with data and gain insights from rapid visualization.

You can perform data analysis with Metatron Discovery using the two methods below:

- **Method 1:** Run [Metatron Discovery demo site](#). Enter “metatron” as your ID and password.
- **Method 2:** Download the single-mode Metatron Discovery to your local PC. [Download](#) is provided in three ways.
 - **Custom install:** Download the source code from the Github repository, or directly run the build file.
 - **Virtual machine:** Run the virtual machine image. This is also available in the Windows OS.
 - **Docker:** Run the Docker image for a quick installation.

Do you see the screen below? Congratulations! You are now ready for quick and easy data analysis with Metatron Discovery.



For a quick start, follow the three-step tutorial below:

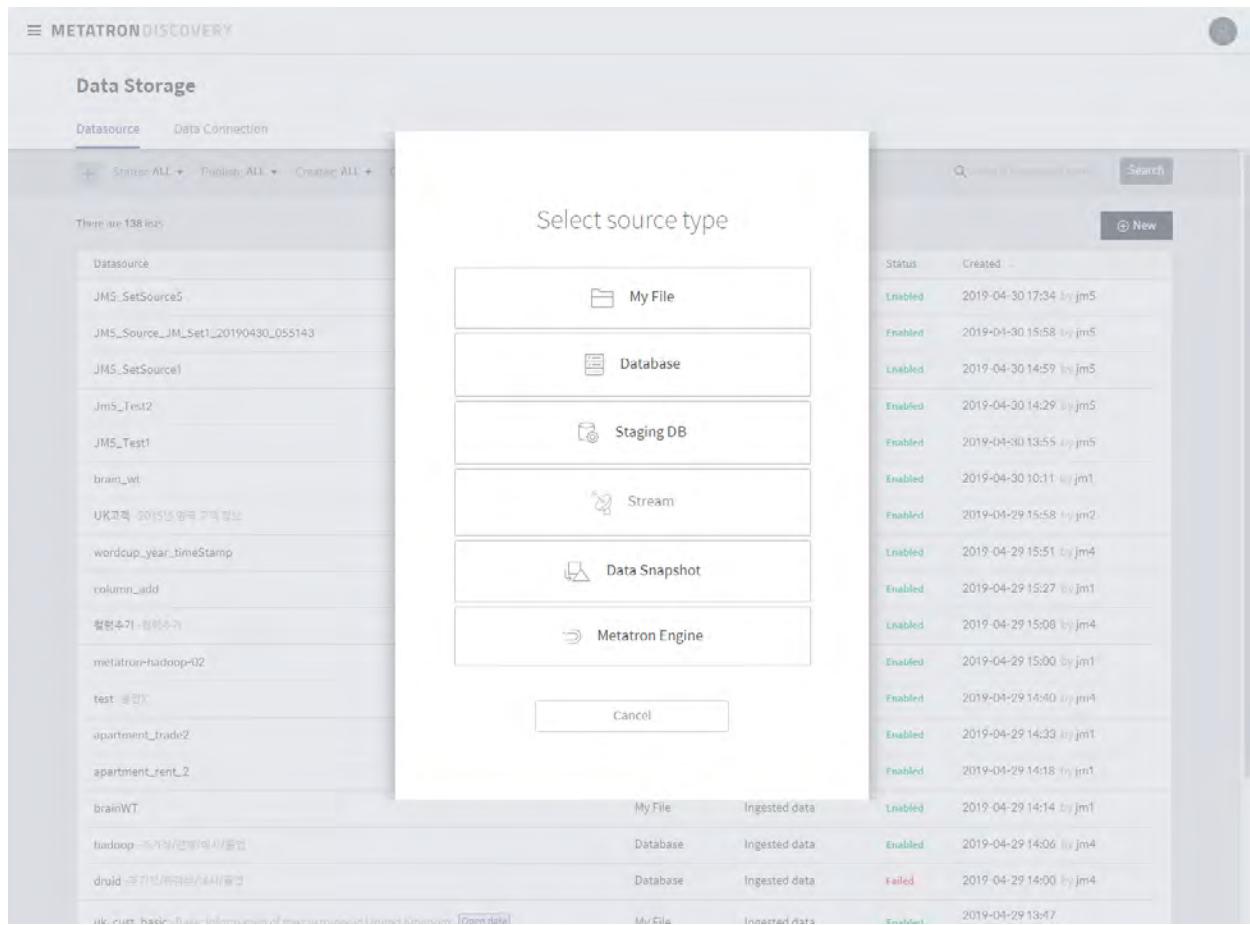
1.1 Step 1. Create a data source

The first step in data analysis is ingesting your data into the system. Metatron Discovery allows you to easily ingest various data sources.

The example in this tutorial shows you how to ingest data from your local directory. First, prepare data. An Excel file (.xls, .xlsx) or .csv file will suffice. This tutorial uses sales data. Download it from the link below:

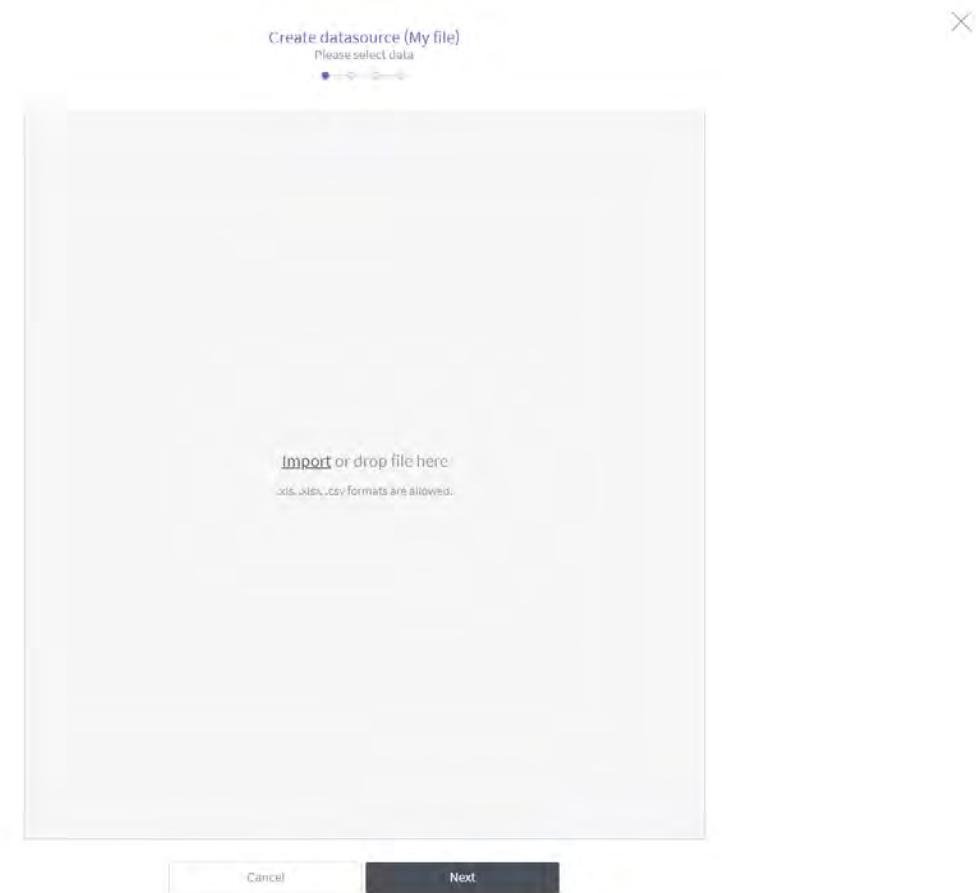
sample data (.csv)

Data sources can be viewed and ingested from **Management** > **Data Storage** > **Data Source**. To create a new data source, click the **New** button on the upper right of the data source list.



In this tutorial, click **File** to retrieve the data from your local directory. See [Create a data source](#) for details on creating a data source from other sources.

Drag and drop the data you wish to analyze, or retrieve it from the directory.



Drag your cursor over the sales data to view up to 100 rows of data with detection of the column delimiter and line separator. This data is properly displayed using the default delimiter and separator. Click **Next**.

Create datasource (My file)
Please select data

sales-data-sample.csv Import or drop file here

ab OrderDate	ab Category	ab City	ab Country	ab CustomerName	ab Discount	ab OrderID	ab Pos
2011-01-04T00:...	Office Supplies	Houston	United States	Darren Powers	0.2	CA-2011-103...	770
2011-01-05T00:...	Office Supplies	Naperville	United States	Phillina Ober	0.2	CA-2011-112...	605
2011-01-05T00:...	Office Supplies	Naperville	United States	Phillina Ober	0.8	CA-2011-112...	605
2011-01-05T00:...	Office Supplies	Naperville	United States	Phillina Ober	0.2	CA-2011-112...	605
2011-01-06T00:...	Office Supplies	Philadelphia	United States	Mick Brown	0.2	CA-2011-141...	191
2011-01-07T00:...	Furniture	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:...	Office supplies	Athens	United States	Jack O'Briant	0.0	CA-2011-106...	306
2011-01-07T00:...	Office Supplies	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:...	Office Supplies	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:...	Office Supplies	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:...	Office Supplies	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:...	Office Supplies	Los Angeles	United States	Lycoris Saunders	0.0	CA-2011-130...	900
2011-01-07T00:...	Technology	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:...	Technology	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-08T00:...	Furniture	Huntsville	United States	Vivek Sundaresam	0.6	CA-2011-105...	773
2011-01-08T00:...	Office Supplies	Huntsville	United States	Vivek Sundaresam	0.0	CA-2011-105...	773

Column delimiter: ,
Line separator: \n
 Use the first row as the head column. (If not checked, a new row is created and is used as the head column)

While viewing the data, adjust the column types properly. This task is called **data schema configuration**.

The screenshot shows the 'Create datasource (My file)' interface. On the left, a list of columns is displayed, each with a 'Dimension' or 'Measure' role and a 'Type' of 'String'. The 'OrderDate' column is selected and shown in detail on the right. The 'Setting' panel for 'OrderDate' includes fields for 'Role' (set to 'Dimension'), 'Type' (set to 'String'), and 'Missing' (set to 'Do not apply'). A note at the bottom states: 'One of the time-type columns or current time must be specified as a Timestamp'.

Column	Type	Setting
OrderDate	Dimension	
Category	Dimension	
City	Dimension	
Country	Dimension	
CustomerName	Dimension	
Discount	Dimension	
OrderId	Dimension	
PostalCode	Dimension	
ProductName	Dimension	
Profit	Dimension	
Quantity	Dimension	
Region	Dimension	
Sales	Dimension	
Segment	Dimension	
ShipDate	Dimension	
ShipMode	Dimension	
State	Dimension	
Sub_Category	Dimension	
DaystoShipActual	Dimension	
SalesForecast	Dimension	
ShipStatus	Dimension	
DaystoShipScheduled	Dimension	
OrderProfitable	Dimension	
SalesperCustomer	Dimension	

OrderDate

Data (50 Row) Setting

Role: Dimension

Type: String

Missing: Do not apply

Note: One of the time-type columns or current time must be specified as a Timestamp

Current time: No selected time-type column

Previous Next

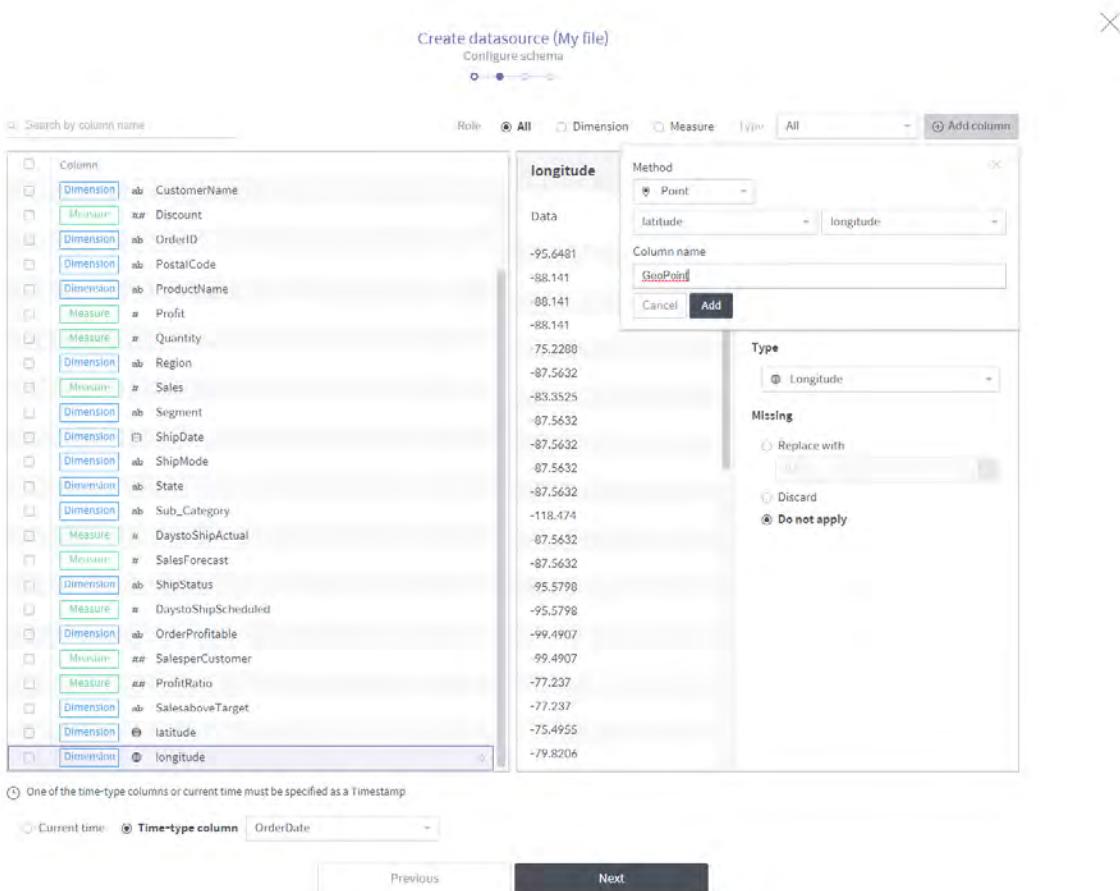
Each column functions as a “dimension” or “measure.” See “Dimensions” and “Measures” for further details. In this data, the Discount, Profit, Quantity, Sales, DaystoShipActual, SalesForecast, DaystoShipScheduled, SalesperCustomer, and ProfitRatio columns must be converted into measures.

Next, the data types of columns must be adjusted properly. The string type is the default setting for dimensions, and the integer type for measures. While viewing the sample, change the data type settings properly. Below is a list of items to be modified in this data.

- Orderdate : Date/Time
- Discount : Decimal
- ShipDate : Date/Time (Change the time format to yyyy. MM. dd. and click the checkbox to validate)

- SalesperCustomer : Decimal
- ProfitRatio : Decimal
- latitude : Latitude
- longitude : Longitude

Lastly, you should create a new column. Since we already have columns for latitude and longitude, we can create a point type column. Click the **Add column** button on the upper right. Select the latitude column for the **Latitude** column, and the longitude column for the **Longitude** column. Name the columns appropriately, and click **Add**. A new point type column is created!



Once you are done with schema configuration, click **Next**. If necessary, you can change the settings for ingestion into Druid. The default settings are sufficient for now.

Create datasource (My file)
Please complete ingestion settings

Timestamp settings

Query Granularity ⓘ

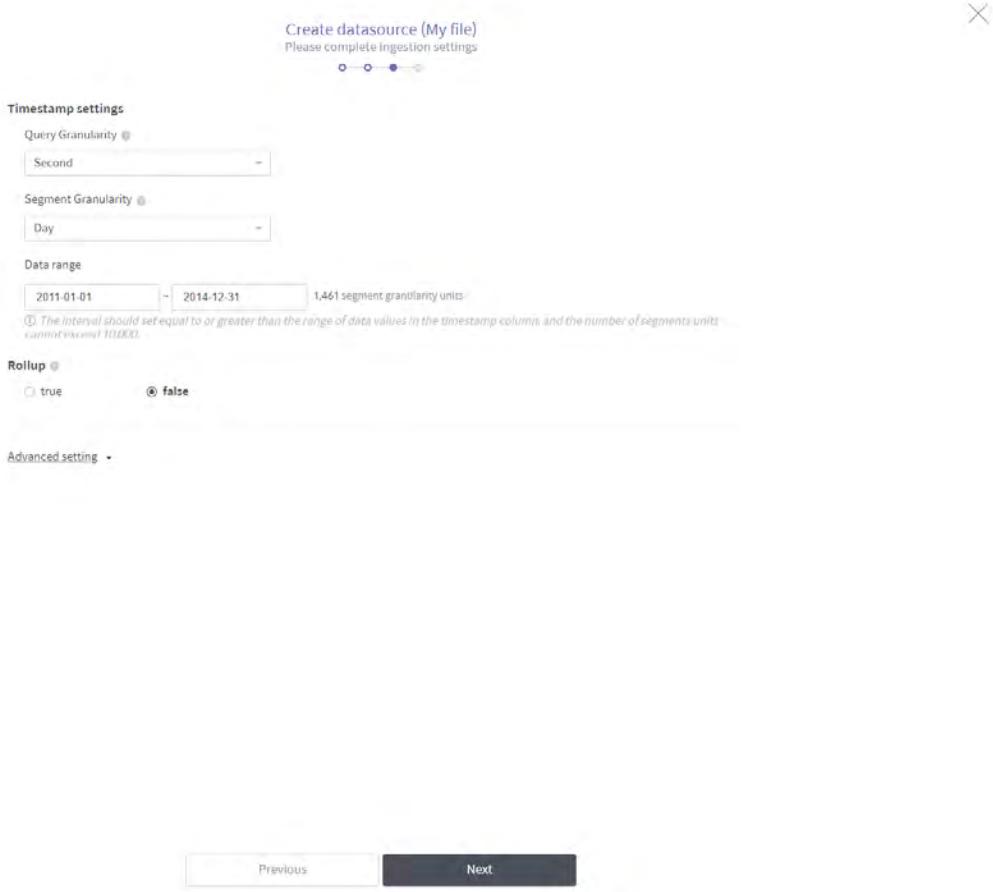
Segment Granularity ⓘ

Data range
 ~ 1,461 segment granularity units
The interval should set equal to or greater than the range of data values in the timestamp column, and the number of segments units cannot exceed 1000.

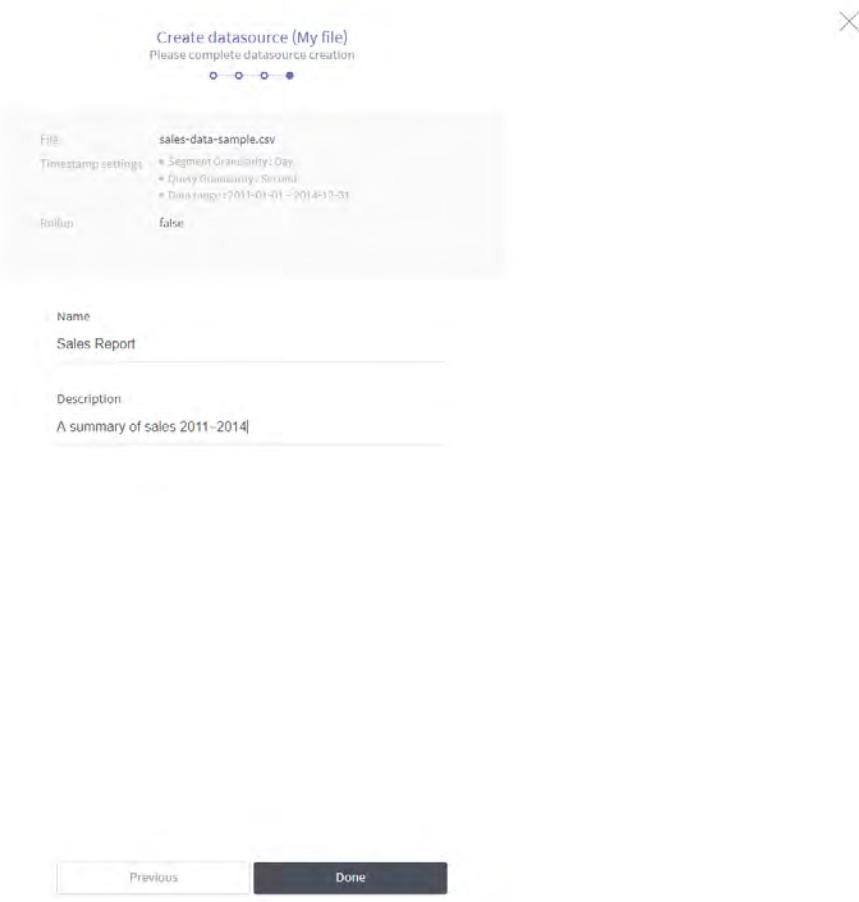
Rollup ⓘ
 true false

Advanced setting ▾

[Previous](#) [Next](#)



Lastly, enter the **Name** and **Description** for the data source. Click **Done** to proceed to the data source details page.



In the data source details page, you can view the ingestion status in real time. The screen below appears after a few minutes, indicating success. A histogram is displayed. If you encounter an error while ingesting another data source, click **Details** to view the Druid ingestion log. Ingestion may be unsuccessful due to a duplicate column name or mismatch between column types and their data. Try ingestion again after addressing the issue.

Sales Report

updated on 2019-05-06 12:15 | Administrator

Information Data Column details Monitoring

Data Information

Description A summary of sales 2011-2014

Go to Metadata

Ingestion type: Ingested data
State: UNABLED

1 Preparing data 2 Registering on engine 3 Checking validity 4 Success

Timestamp settings: Query Granularity: SECOND
Segment Granularity: DAY
Data range: 2011-01-01 ~ 2014-12-31

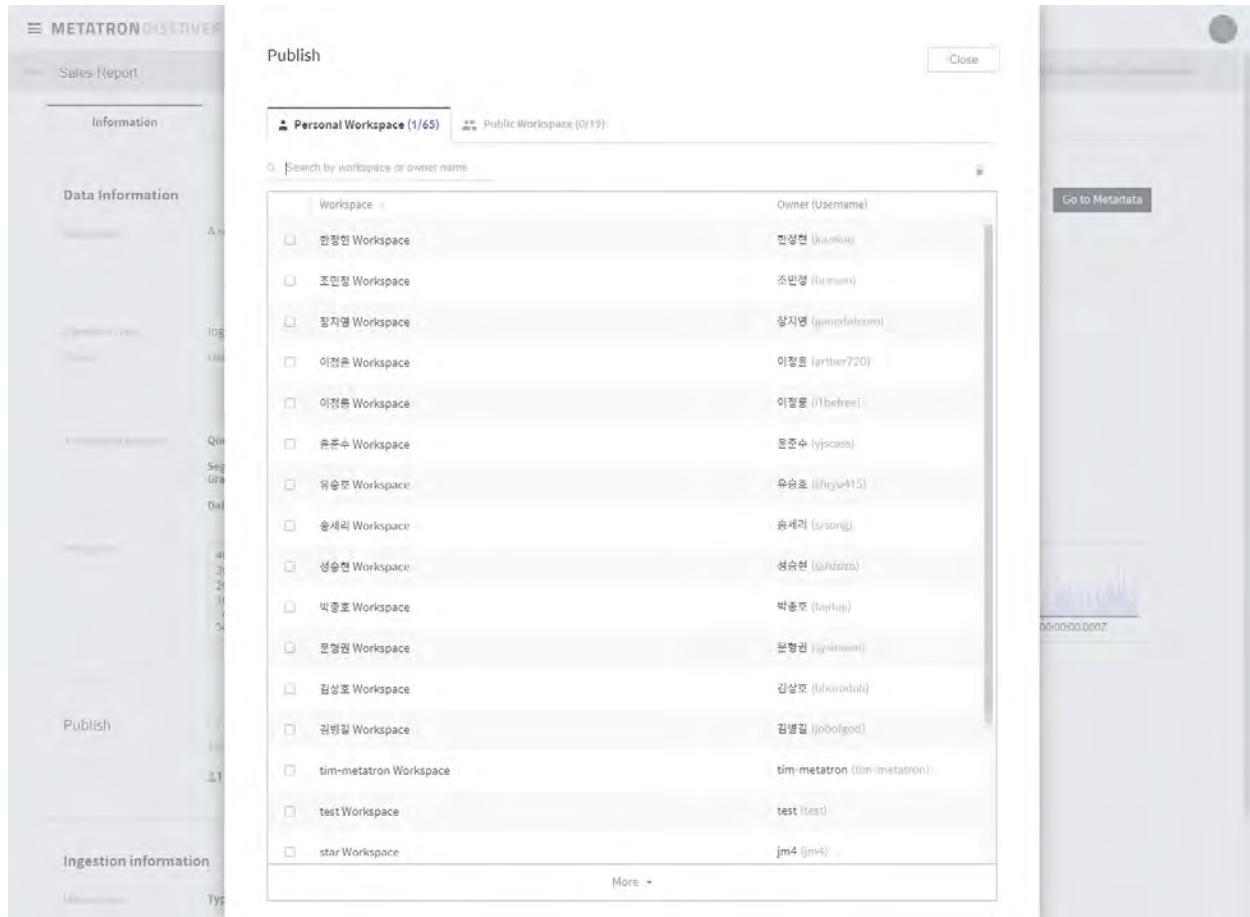
Flotogram: 

Publish: Allow all workspaces to use this datasource
[Edit](#)
[1 workspace](#)

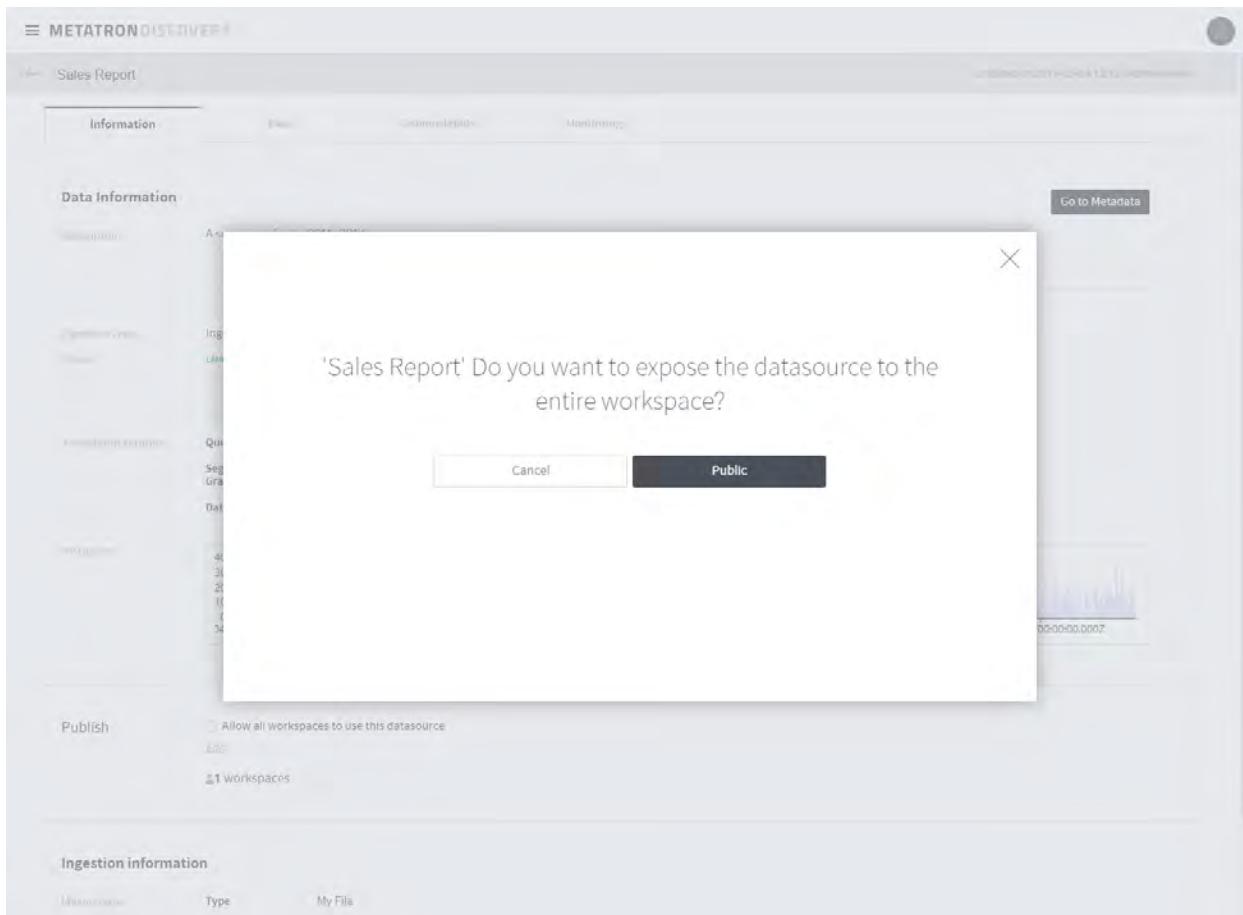
Ingestion information

Master data Type My File

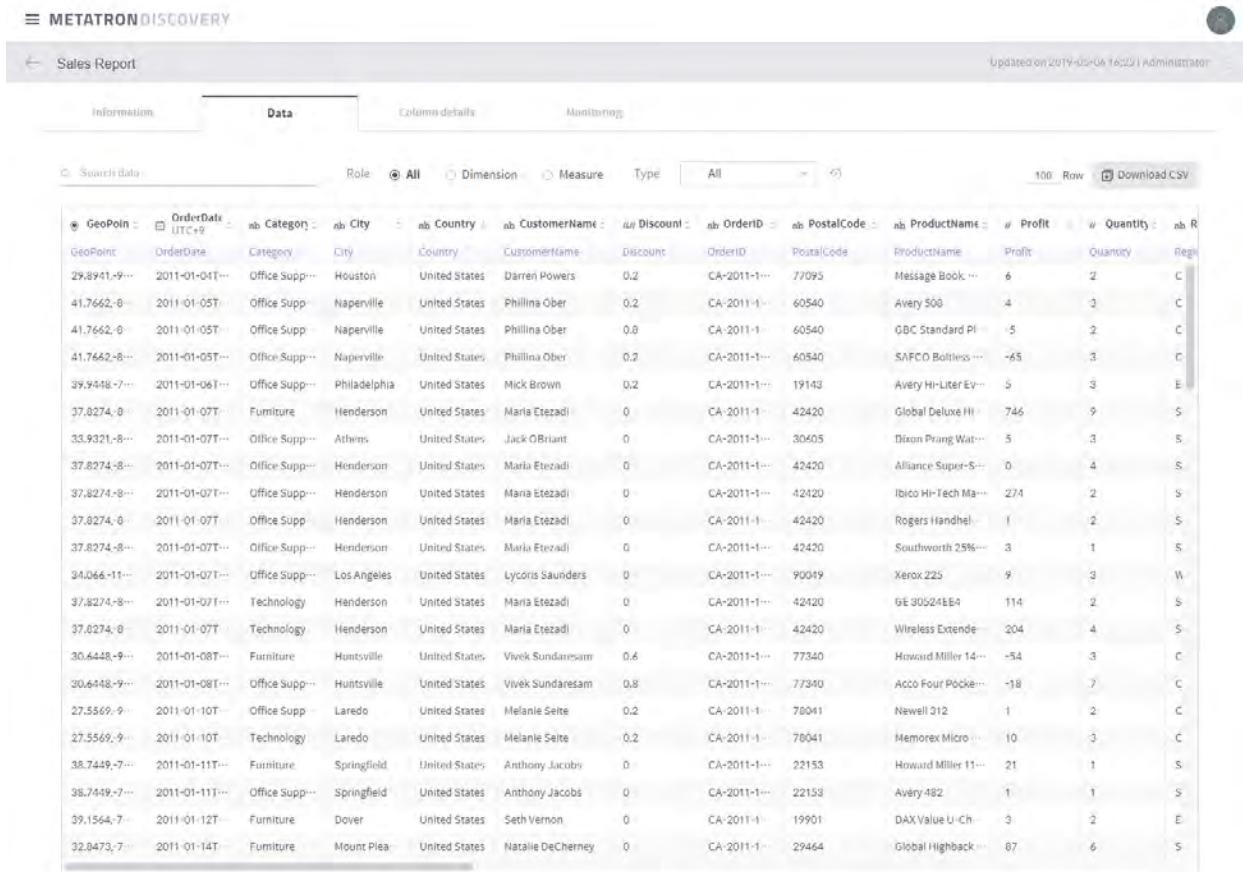
To make the data source available to other users, check the checkbox next to **Allow all workspaces to use this datasource** under **Publish**. To make the data source available only to specific users, click **Edit** and select individual users' or teams' workspaces as desired.



In this example, we will choose **Open Data** to make it available to all users.



The ingested data can be viewed under the **Data** tab.



The screenshot shows the Metatron Discovery interface with a "Sales Report" selected. The top navigation bar includes "METATRON DISCOVERY", a back arrow, and "Sales Report". A timestamp "Updated on 2019-05-06 16:22" and "Administrator" are also present. Below the header are tabs for "Information", "Data" (which is selected), "Column details", and "Monitoring". The main area features a search bar and several filter dropdowns: "GeoPoint", "OrderDate UTC+9", "Category", "City", "Country", "CustomerName", "Discount", "OrderID", "PostalCode", "ProductName", "Profit", "Quantity", and "Region". There are also buttons for "All", "Dimension", "Measure", and "Type". A "100 Row" button and a "Download CSV" link are at the bottom of the filter section. The main data grid displays 30 rows of sales data with columns corresponding to the filters.

GeoPoint	OrderDate UTC+9	Category	City	Country	CustomerName	Discount	OrderID	PostalCode	ProductName	Profit	Quantity	Region
29.8941,-9...	2011-01-04T...	Office Supp...	Houston	United States	Darren Powers	0.2	CA-2011-1...	77095	Message Book...	6	2	C
41.7662,-8...	2011-01-05T...	Office Supp...	Naperville	United States	Phillina Ober	0.2	CA-2011-1...	60540	Avery 500...	4	3	C
41.7662,-8...	2011-01-05T...	Office Supp...	Naperville	United States	Phillina Ober	0.8	CA-2011-1...	60540	GBC Standard PI...	-5	2	C
41.7662,-8...	2011-01-05T...	Office Supp...	Naperville	United States	Phillina Ober	0.2	CA-2011-1...	60540	SACFO Boltless...	-65	3	C
39.9448,-7...	2011-01-06T...	Office Supp...	Philadelphia	United States	Mick Brown	0.2	CA-2011-1...	19143	Avery Hi-Liter Ev...	5	3	E
37.8274,-8...	2011-01-07T...	Furniture	Henderson	United States	Maria Etezadi	0	CA-2011-1...	42420	Global Deluxe HI...	746	9	S
33.9321,-8...	2011-01-07T...	Office Supp...	Athens	United States	Jack OBriant	0	CA-2011-1...	30605	Dixon Prang Wat...	5	3	S
37.8274,-8...	2011-01-07T...	Office Supp...	Henderson	United States	Maria Etezadi	0	CA-2011-1...	42420	Alliance Super-S...	0	4	S
37.8274,-8...	2011-01-07T...	Office Supp...	Henderson	United States	Maria Etezadi	0	CA-2011-1...	42420	Ibico Hi-Tech Ma...	274	2	S
37.8274,-8...	2011-01-07T...	Office Supp...	Henderson	United States	Maria Etezadi	0	CA-2011-1...	42420	Rogers Handel...	1	2	S
37.8274,-8...	2011-01-07T...	Office Supp...	Henderson	United States	Maria Etezadi	0	CA-2011-1...	42420	Southworth 25%	3	1	S
34.066,-11...	2011-01-07T...	Office Supp...	Los Angeles	United States	Lycoris Saunders	0	CA-2011-1...	90049	Xerox 225	9	3	W
37.8274,-8...	2011-01-07T...	Technology	Henderson	United States	Maria Etezadi	0	CA-2011-1...	42420	GE 30524EB4	114	2	S
37.8274,-8...	2011-01-07T...	Technology	Henderson	United States	Maria Etezadi	0	CA-2011-1...	42420	Wireless Extende...	204	4	S
30.6448,-9...	2011-01-08T...	Furniture	Huntsville	United States	Vivek Sundaresam	0.6	CA-2011-1...	77340	Howard Miller 14...	-54	3	C
30.6448,-9...	2011-01-08T...	Office Supp...	Huntsville	United States	Vivek Sundaresam	0.8	CA-2011-1...	77340	Acco Four Pocket...	-18	7	C
27.5569,-9...	2011-01-10T...	Office Supp...	Laredo	United States	Melanie Seite	0.2	CA-2011-1...	78041	Newell 312	1	2	C
27.5569,-9...	2011-01-10T...	Technology	Laredo	United States	Melanie Seite	0.2	CA-2011-1...	78041	Memorex Micro...	10	3	C
38.7449,-7...	2011-01-11T...	Furniture	Springfield	United States	Anthony Jacobs	0	CA-2011-1...	22153	Howard Miller 11...	21	1	S
38.7449,-7...	2011-01-11T...	Office Supp...	Springfield	United States	Anthony Jacobs	0	CA-2011-1...	22153	Avery 482	1	1	S
39.1564,-7...	2011-01-12T...	Furniture	Dover	United States	Seth Vernon	0	CA-2011-1...	19901	DAX Value U-Ch...	3	2	E
32.8473,-7...	2011-01-14T...	Furniture	Mount Plea...	United States	Natalie DeCherney	0	CA-2011-1...	29464	Global Highback...	87	6	S

Congratulations! Now, it's time to use the data source. Let's proceed to the next step.

- Step 2. Create a workbook

1.2 Step 2. Create a workbook

Do you have the data ready for analysis? Now, it's time to create a workbook. The Workbook module supports the visualization of data. Click the Metatron Discovery logo on the upper left to enter your personal workspace.

The screenshot shows the 'Admin Workspace' interface in Metatron Discovery. At the top, there are navigation links for 'Workbook' (64), 'Workbench' (33), and '77 Unfinished'. A search bar is present, along with a message indicating the workspace was last updated on 2019-02-19 by Administrator. The main area displays a grid of 16 items, organized into four rows and four columns. Each item is represented by a card:

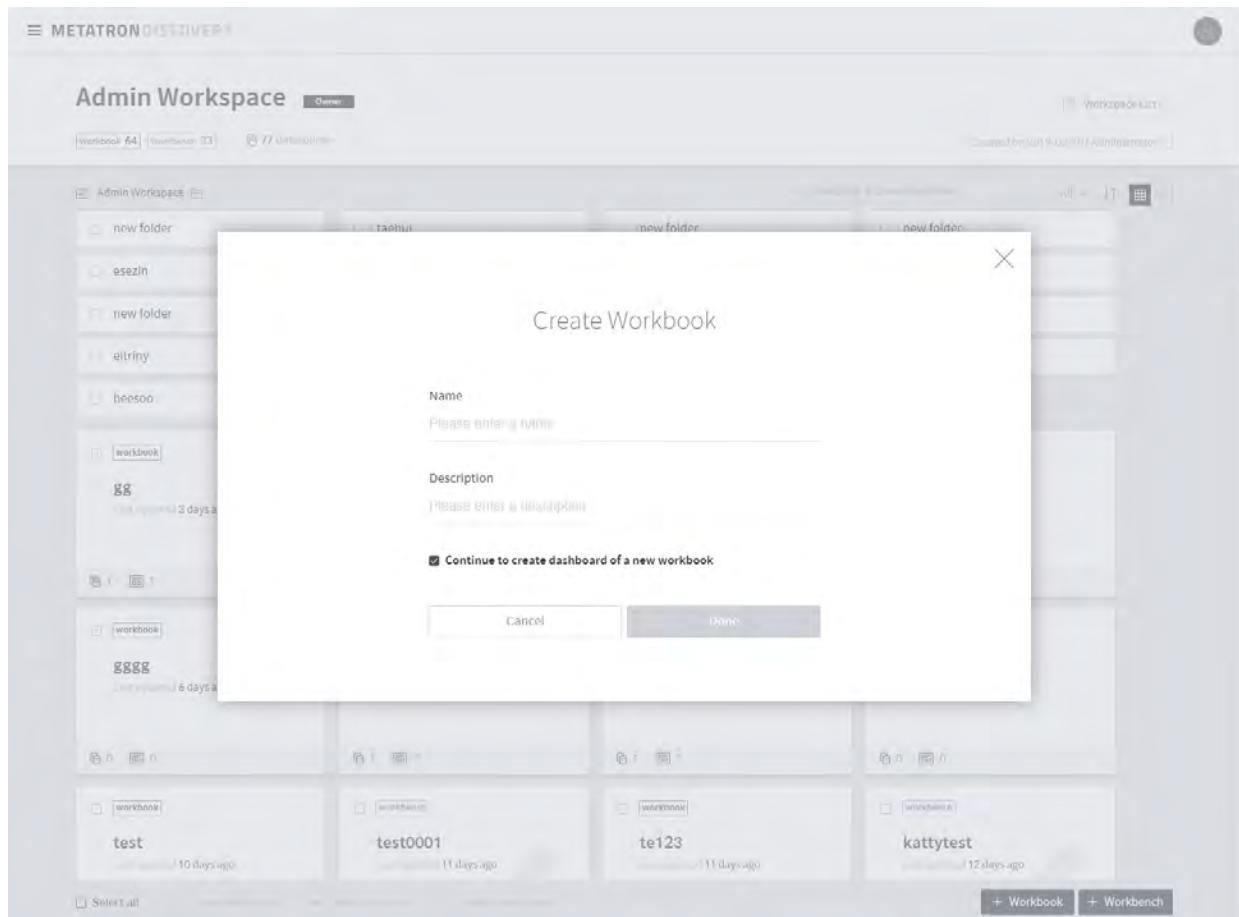
- Row 1:**
 - new folder
 - taehui
 - new folder
 - new folder
- Row 2:**
 - esezin
 - 325
 - Ryan
 - KJ
- Row 3:**
 - new folder
 - comefeel
 - H_VM
 - Realty
- Row 4:**
 - eltriny
 - sting
 - systemshock
 - al

- Row 1:**
 - workbook
 - gg
 - workbook
 - workbook
- Row 2:**
 - ggg
 - Last updated 4 days ago
 - b
 - a
- Row 3:**
 - workbook
 - gggg
 - 3.2 집중테스트 통계
 - workbook
- Row 4:**
 - workbook
 - Last updated 6 days ago
 - asa
 - workbook

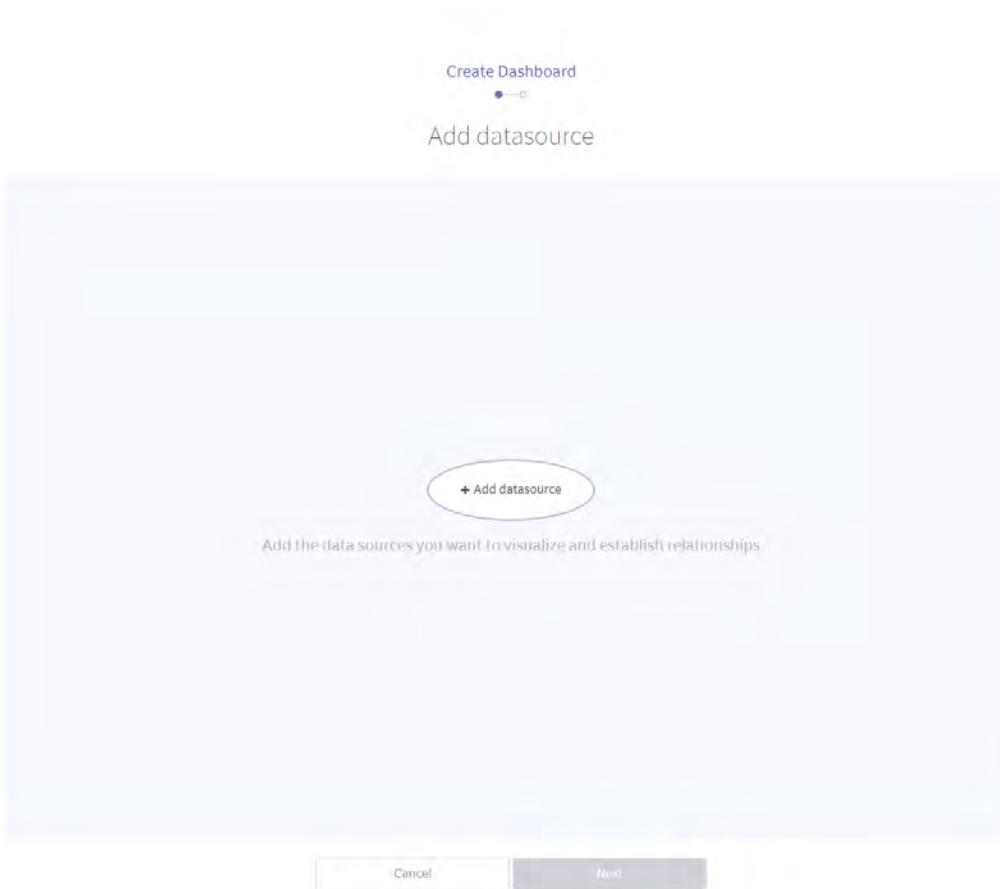
- Row 1:**
 - workbook
 - test
 - workbench
 - workbook
- Row 2:**
 - test
 - Last updated 10 days ago
 - test0001
 - workbench
- Row 3:**
 - workbook
 - te123
 - workbook
 - kattytest
- Row 4:**
 - workbench
 - Last updated 11 days ago
 - te123
 - Last updated 12 days ago

At the bottom right of the grid, there are two buttons: '+ Workbook' and '+ Workbench'.

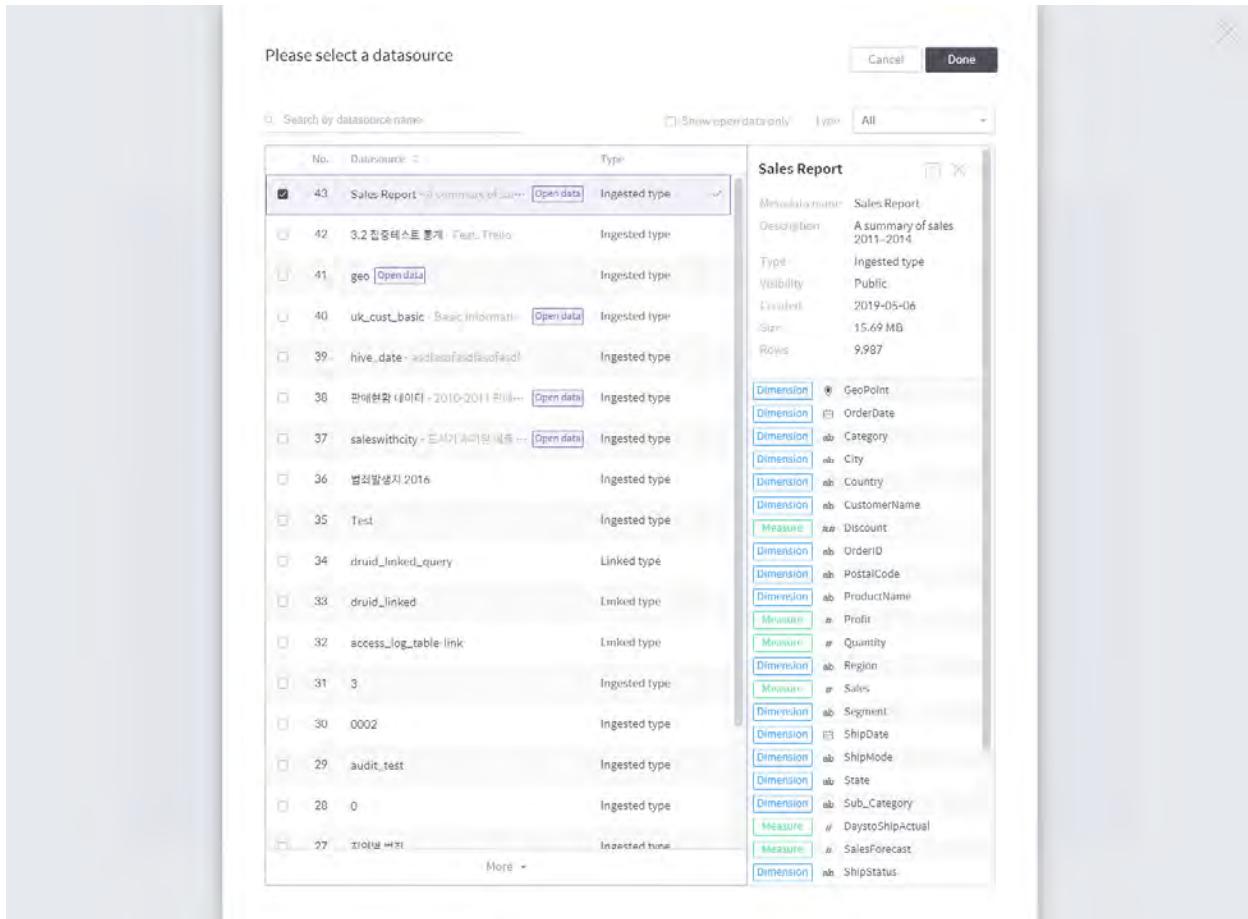
Let's begin by clicking the **+ Workbook** button on the bottom right. Enter the name and description for the workbook. The checkbox is marked by default for you to create a dashboard once a workbook is created. A single workbook contains multiple dashboards, and each single dashboard contains multiple charts.



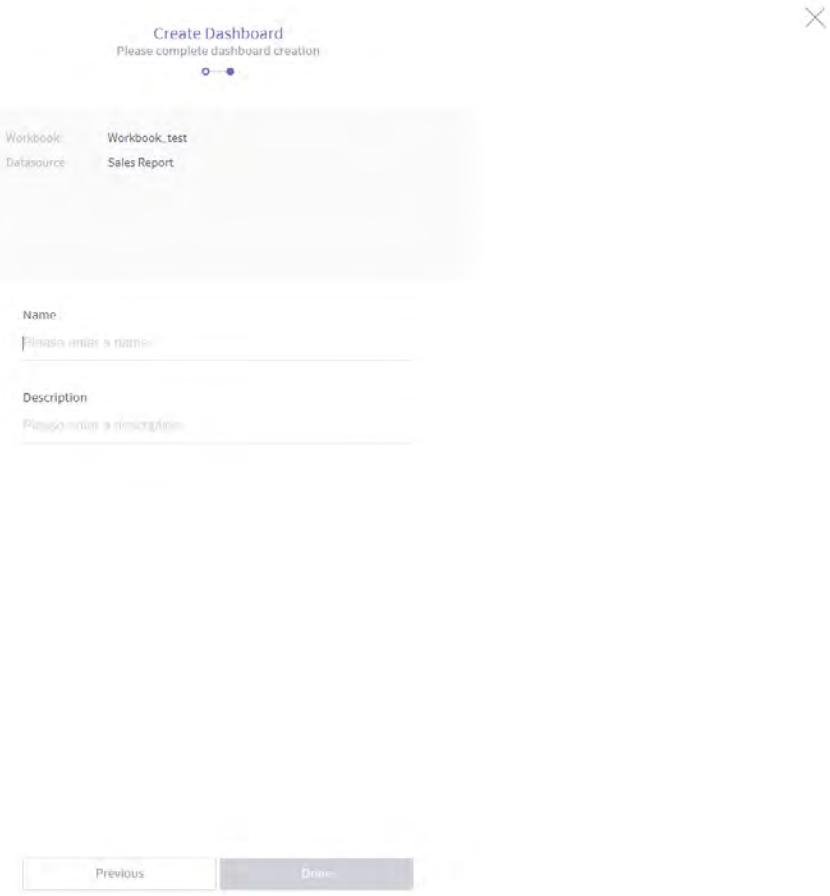
Proceed with creating a dashboard. A dashboard requires a data source for visualization. This data source can be either a single source, or joined data sources. See [Create a dashboard](#) for further details. This tutorial uses Sales Report, ingested previously in Step 1.



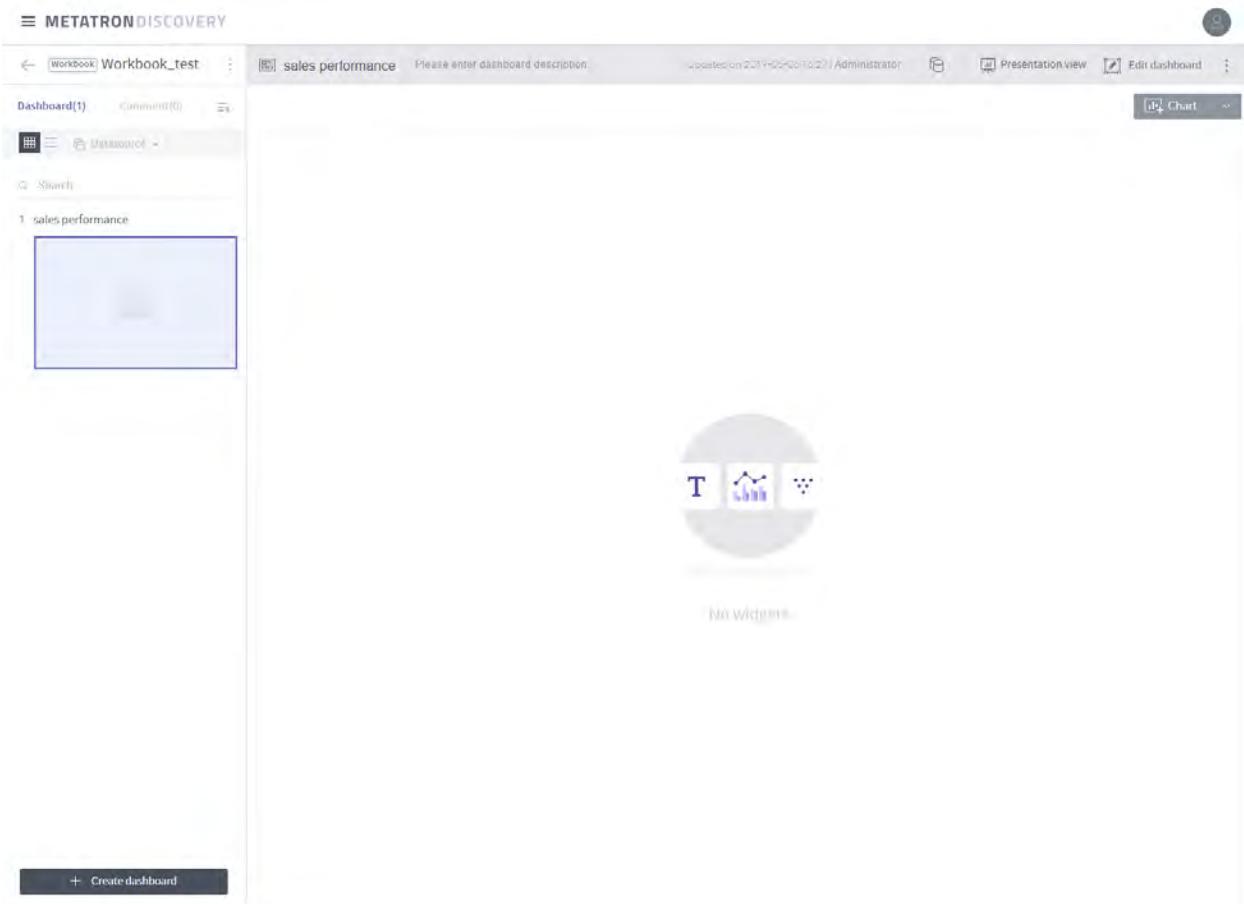
Click the **+ Add data source** button for the data source selection popup. Search Sales Report, or select the **Show open data only** checkbox and choose from the results.



Finally, enter the **Name** and **Description** for the dashboard.



You have created a dashboard in the workbook. Now, you can add widgets to the dashboard.

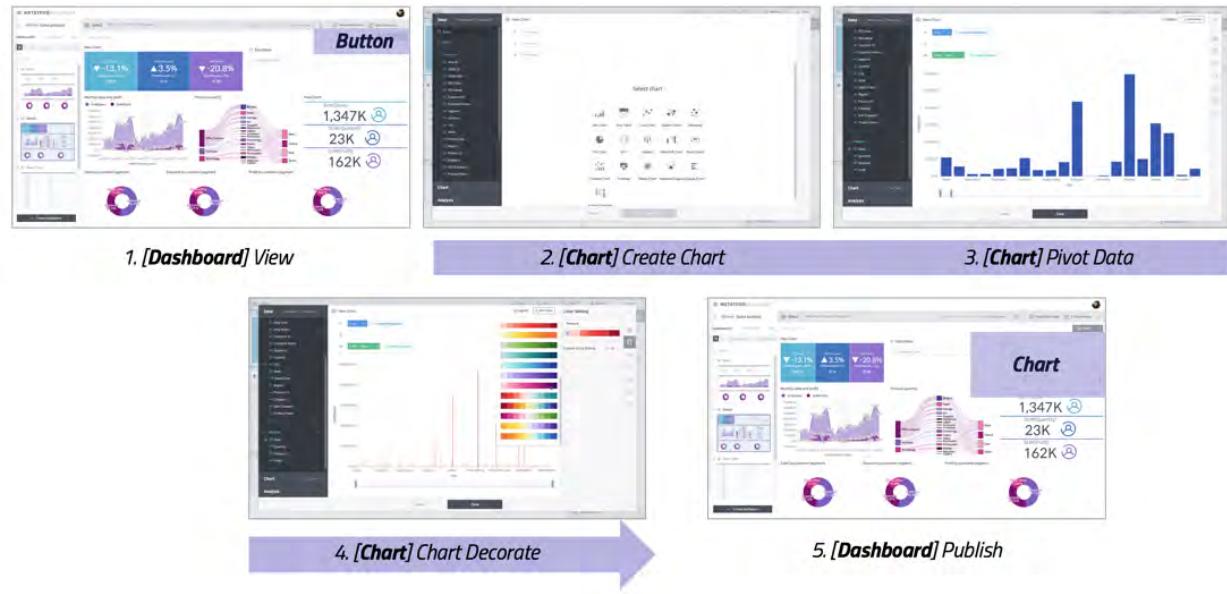


Let's proceed to the next step.

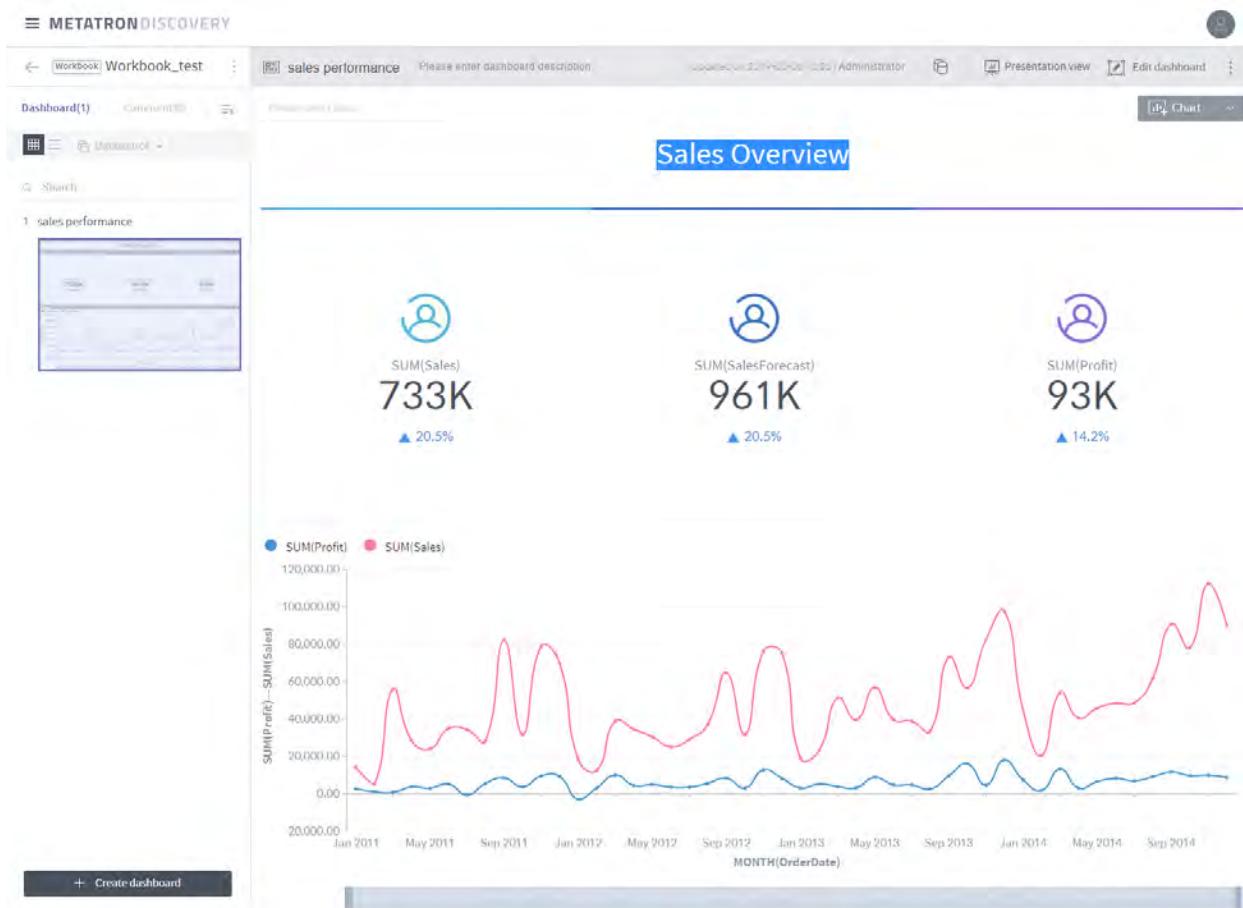
- Step 3. Organize a dashboard

1.3 Step 3. Organize a dashboard

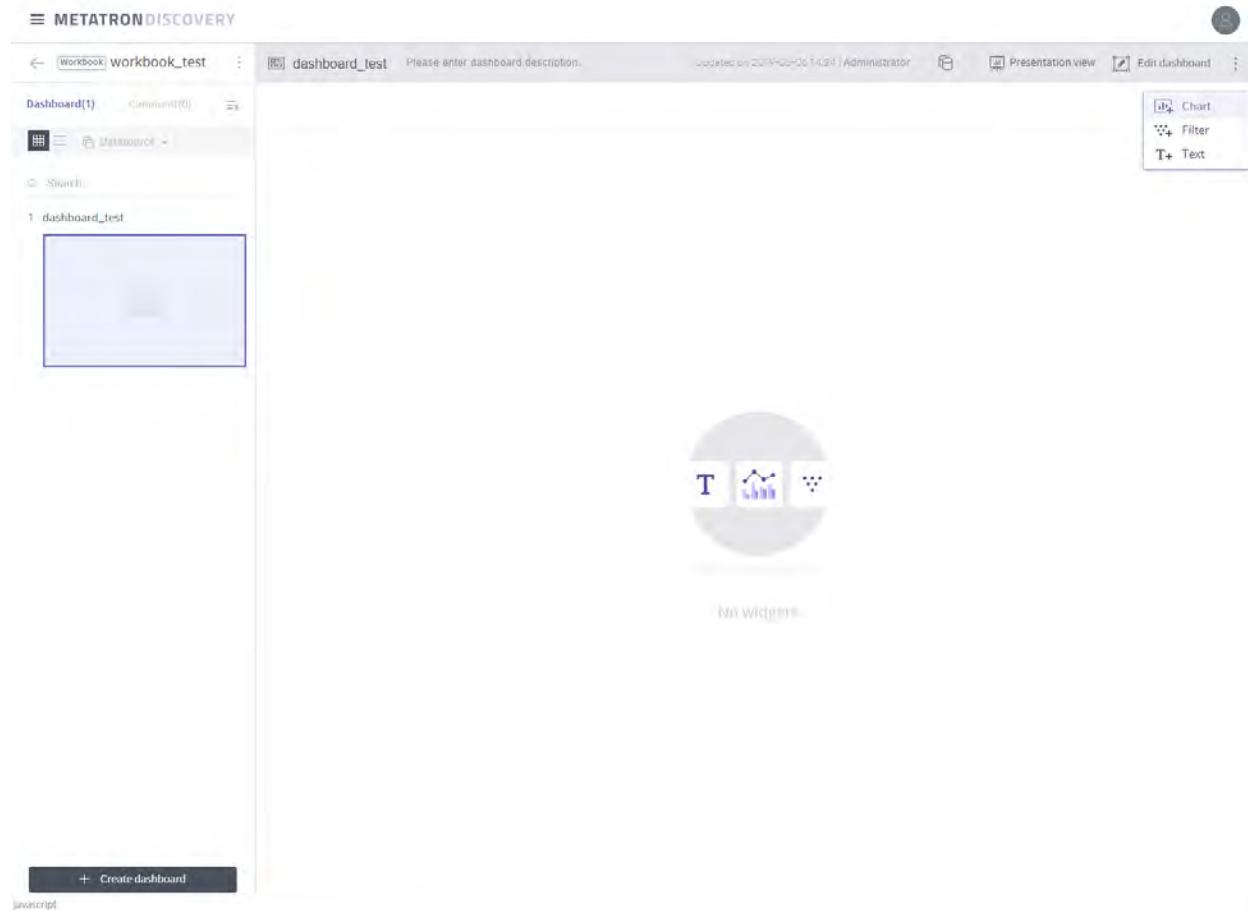
The final step is to create chart widgets, text widgets, and filter widgets to fill the empty dashboard. The dashboard can be edited in the following order:



Using the Sales Report created earlier, let's add a key performance indicator chart and a line chart to the dashboard.

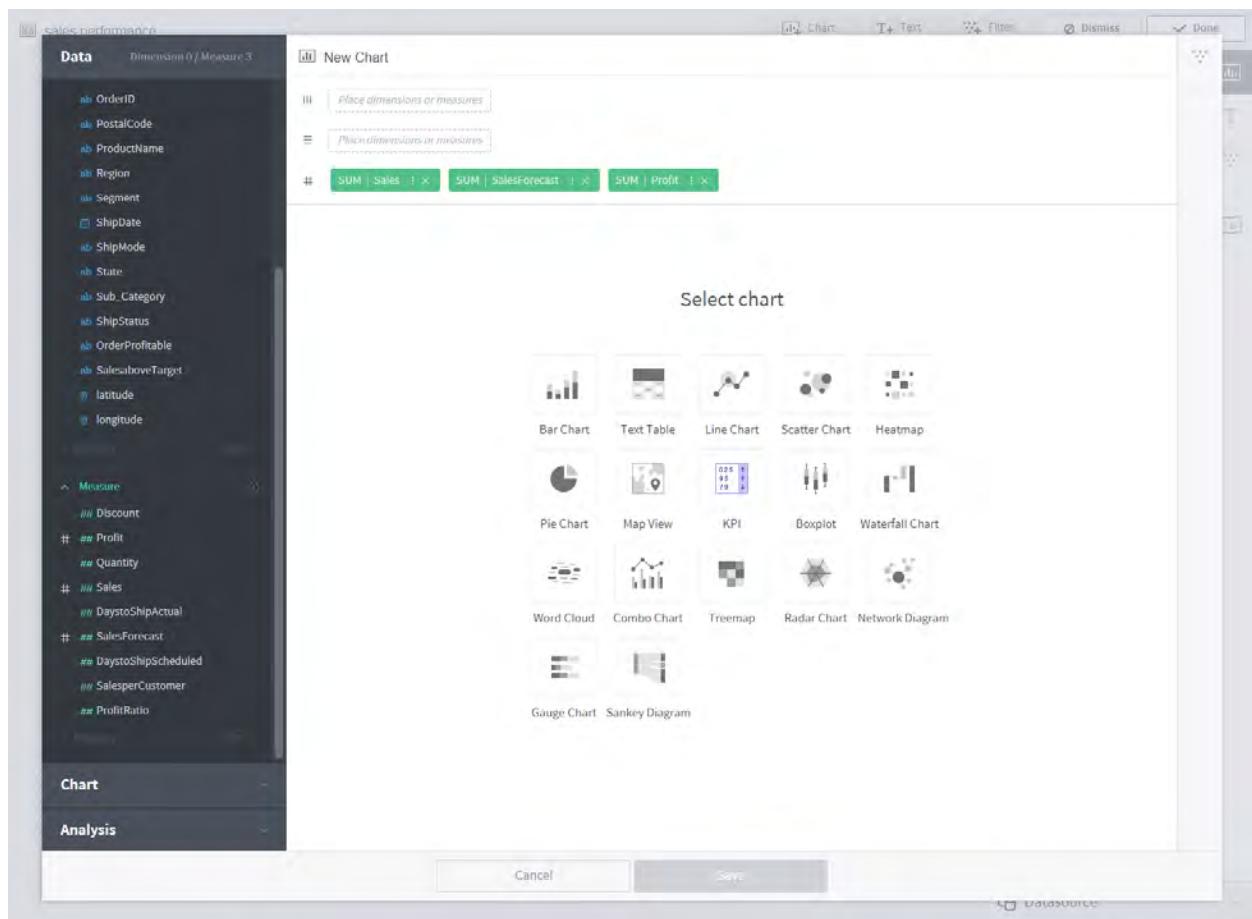


In the empty dashboard, click the **Chart** button to create a chart.

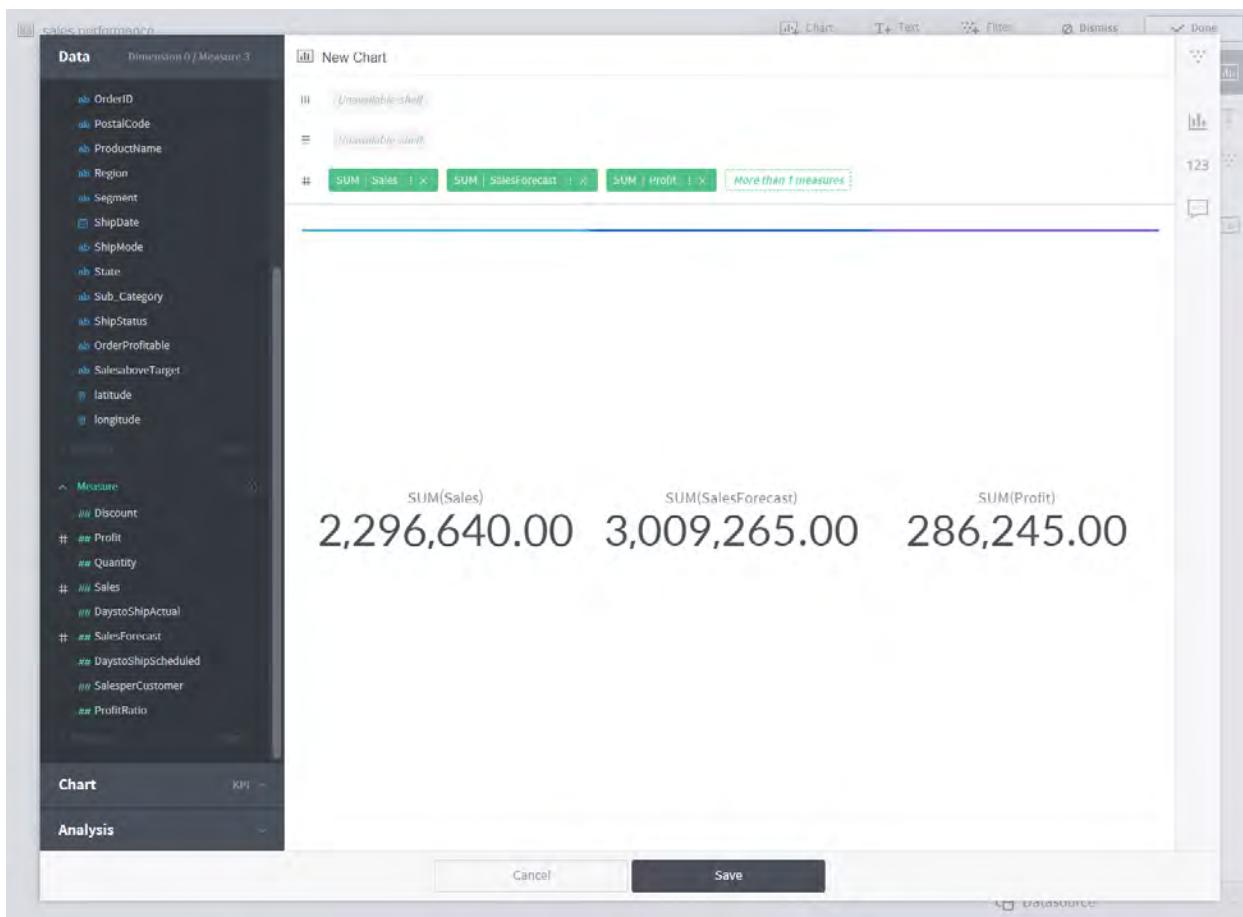


1.3.1 Creating a key performance indicator chart

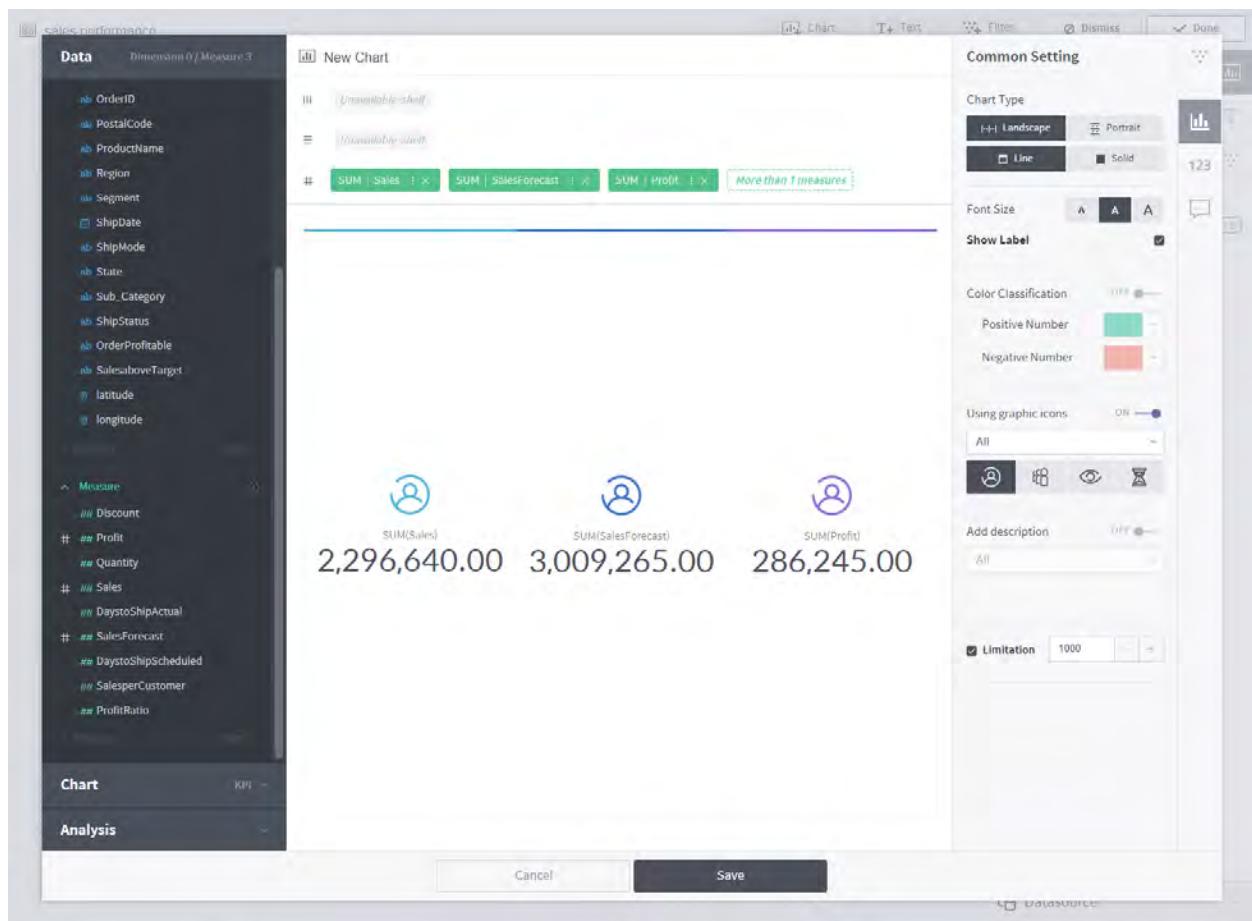
The first chart you will be creating is a key performance indicator (KPI) chart. The KPI chart is a simple yet powerful chart that displays the goals of an organization in an intuitive manner. The goal of our dashboard is to clearly present sales data. As such, the KPI chart should include total sales, sales forecast, and profit. What should we do? Simply click the three measurement columns named “Sales,” “SalesForecast,” and “Profit” under the Data menu. This task is called pivoting. The pivoted columns are automatically aggregated and placed on shelves. Once columns are on shelves, suitable charts are recommended. How about clicking the recommended KPI chart?



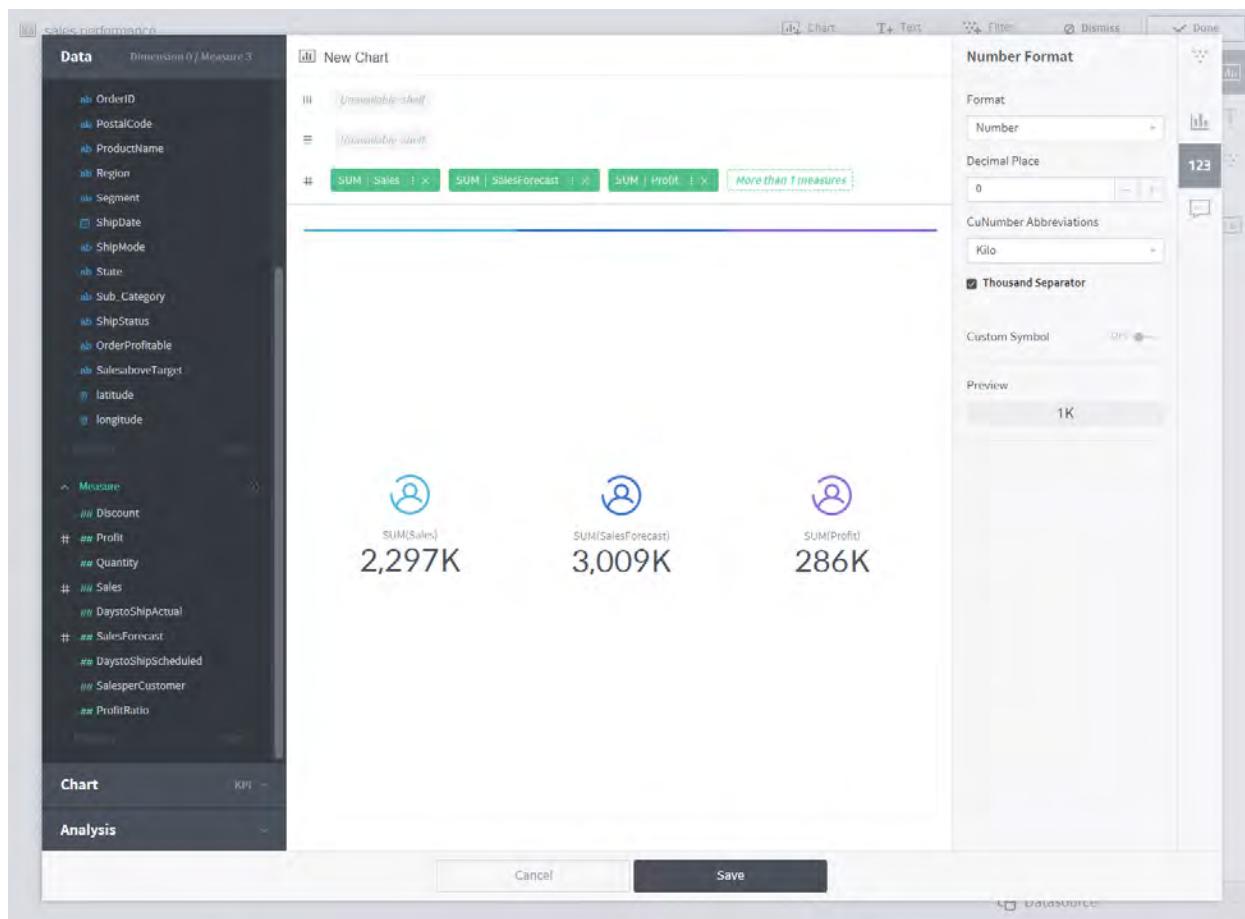
The KPI chart is created as follows: To make it more presentable, let's enter the chart properties menu on the right.



Click to enter the **Common Setting** panel and add an icon to each measure column.

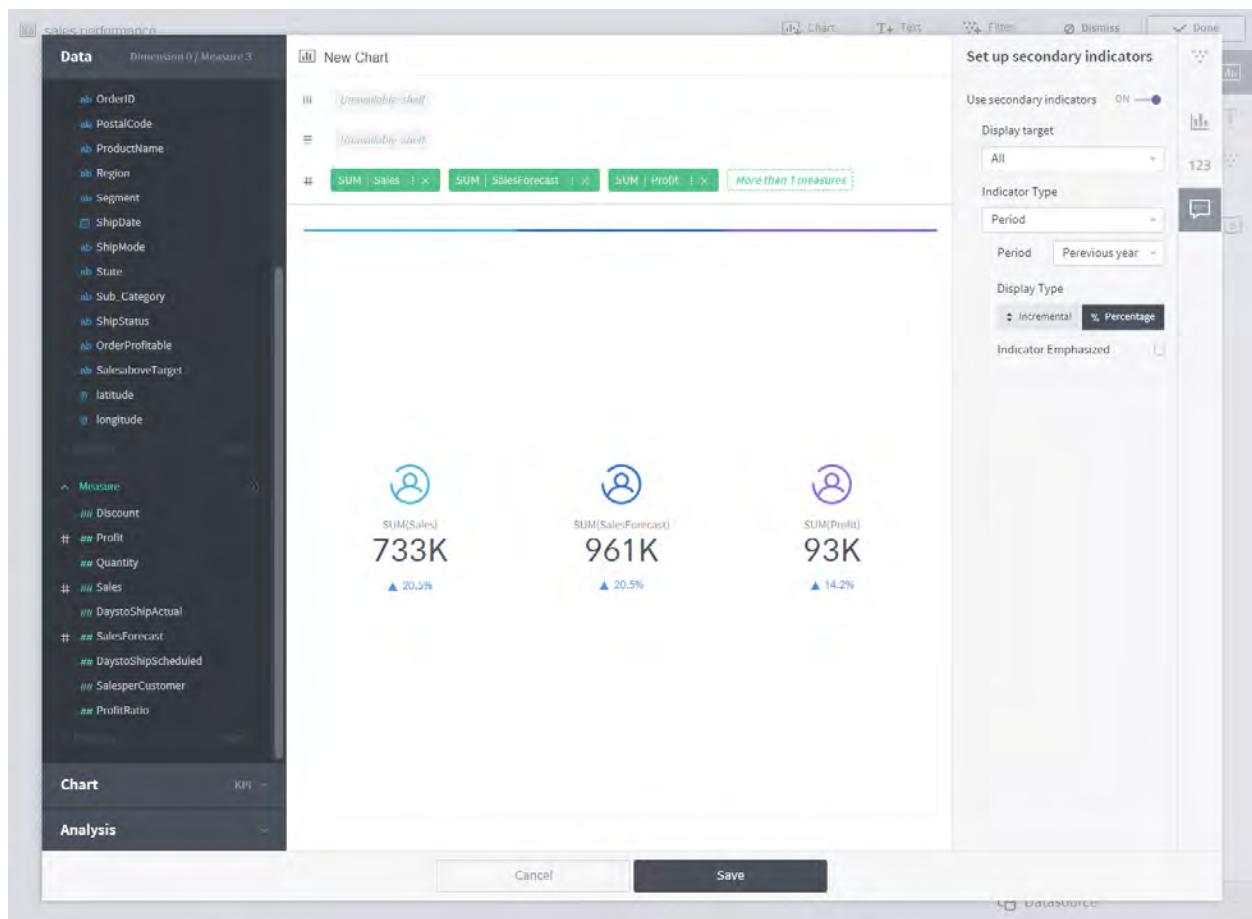


Click **123** to enter the **Number Format** panel and change the decimal place and abbreviation display.

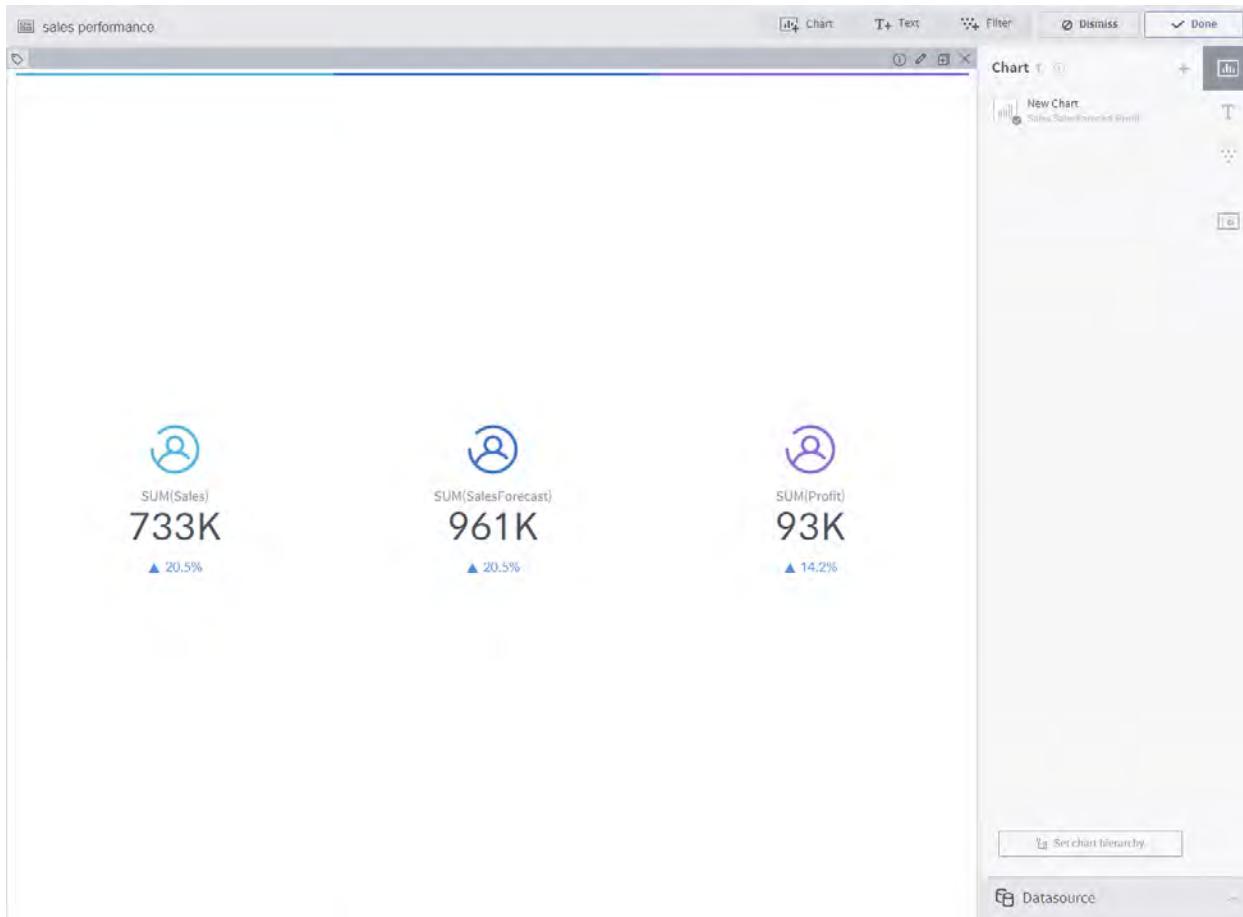


The most important feature of the KPI chart is comparing present achievements with past performance.

Click to enter the **Set up secondary indicators** panel. Set a secondary indicator, and check the % improvement in performance compared to the previous month. If you wish, you can emphasize the secondary indicator instead of the original indicator.

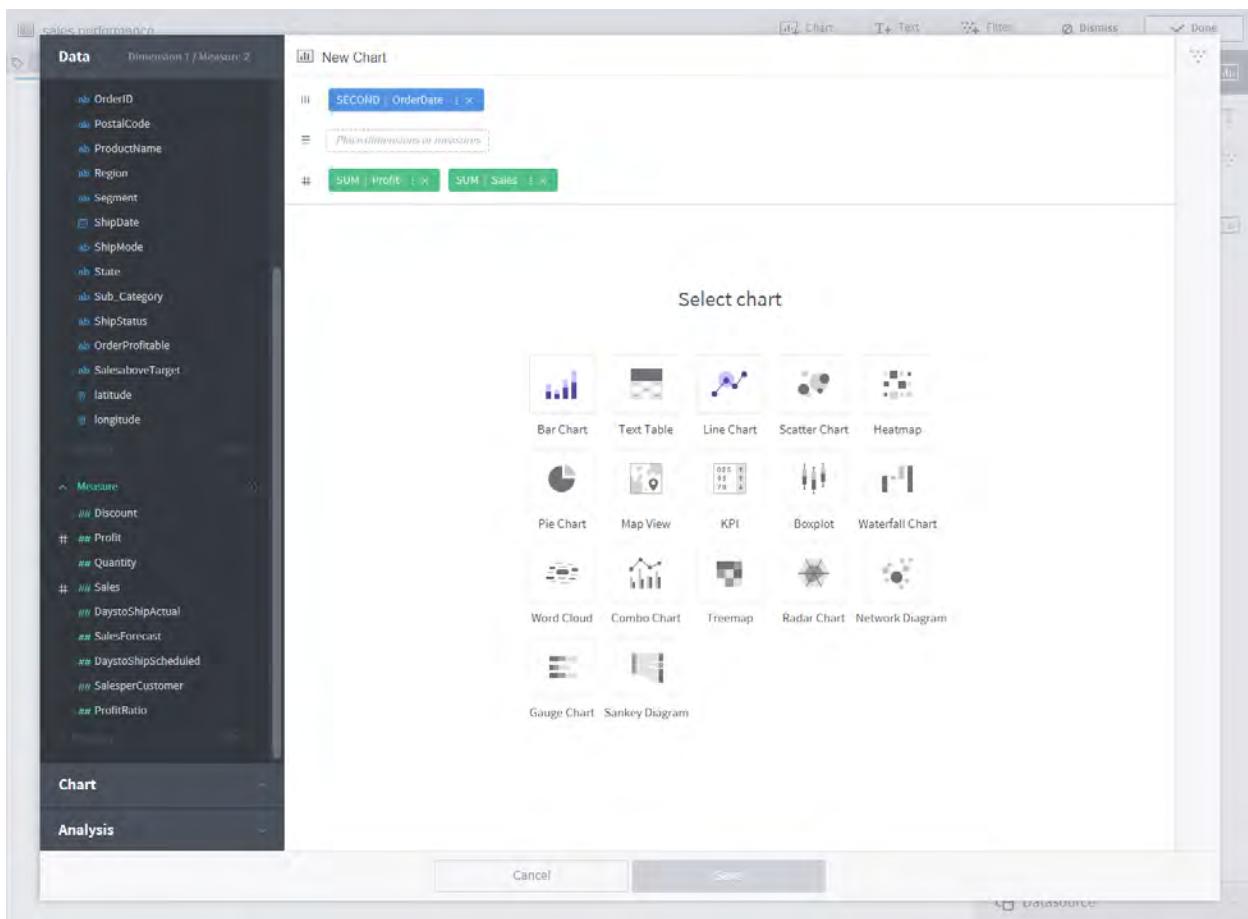


Click **Save** to display the chart in the dashboard.

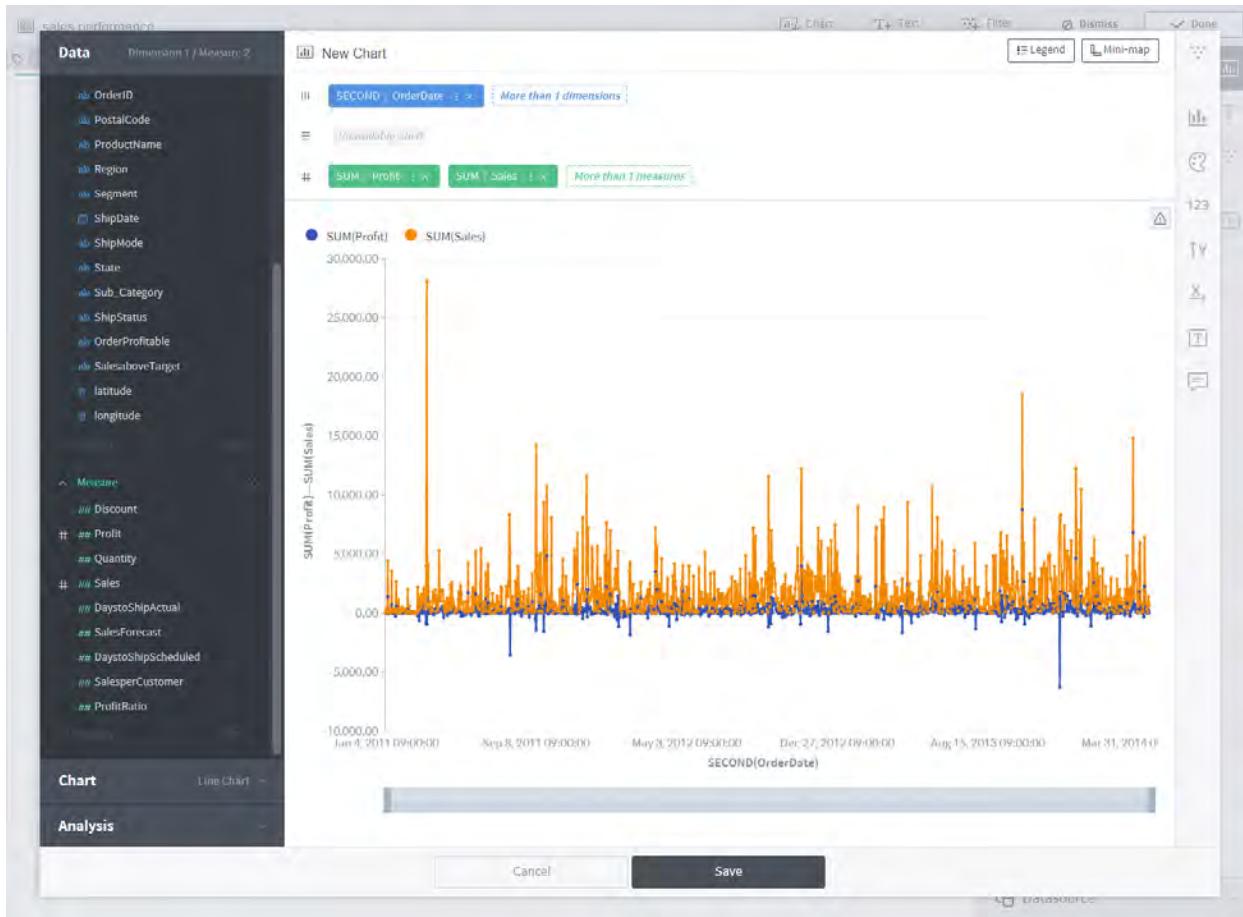


1.3.2 Creating a line chart

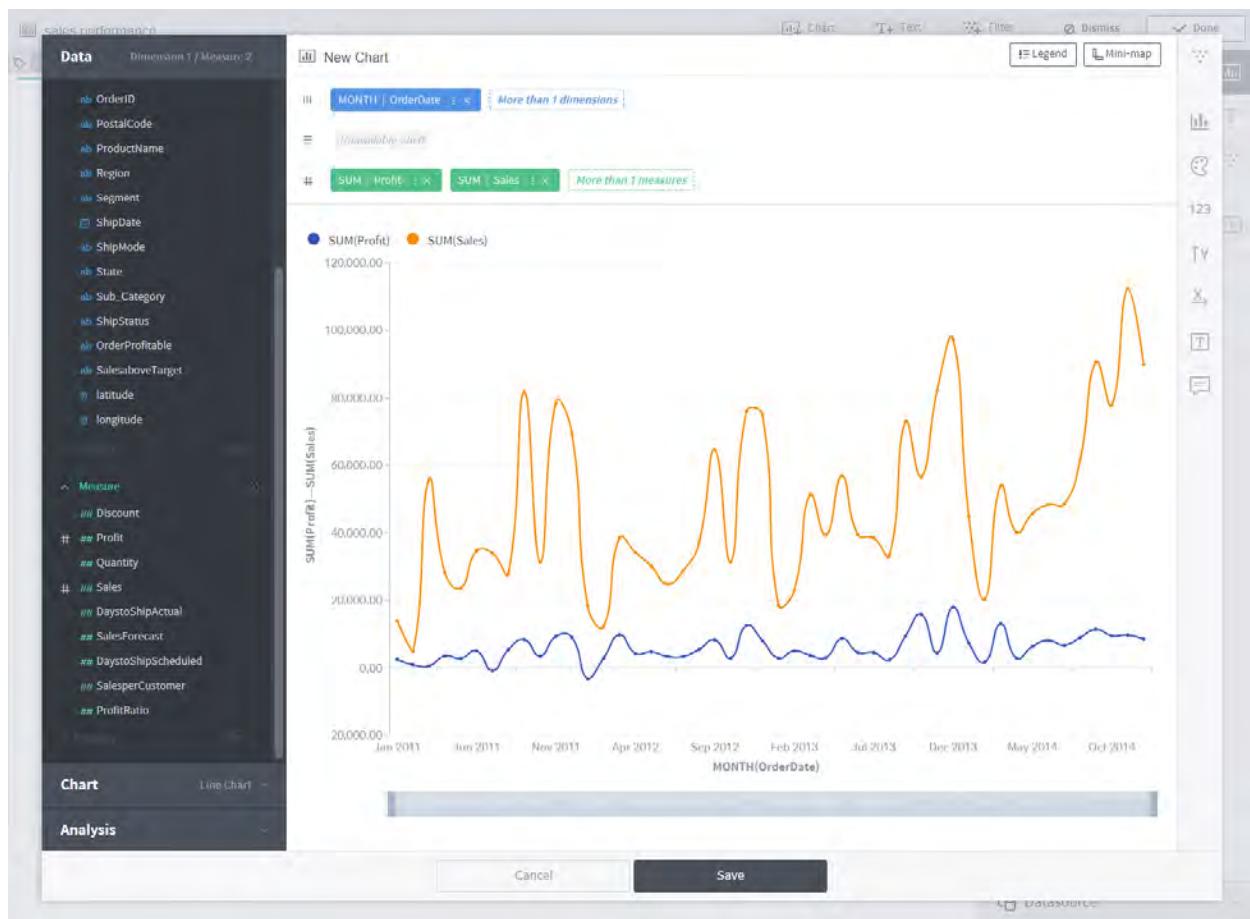
Next, let's create a line chart, the most basic type of chart. Shall we take a look at how sales and profit change over time? Again, click the **Chart** button to begin drawing a new chart. Click the OrderDate, Profit, and Sales columns to see how the values change over time. Click the recommended **Line Chart**.



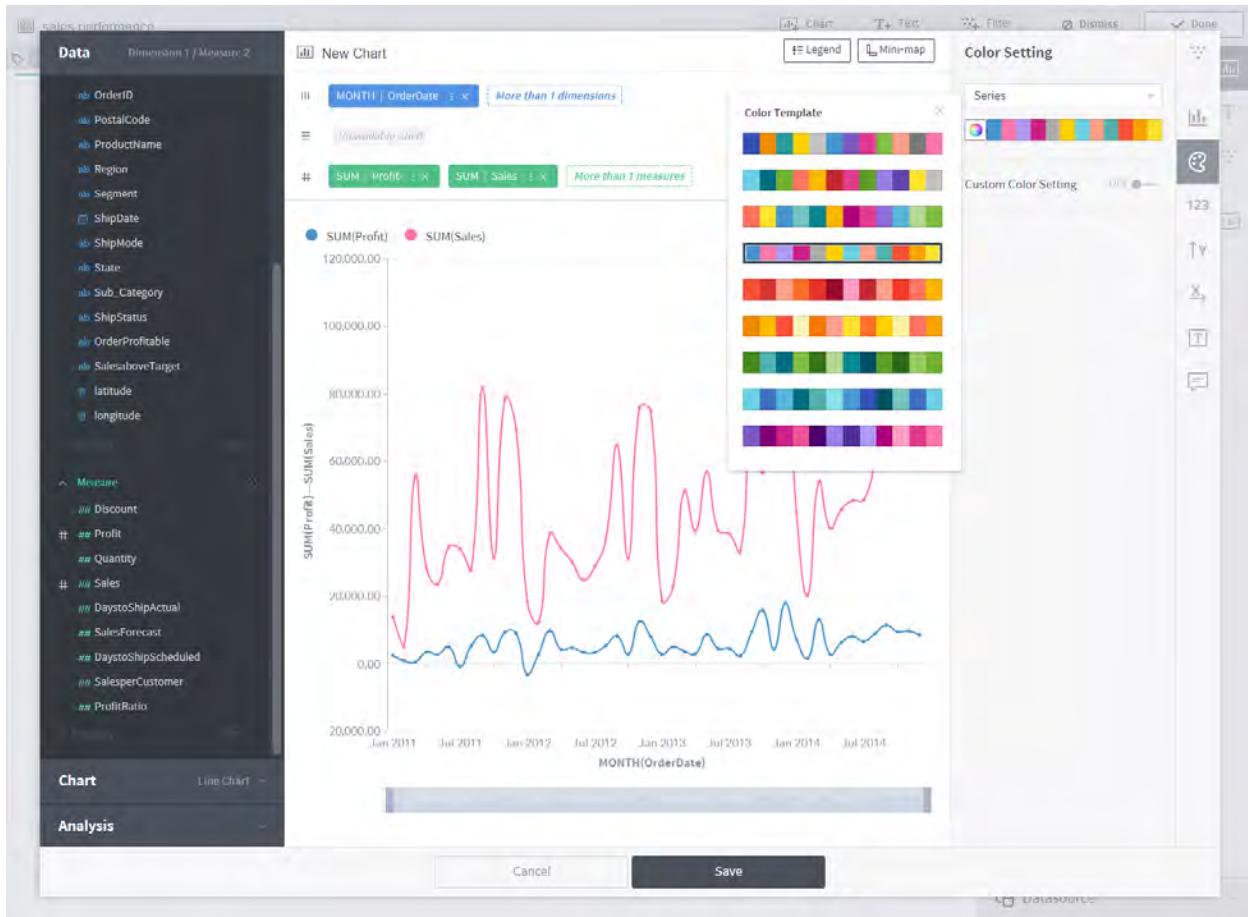
A line chart is drawn. Open the chart properties panel, and change the line shape to “round.”



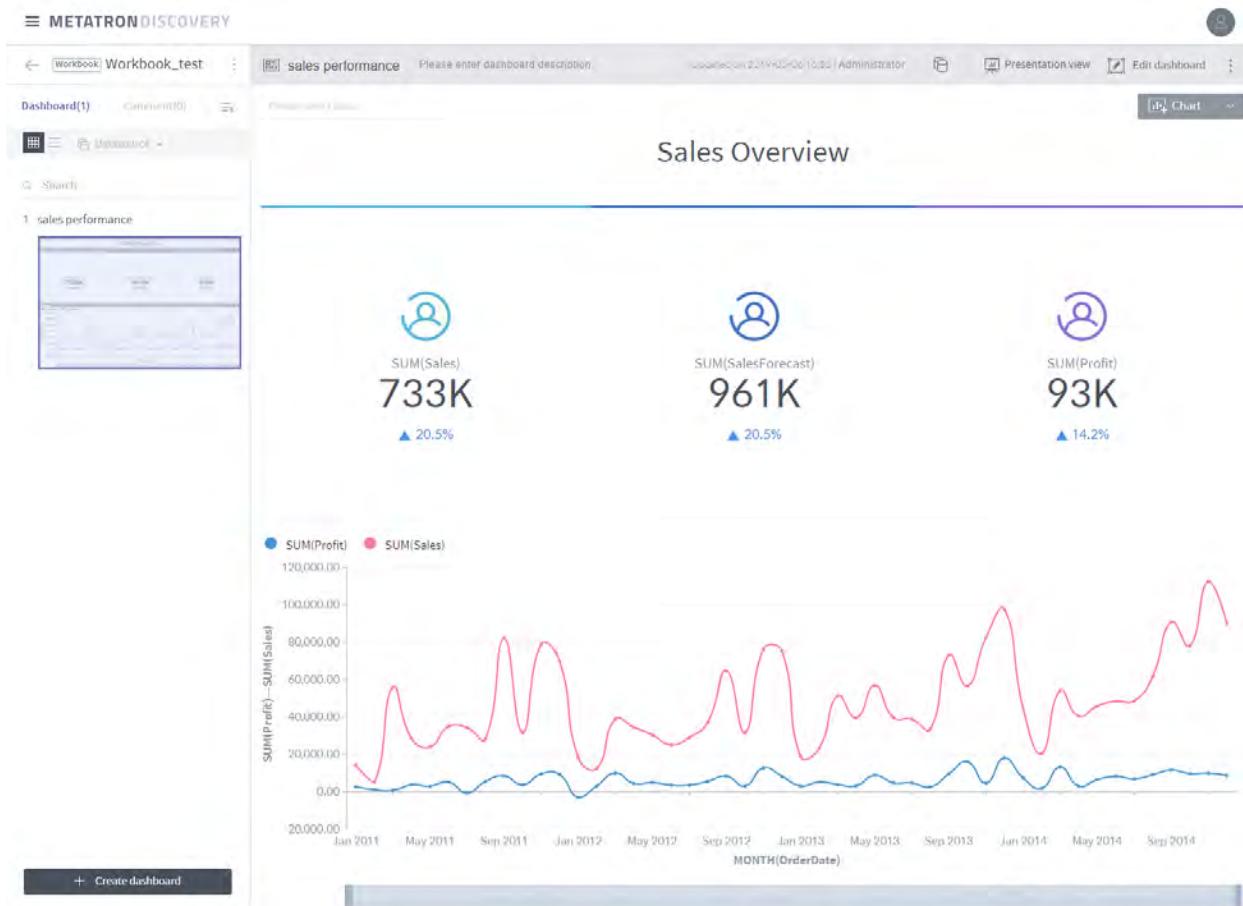
There is too much data as OrderDate is aggregated on an hourly basis. To view by month, go to the menu of the OrderDate column, and set **Granularity as Month**. The entire data is displayed now! Click **Mini Map** on the upper right to remove the mini map from the chart.



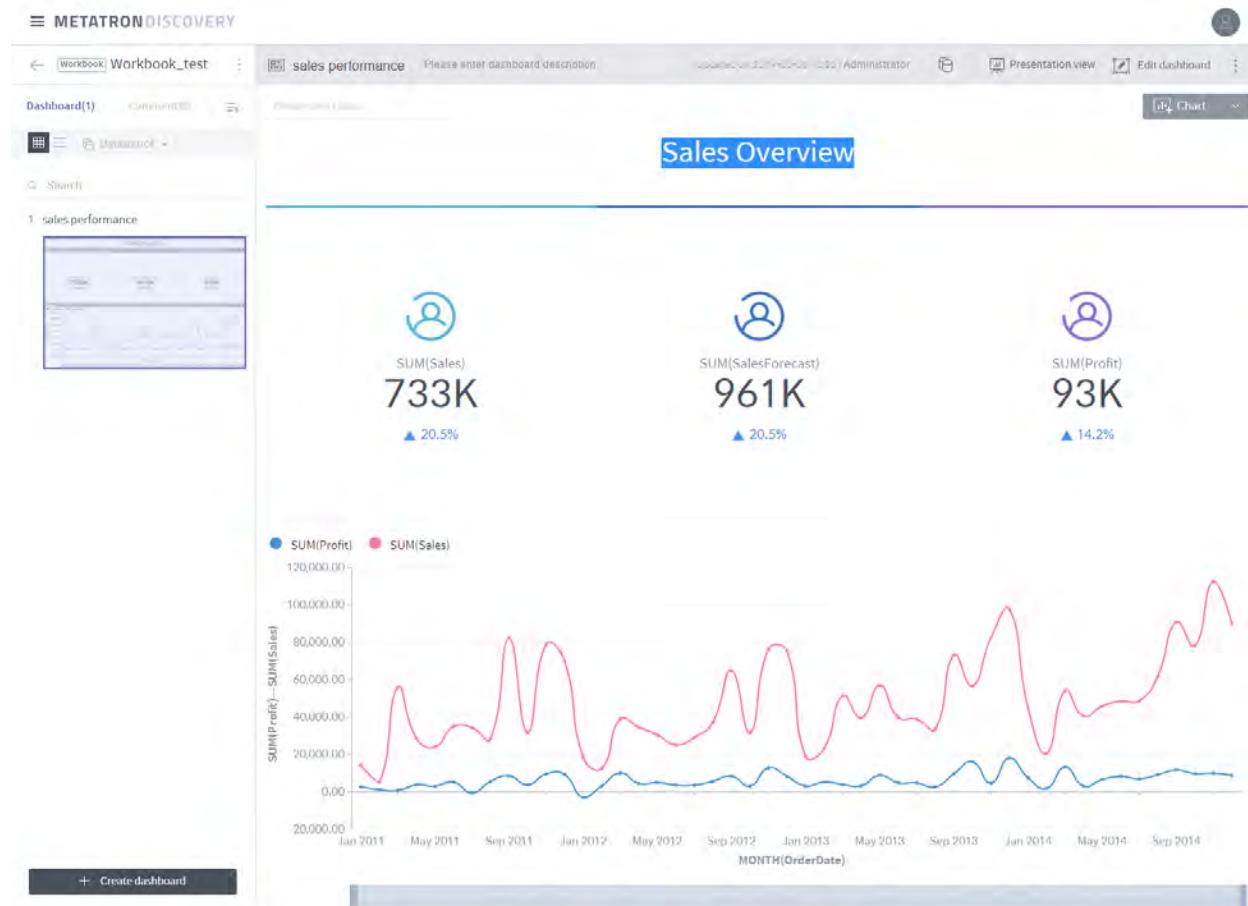
Click on the right menu, and change colors using the Color Setting panel.



Click Save, and drag and drop the chart to the desired position. Add information to the dashboard by adding a **text widget**. Click **Done** to finish dashboard editing.



In this tutorial, you learned how to draw two chart types. Using the interactive dashboard, you can select a chart or add filters to present data as desired. You can also modify, add, or delete charts if required.



Are you ready to learn more about Metatron Discovery?

- Overview of Metatron Discovery
- Components of Metatron Discovery
- Metatron engine: Druid

CHAPTER
TWO

INTRODUCTION OF METATRON DISCOVERY

Metatron Discovery is a solution that analyzes data ingested into the Metatron server cluster in a simple, sophisticated manner, and visualizes the results in the user PC in the form of charts and reports. A web-based application, it is highly accessible such that it can be remotely accessed by from any PC.

This section introduces the technical background and structure of Metatron Discovery, and the Druid engine powering Metatron.

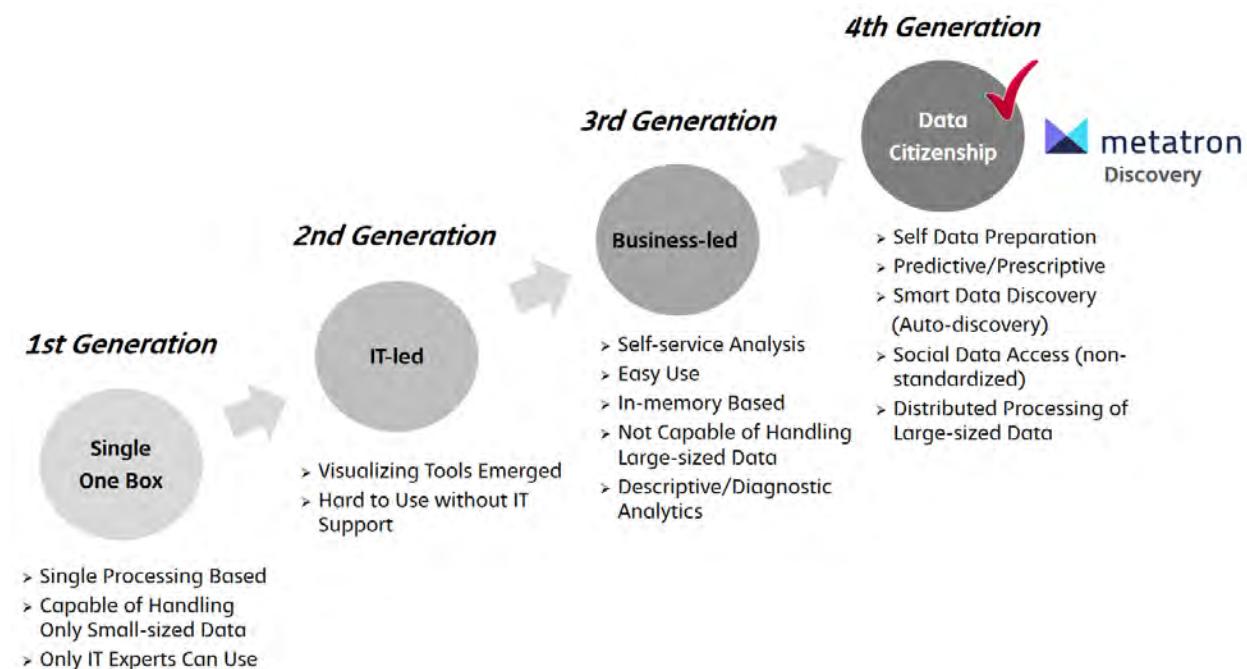
2.1 Overview of Metatron Discovery

Metatron Discovery is a 4th-generation OLAP-based business intelligence (BI) solution that combines OLAP, visualization, and machine learning technologies for even non-experts to quickly and easily derive higher-level value from data.



2.1.1 4th-generation BI solution

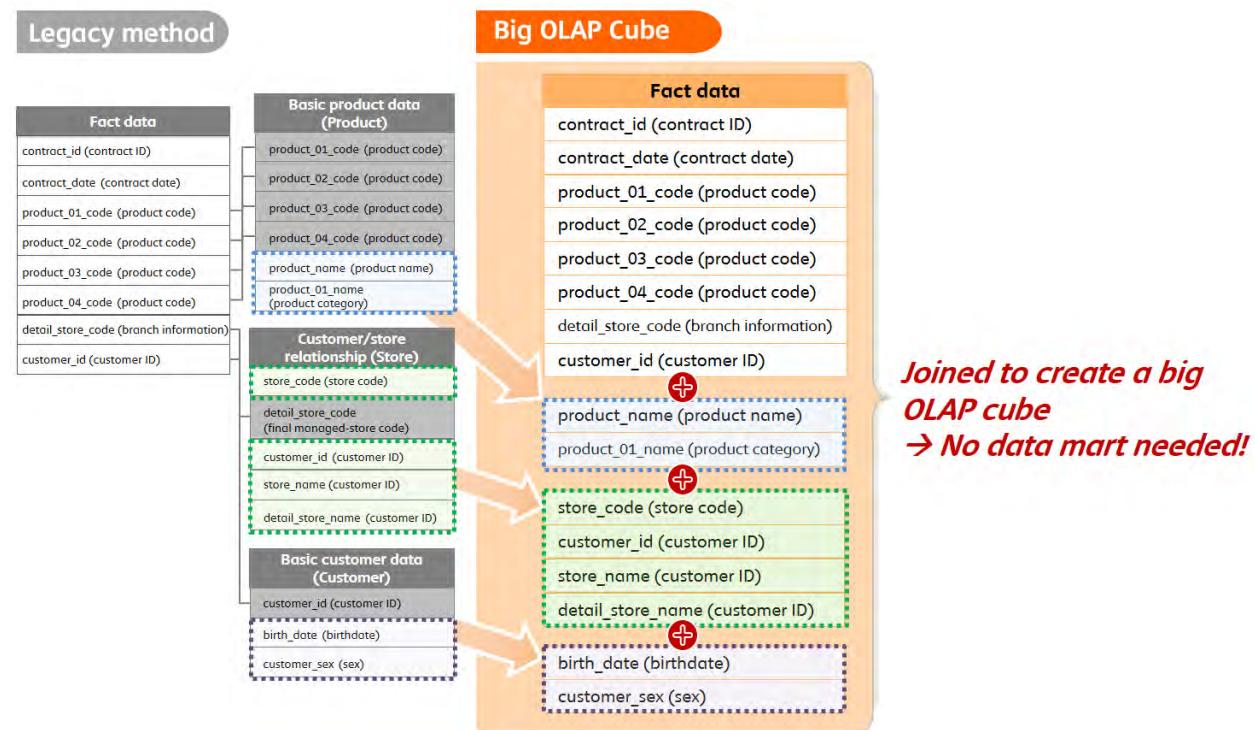
The figure below shows BI trends from the 1st to 4th generation.



The mainstream products in the current BI market belong to the 2nd and 3rd generations, and 4th generation products are beginning to come under the spotlight. As a 4th generation BI solution, Metatron Discovery supports self & ad-hoc data discovery and guarantees rapid response to big data.

2.1.2 Built on Big OLAP

Metatron Discovery combines data of various dimensions for large-sized fact data to produce a single big OLAP cube (data mart).



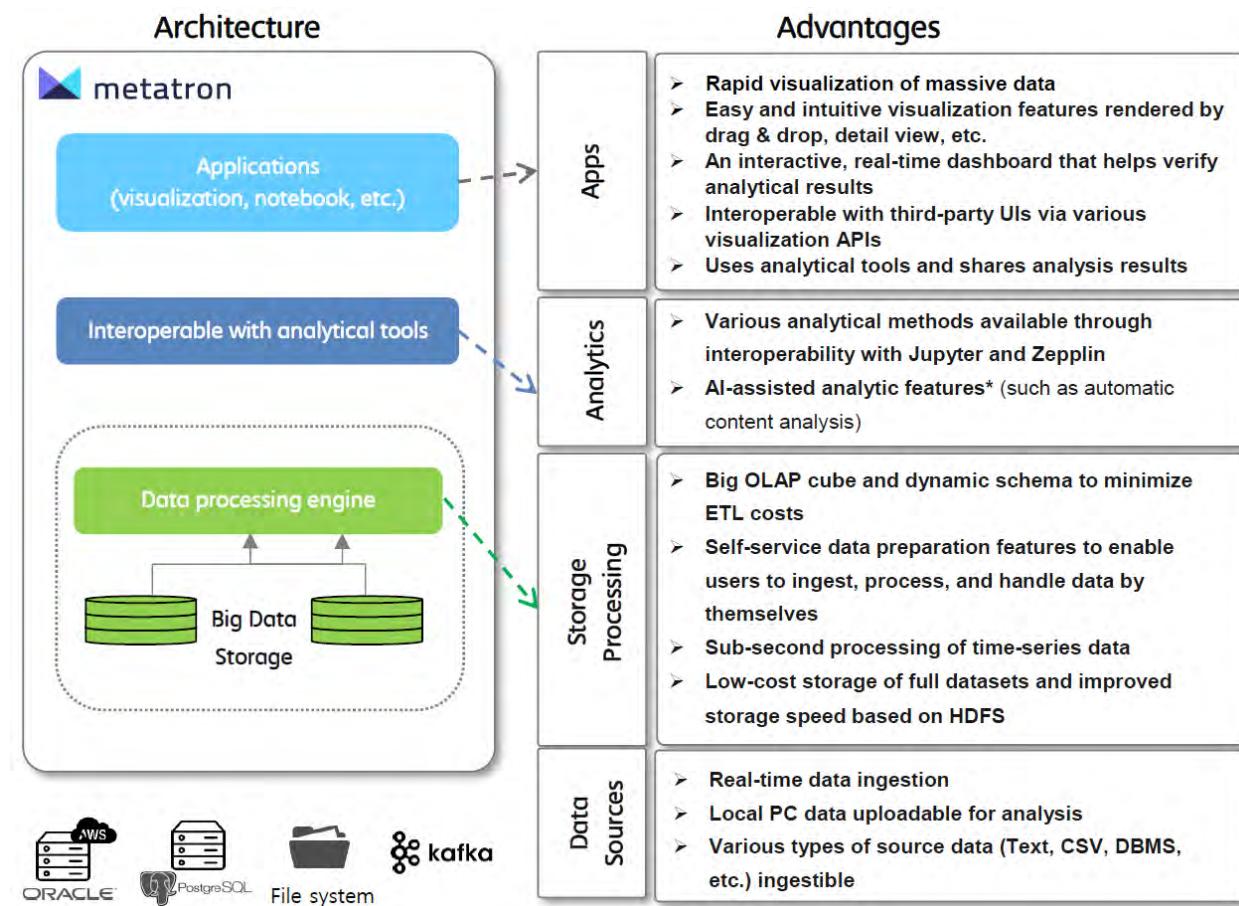
The use of a big OLAP cube offers the following advantages:

- Minimizes the number of data marts.
 - Lower ETL cost for data mart production.
 - Influence of structural change can be minimized.
 - Satisfies diverse demands by saving all fact data.
- Distributed architecture allows storing of large-scale data and ensures fast data processing.

- With a dynamic schema approach, schema changes do not require schema redefining.
- Data can be processed at the record level in real time as tables are saved with no data loss.

2.1.3 Architecture of Metatron Discovery

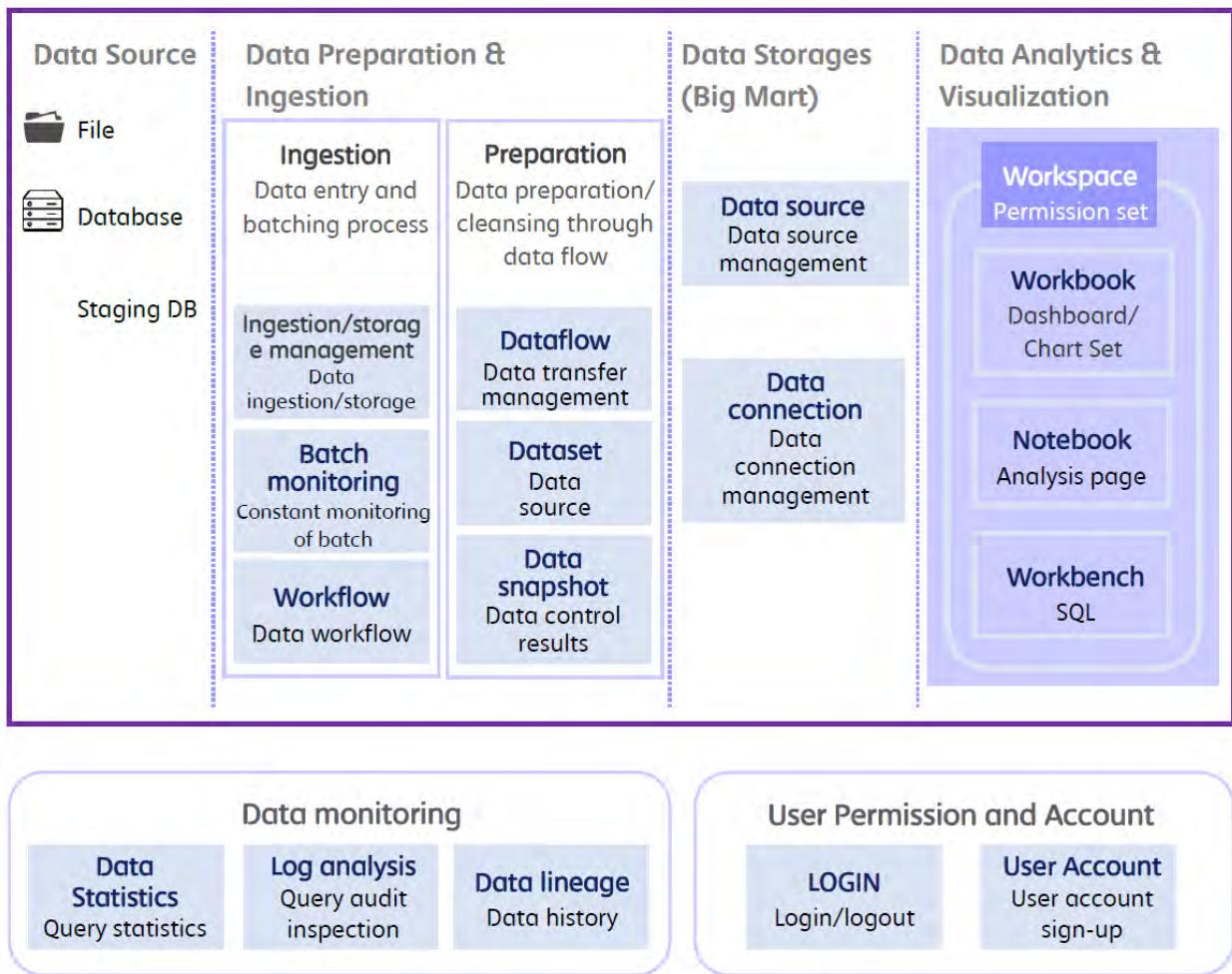
Metatron Discovery is an end-to-end solution that supports the entire process of data discovery, from preparation of large-scale data to data visualization and exploration and to advanced analytics. The figure below is a summary of Metatron's architecture and key features.



2.2 Components of Metatron Discovery

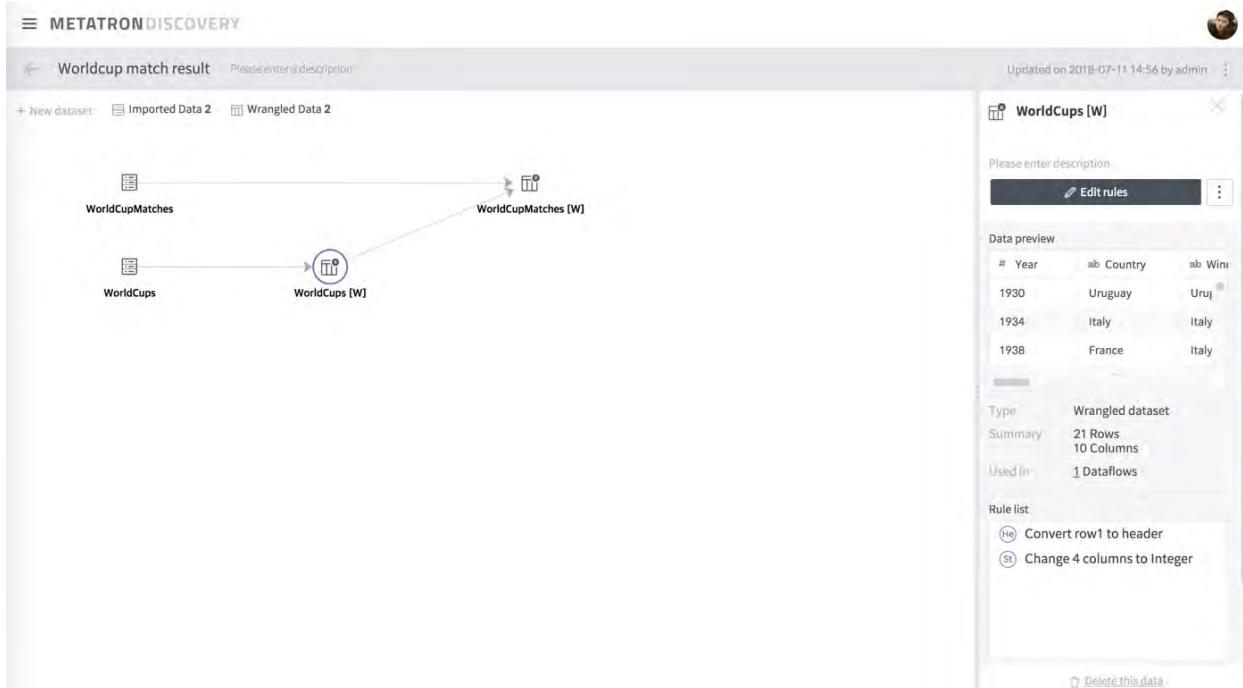
Metatron Discovery performs analytics on its ingested data sources or other external data sources using various analytical tools and outputs analytical results in charts and reports. To utilize this system, you

must understand its overall structure shown below:



2.2.1 Data Preparation

Data Preparation refines data from source data to be ingested into Metatron. See [Data Preparation](#) for details on data preparation.



This screenshot shows a detailed data preview for the dataset "Order_data [W]".

Data Preview: The preview shows a table with 100% valid data, 0% mismatched data, and 0% missing data. The columns listed are o_orderkey, o_custkey, o_orderpriority, o_totalprice, o_orderdate, and o_clerk.

Database Summary:

Database	Table	Summary	Size	Elapsed Time	Created
default	Order_list_Snapshot_1103	300,000,000 Rows 9 Columns	10 GB	0:1:11.0	2017-11-17 14:19:50

Summary: The summary section includes:

- Analyze order lists by customer (0:09)
- Dataset: Order_data [W] (19:44)
- Origin: Imported dataset (15:56)
- DataSource: Order_data (14:34)
- QueryStatement: SELECT * FROM tpch.orders (12:47)
- Created at: 2017-11-17 14:19:50 (42:46)
- Last modified at: 2017-11-17 14:19:50 (51:50)

2.2.2 Data Storage

Data Storage manages data ingested into the Metatron engine for analysis and visualization. See [Data Management](#) for details on data management.

Financial_data SBC Financial ERP Data

Updated on 2018-08-16 18:15 by Administrator

Information

Data Information

Source type:	FILE
Status:	Enabled (Datasource available via engine rules)
Size:	115.09 KB
Duration:	2013-09-01T00:00:00.000Z - 2013-09-01T00:00:00.000Z
Timestamp settings:	Segment Granularity: MONTH Query Granularity: NONE

Histogram

Detail

Permission

Allow all workspaces to use this datasource

All workspaces

Ingestion information

Master data

Type

excel

Create data connection

Please set required items and complete data connection creation

DB type

- Oracle** ✓
- MySQL
- PostgreSQL
- Hive
- presto
- APACHE PHOENIX
- Tibero

Server

Host	Port	SID
http://192.10.20.85	3306	
<input checked="" type="checkbox"/> URL only		
User ID for test	Password for test	
polaris	*****	

Security

- Always connect
- Connect by user's account
- Connect with ID and password

Validation Check Invalid Connection: Please check server and account information

Permission

1 Workspace [Edit](#)

Allow all workspaces to use this datasource

Advanced setting ↗

Socket timeout

Connection name

Previous **Next**

2.2.3 Data analysis and visualization

Each module below allows users to perform visualization-based exploration and analysis of stored data.

Workspace

Workspace provides an interface to manage its workbooks, workbenches, and notebooks used in an organization according to user access. See [Workspace](#) for details on the use of the workspace.

The screenshot shows the Metatron Discovery interface. At the top, there's a navigation bar with 'METATRON DISCOVERY' and a user profile icon. Below it, a header bar indicates 'Admin workspace' and 'Owner'. The main content area is titled 'Datasource (23)'. It features a search bar and filters for 'Show open data only' and 'Type: All'. A table lists 23 data sources with columns for No., Datasource name, Type, Used in, Full size, and Updated date. Each row includes an 'Open data' link. A 'Close' button is at the bottom of the modal.

No.	Datasource	Type	Used in	Full size	Updated
16	The_2014_Inc_5000	Ingested type	Open data	1.19 MB	2018-07-10
17	EMSI_JobChange_UK	Ingested type	Open data	46.73 KB	2018-07-10
18	OECD_TAX_ALL_02	Ingested type	Open data	926.70 KB	2018-07-09
19	WorldCup_Matches	Ingested type	Open data	69.31 KB	2018-07-06
20	oeecd_test	Ingested type	Open data	30.61 KB	2018-07-06
21	tour de france	Ingested type	Open data	27.94 KB	2018-07-06
22	cell_1h	Ingested type	2 Workspaces	90.79 MB	2018-07-06
23	FIFA_18_Player_Ratings	Ingested type	Open data	3.41 MB	2018-07-06

Workbook, dashboard, chart

Workbook supports working on, sharing, and making a presentation with dashboards and charts using a PowerPoint-like interface. See [Workbook](#) for details on the workbook module.

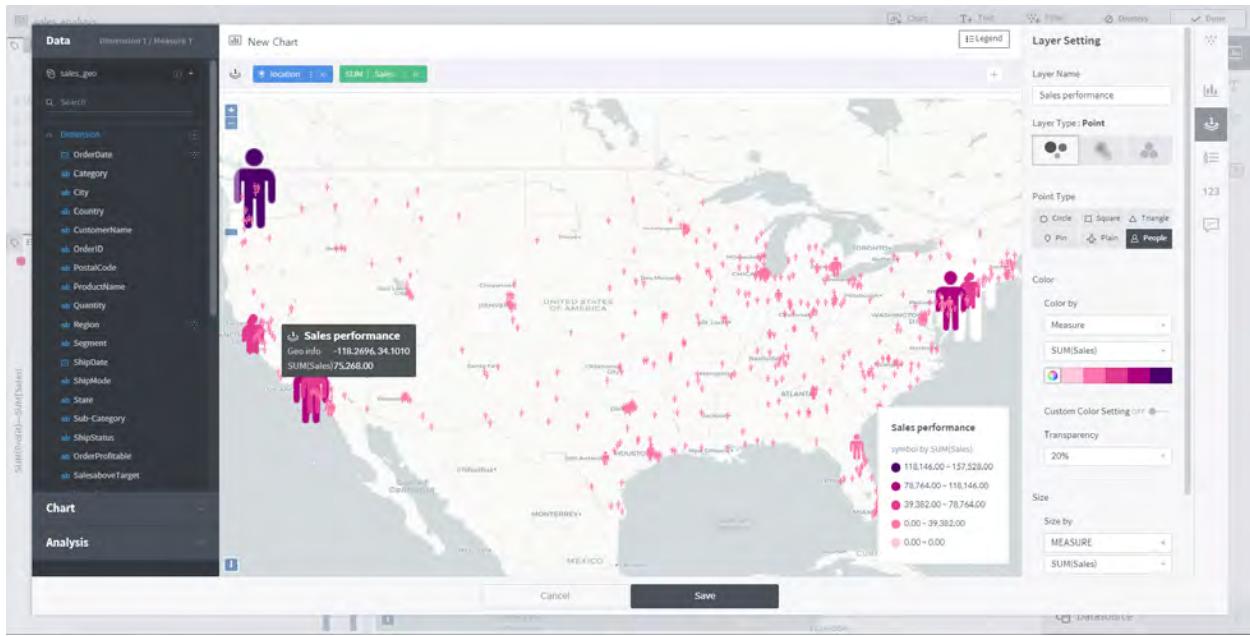
The screenshot shows the Metatron Discovery interface with a dashboard titled "Sales analysis". The dashboard includes:

- A line chart showing "COUNT(Sales)" over time from April 2012 to December 2014.
- A bubble chart showing "Profit" vs "Sales" for different categories like Technology, Office Supplies, Furniture, etc.
- A circular sunburst chart showing sales distribution by category.
- A data table with columns: OrderID, DAY(OrderDate), ProductName, Quantity, State, ShipMode, Sales, Profit, Day.

The screenshot shows a detailed view of a pie chart titled "Profit by City". The chart displays the following data:

City	Profit
New York City	1,000
Los Angeles	500
Seattle	300
San Francisco	200
San Diego	100
Springfield	50

The interface also includes a sidebar for "Data" and "Analysis", and a "Number Format" panel on the right.



Notebook

Notebook enables advanced analytics based on machine learning. See [Notebook](#) for details on the notebook module.

```
// 1. load dataset
import app.metatron.discovery.connector._;
val conf = new MetisClientSetting();
conf.setting("host", "metatron-web-01").setting("port", "8080");
val client = new MetisClient(conf);
val dataset = client.loadData(spark, "datasources", "ds-gis-37", "1000")

// 2. analyze
dataset.show()
```

Workbench

Workbench enables SQL data analytics. See [Workbench](#) for details on the workbench module.

The screenshot shows the Metatron Discovery interface. On the left, there's a sidebar titled 'Hive(2.3)' containing a 'Table list' section with various tables like 'contract', 'contract_part', 'contract_date_non_part_1', etc. The main area has a title 'METATRON DISCOVERY'. It contains a code editor with a query named '쿼리 01' and a results table below it.

```

    쿼리 01
1 SELECT A.C_ONE,
2          A.C_TWO,
3          SUM(A.C_TEN)
4     FROM TB_NUM AS A
5    WHERE A.C_ONE = 5
6   GROUP BY A.C_ONE, A.C_TWO;
7
8   USING 'GROUP BY' QUERY EXAMPLE
9 COMMENT ON TABLE USER_INFO_EX
10 IS '고객 정보 퀘리'; -- USER_INFO_EX 테이블에 주석 추가
11
12 SELECT *
13   FROM USER_TAB_COMMENTS
14  WHERE TABLE_NAME = 'USER_INFO_EX'; -- USER_INFO_EX 테이블의 주석 확인
15
16
17 COMMENT ON COLUMN USER_INFO_EX.RNAME
18 IS '고객 설계 이름'; -- USER_INFO_EX 의 RNAME 칼럼에 주석 추가
  
```

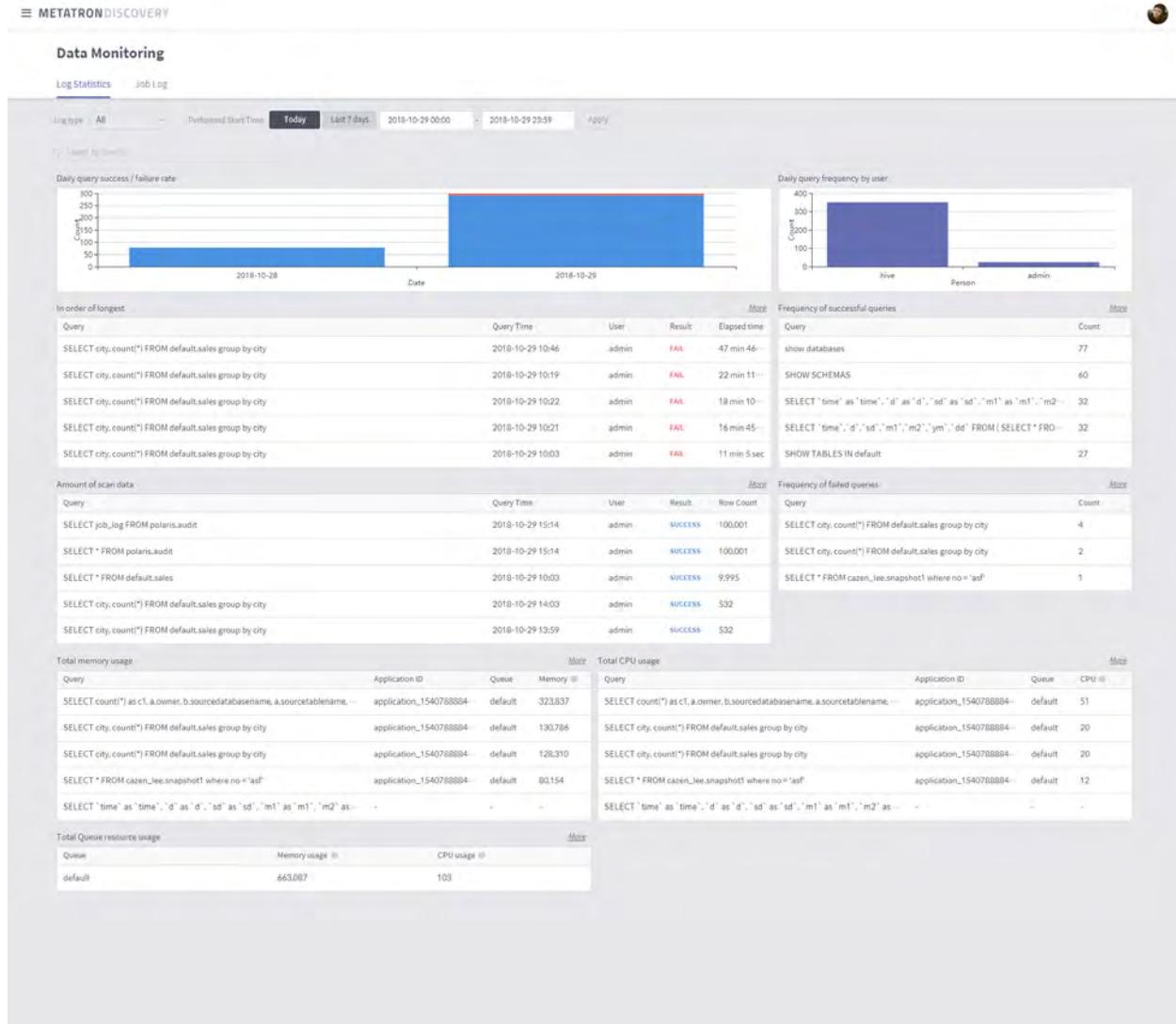
쿼리 01 - 결과 1

SEQ	L_orderkey	L_partkey	L_suppkey	L_linenumber	L_quantity	L_extendedprice	L_discount
1	1	31037869	1537885	1	17.0	30690.27	0.04
2	1	13461816	1461817	2	36.0	63977.04	0.09
3	1	12739956	739957	3	8.0	15962.56	0.1
4	1	426299	926300	4	28.0	34307.56	0.09

1000 Rows

2.2.4 Data Monitoring

This function monitors data use based on data query statistics and query logs. See [Data Monitoring](#) for details on the data monitoring functionality.



2.2.5 User permission and account administration

You can add/delete users or manage user permission.

2.2.6 Login/Logout

Users with accounts can login to Metatron Discovery and freely use within the assigned permission. Current login can be logged out from external systems as well.

2.3 Metatron engine: Druid

The development of information and communications technology has been accompanied by a rapid increase in the amount of data generated, highlighting the importance of efficient data collection, management, and utilization. However, RDBMS-based legacy tools are unable to process mass amounts of multidimensional data. This has led to the emergence of new methodologies and solutions aimed at satisfying the demand for big data.

Metamarkets, a technology startup based in Silicon Valley, launched a column-oriented distributed data store known as Druid in 2011, and open sourced it in October 2012. Many companies have turned to Druid for their backend technology because it offers various advantages, including fast and efficient data processing.

As a B2C telecommunications service provider, SK Telecom recognized the need to effectively manage and analyze the vast amounts of network data generated by its users every minute. Metatron, an end-to-end business intelligence solution with Druid as the underlying engine, was thus developed and launched in 2016.



The following sections discuss the features of Druid that make it suitable for time-series data processing, and introduce how they were adapted and improved by SK Telecom for Metatron.

2.3.1 Background of Druid development

Druid was originally designed to satisfy the following needs around ingesting and exploring large quantities of transactional events (log data):

- The developers wanted to be able to rapidly and arbitrarily slice and dice data and drill into that data effectively without any restrictions, along with sub-second queries over any arbitrary combination of dimensions. These capabilities were needed to allow users of their data dashboard to arbitrarily and interactively explore and visualize event streams.
- The developers wanted to be able to ingest events and make them exportable almost immediately after their occurrence. This was crucial to enable users to collect and analyze data in real time for timely situational assessments, predictions and business decisions. Popular open source data warehousing systems such as Hadoop were unable to provide the sub-second data ingestion latencies as required.
- The developers wanted to ensure multitenancy and high availability for their solution services. Their systems needed to be constantly up and be able to withstand all sorts of potential failures without going down or taking any downtime. Downtime is costly and many businesses cannot afford to wait if a system is unavailable in the face of software upgrades or network failure.

2.3.2 Druid features

Data table components

Data tables in Druid (called data sources) are collections of timestamped events designed for OLAP queries. A data source is composed of three distinct types of columns (here we use an example dataset from online advertising).

Timestamp column	Dimension columns					Metric columns	
timestamp	publisher	advertiser	gender	country	click	price	
2011-01-01T01:01:35Z	bieberfever.com	google.com	Male	USA	0	0.65	
2011-01-01T01:03:63Z	bieberfever.com	google.com	Male	USA	0	0.62	
2011-01-01T01:04:51Z	bieberfever.com	google.com	Male	USA	1	0.45	
2011-01-01T01:00:00Z	ultratrimfast.com	google.com	Female	UK	0	0.87	
2011-01-01T02:00:00Z	ultratrimfast.com	google.com	Female	UK	0	0.99	
2011-01-01T02:00:00Z	ultratrimfast.com	google.com	Female	UK	1	1.53	

Fig. 1: Source: <http://druid.io>

- **Timestamp column:** Druid treats timestamp separately in a data source because all its queries center around the time axis (If non-time series data is ingested in batch, all records are timestamped with the current time for use in Druid).
- **Dimension columns:** Dimensions are string attributes of an event, and the columns most commonly used in filtering the data. Four dimensions are involved in the example dataset: publisher, advertiser, gender, and country. They each represent an axis of the data chosen to slice across.
- **Metric columns:** Metrics are columns used in aggregations and computations. In the example, the metrics are clicks and price. Metrics are usually numeric values, and computations include operations such as count, sum, and mean (Metatron has extended supported Druid data types).

Data ingestion

Druid supports real-time and batch ingestion.

One major characteristic of Druid is real-time ingestion, which is enabled by real-time nodes (For details, see [Real-time nodes](#)). Events ingested in real-time from a data stream get indexed in seconds to become queryable in the Druid cluster.

Data roll-up

The individual events in our example dataset are not very interesting because there may be trillions of such events. However, summarizations of this type of data by time interval can yield many useful insights.

Druid summarizes this raw data when ingesting it using an optional process called “roll-up.” Below is an example of roll-up.

The diagram illustrates the roll-up process. On the left, a table shows raw domain click events from January 1, 2011, at various times between 00:01:35Z and 01:00:53Z. The columns are timestamp, domain, gender, and clicked. On the right, an arrow points to a summary table showing the total number of clicks for each gender across one-hour intervals (00:00:00Z to 01:00:00Z). The summary table has columns timestamp, domain, gender, and clicked.

timestamp	domain	gender	clicked
2011-01-01T00:01:35Z	bieber.com	Female	1
2011-01-01T00:03:03Z	bieber.com	Female	0
2011-01-01T00:04:51Z	ultra.com	Male	1
2011-01-01T00:05:33Z	ultra.com	Male	1
2011-01-01T00:05:53Z	ultra.com	Female	0
2011-01-01T00:06:17Z	ultra.com	Female	1
2011-01-01T00:23:15Z	bieber.com	Female	0
2011-01-01T00:38:51Z	ultra.com	Male	1
2011-01-01T00:49:33Z	bieber.com	Female	1
2011-01-01T00:49:53Z	ultra.com	Female	0

timestamp	domain	gender	clicked
2011-01-01T00:00:00Z	bieber.com	Female	1
2011-01-01T00:00:00Z	ultra.com	Female	2
2011-01-01T00:00:00Z	ultra.com	Male	3

Fig. 2: Source: Interactive Exploratory Analytics with Druid | DataEngConf SF ‘17

The table on the left lists the domain click events that occurred from 00:00:00 to 01:00:00 on January 1, 2011. Since individual events recorded in seconds do not have much significance from the analyst’s perspective, the data was compiled at a granularity of one hour. This results in the more meaningful table on the right, which shows the number of clicks by gender for the same time period.

In practice, rolling up data can dramatically reduce the size of data that needs to be stored (up to a factor of 100), thereby saving on storage resources and enabling faster queries.

But, as data is rolled up, individual events can no longer be queried; the rollup granularity is the minimum granularity you will be able to explore data at and events are floored to this granularity. The unit of granularity can be set as desired by users. If necessary, the roll-up process may be disabled to ingest every individual event.

Data sharding

A data source is a collection of timestamped events and partitioned into a set of shards. A shard is called a segment in Druid and each segment is typically 5? 10 million rows. Druid partitions its data sources into well-defined time intervals, typically an hour or a day, and may further partition on values from other columns to achieve the desired segment size.

The example below shows a data table segmented by hour:

Segment sampleData_2011-01-01T01:00:00:00Z_2011-01-01T02:00:00:00Z_v1_0:

2011-01-01T01:00:00Z	ultratrimfast.com	google.com	Male	USA	1800	25	15.70
2011-01-01T01:00:00Z	bieberfever.com	google.com	Male	USA	2912	42	29.18

Segment sampleData_2011-01-01T02:00:00:00Z_2011-01-01T03:00:00:00Z_v1_0:

2011-01-01T02:00:00Z	ultratrimfast.com	google.com	Male	UK	1953	17	17.31
2011-01-01T02:00:00Z	bieberfever.com	google.com	Male	UK	3194	170	34.01

This segmentation by time can be achieved because every single event in a data source is timestamped.

Segments represent the fundamental storage unit in Druid and replication and distribution are done at a segment level. They are designed to be immutable, which means that once a segment is created, it cannot be edited. This ensures no contention between reads and writes. Druid segments are just designed to be read very fast.

In addition, this data segmentation is key to parallel processing in Druid's distributed environment: As one CPU can scan one segment at a time, data partitioned into multiple segments can be scanned by multiple CPUs simultaneously in parallel, thereby ensuring fast query returns and stable load balancing.

Data storage format and indexing

The way Druid stores data contributes to its data structures highly optimized for analytic queries. This section uses the Druid table below as an example:

Timestamp	Page	Username	Gender	City	Characters Added	Characters Removed
2011-01-01T01:00:00Z	Justin Bieber	Boxer	Male	San Francisco	1800	25
2011-01-01T01:00:00Z	Justin Bieber	Reach	Male	Waterloo	2912	42
2011-01-01T02:00:00Z	Ke\$ha	Helz	Male	Calgary	1953	17
2011-01-01T02:00:00Z	Ke\$ha	Xeno	Male	Taiyuan	3194	170

Fig. 3: Source: Druid: A Real-time Analytical Data Store

Columnar storage and indexing

Druid is a column store, which means each individual column is stored separately. Given that Druid is best used for aggregating event streams, column storage allows for more efficient CPU usage as only the columns pertaining to a query are actually loaded and scanned in that query. In a row oriented data store, all columns associated with a row must be scanned as part of an aggregation. The additional scan time can introduce significant performance degradations. In the example above, the page, user, gender, and city columns only contain strings. Storing strings directly is unnecessarily costly; instead, they can be mapped into unique integer identifiers. For example,

```
Justin Bieber -> 0
Ke$ha -> 1
```

This mapping allows the page column to be represented as an integer array where the array indices correspond to the rows of the original dataset. For the page column, we can represent the unique pages as follows:

```
[0, 0, 1, 1]
```

Thus, strings are replaced by fixed-length integers in storage, which are much easier to compress. Druid indexes data on a per-shard (segment) level.

Indices for filtering data

Druid creates additional lookup indices that facilitate filtering on string columns. Let us consider the above example table again. A query might be: “How many Wikipedia edits were done by users in San Francisco who are also male?” This example query involves two dimensions: City (San Francisco) and Gender (Male). For each dimension, a binary array is created where the array indices represent whether or not their corresponding rows match the query filter, as shown below:

```
San Francisco (City) -> rows [1] -> [1][0][0][0]
Male (Gender) -> rows [1, 2, 3, 4] -> [1][1][1][1]
```

And the query filter performs the AND operation between the two arrays:

```
[1][0][0][0] AND [1][1][1][1] = [1][0][0][0]
```

As a result, only row 1 is subject to scanning, which retrieves only the filtered rows and eliminates unnecessary workload. And these binary arrays are very easy to compress as well.

This lookup can be used for the OR operation too. If a query filters on San Francisco or Calgary, array indices will be for each dimension value:

```
San Francisco (City) -> rows [1] -> [1][0][0][0]
Calgary (City) -> rows [3] -> [0][0][1][0]
```

And then the OR operation is performed on the two arrays:

```
[1][0][0][0] OR [0][0][1][0] = [1][0][1][0]
```

Thus the query scans rows 1 and 3 only.

This approach of performing Boolean operations on large bitmap sets is commonly used in search engines.

Query languages

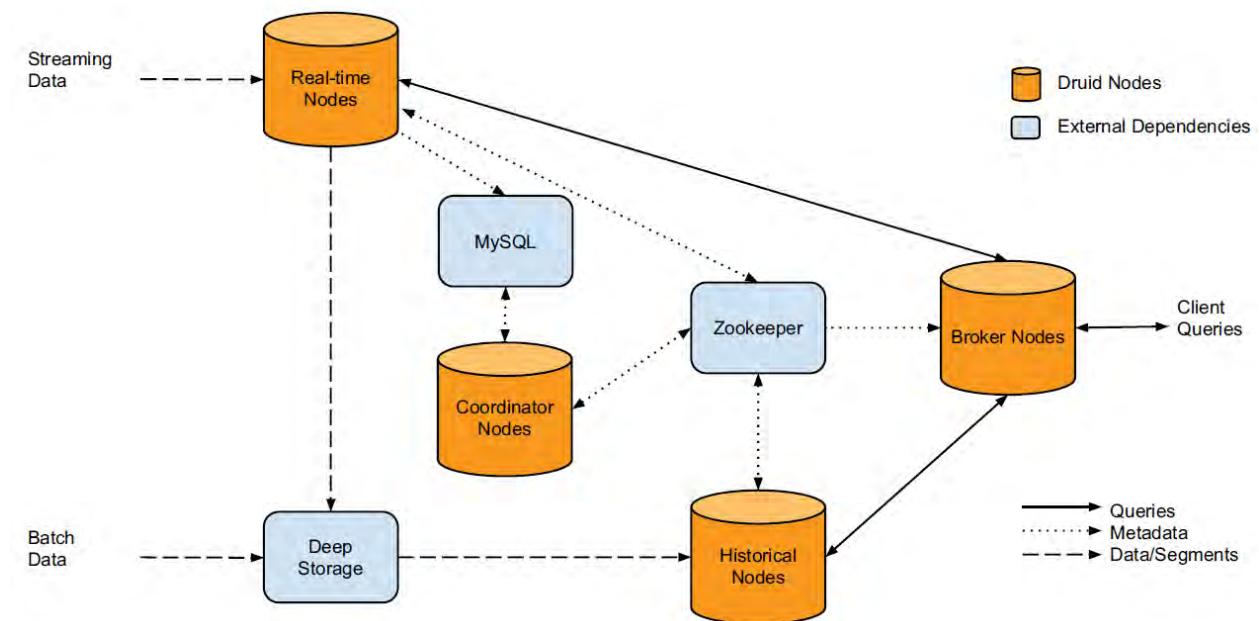
Druid's native query language is JSON over HTTP. Druid queries include:

- Group By
- Time-series roll-ups
- Arbitrary Boolean filters
- Sum, Min, Max, Avg and other aggregation functions
- Dimensional Search

In addition to these, query libraries in numerous languages, including SQL, are developed and shared.

2.3.3 Druid cluster architecture

A Druid cluster consists of different types of nodes and each node type is designed to perform a specific set of things:



Real-time nodes

Real-time nodes function to ingest and query event streams. The nodes are only concerned with events for some small time range and periodically hand them off to the deep storage in the following steps:

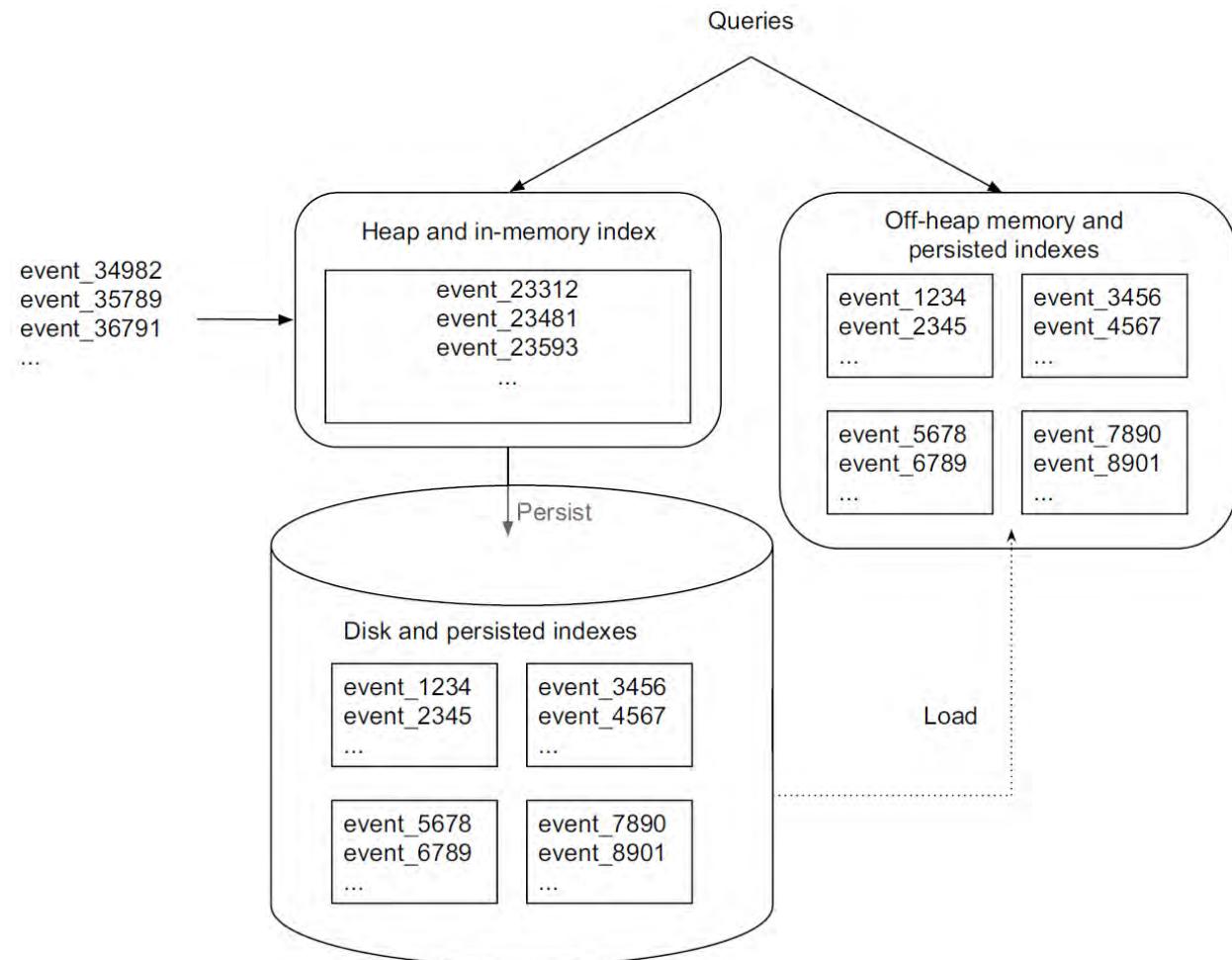


Fig. 4: Source: Druid: A Real-time Analytical Data Store

1. Incoming events are indexed in memory and immediately become available for querying.
2. The in-memory data is regularly persisted to disk and converted into an immutable, columnar storage format.
3. The persisted data is loaded into off-heap memory to be still queryable.
4. On a periodic basis, the persisted indexes are merged together to form a “segment” of

data and then get handed off to deep storage.

In this way, all events ingested into real-time nodes, regardless before or after persisted, are present in memory (either on- or off-heap) and thus can be queried (queries hit both the in-memory and persisted indexes). This functionality of real-time nodes enables Druid to conduct real-time data ingestion meaning that events can be queried almost as soon as they occur. In addition, there is no data loss during these steps. In addition, there is no data loss during these steps.

Real-time nodes announce their online state and the data they serve in Zookeeper (see [External dependencies](#)) for the purpose of coordination with the rest of the Druid cluster.

Historical nodes

Historical nodes function to load and serve the immutable blocks of data (segments) created by real-time nodes. These nodes download immutable segments locally from the deep storage and serve queries over those segments (e.g., data aggregation/filtering). The nodes are operationally simple based on a shared-nothing architecture; they have no single point of contention and simply load, drop, and serve segments as instructed by Zookeeper.

A historical node's process of serving a query is as follows:

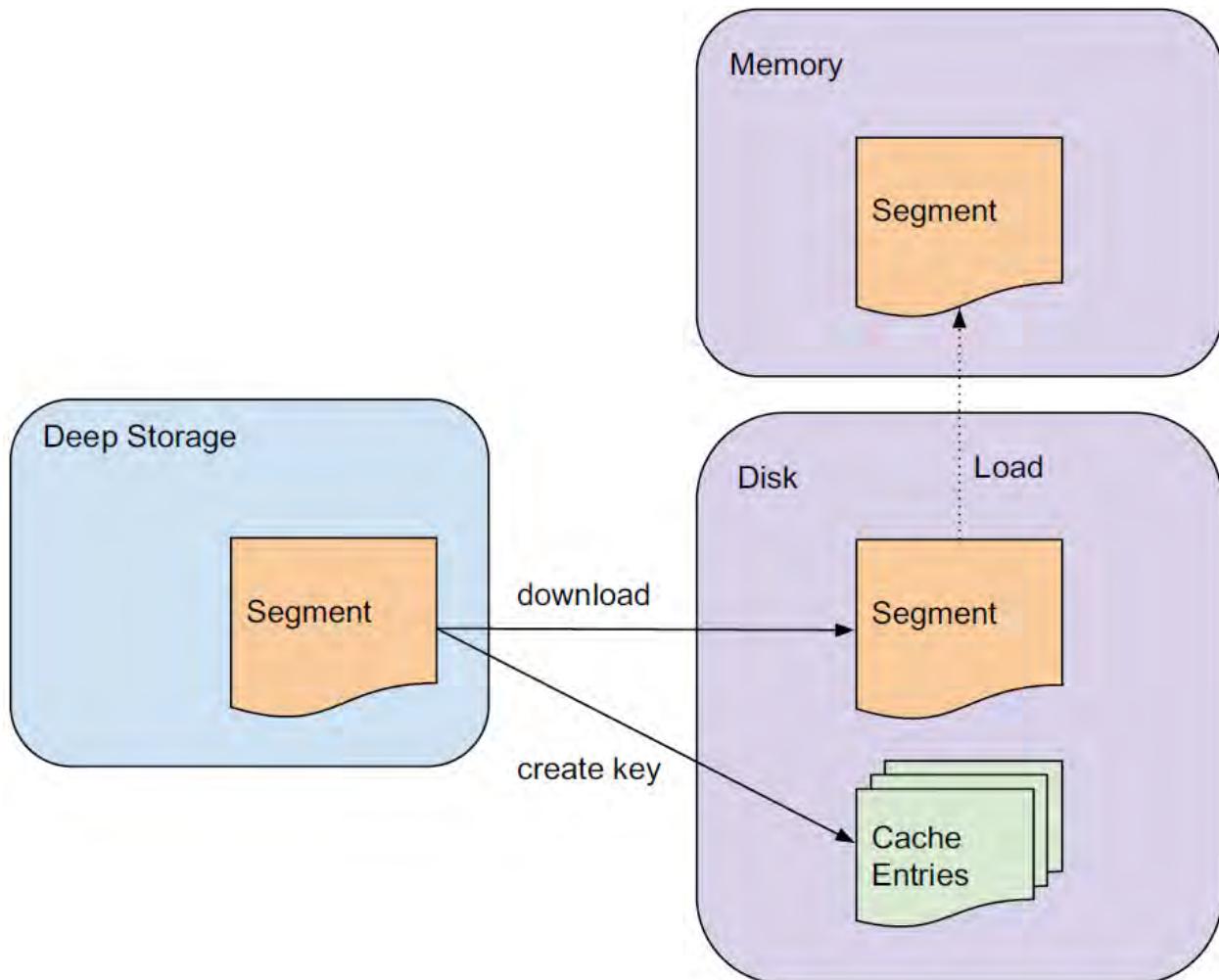


Fig. 5: Source: Druid: A Real-time Analytical Data Store

Once a query is received, the historical node first checks a local cache that maintains information about what segments already exist on the node. If information about a segment in question is not present in the cache, the node will proceed to download the segment from deep storage. On the completion of the processing, the segment is announced in Zookeeper to become queryable and the node performs the requested query on the segment.

Historical nodes can support read consistency because they only deal with immutable data. Immutable data blocks also enable a simple parallelization model: historical nodes can concurrently scan and aggregate immutable blocks without blocking.

Similar to real-time nodes, historical nodes announce their online state and the data they are serving in Zookeeper.

Broker nodes

Broker nodes understand the metadata published in Zookeeper about what segments are queryable and where those segments are located. Broker nodes route incoming queries such that the queries hit the right historical or real-time nodes. Broker nodes also merge partial results from historical and real-time nodes before returning a final consolidated result to the caller.

Broker nodes use a cache for resource efficiency as follows:

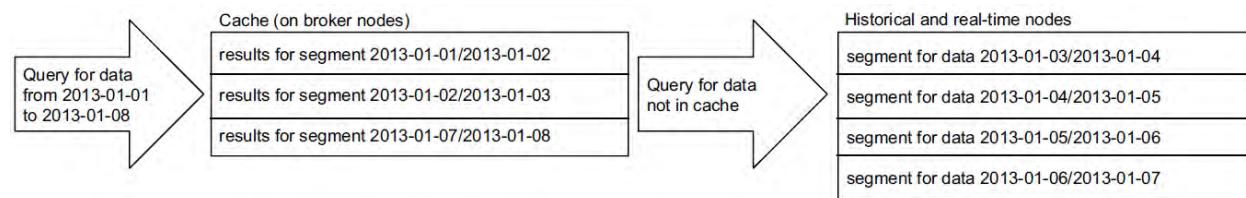


Fig. 6: Source: Druid: A Real-time Analytical Data Store

Once a broker node receives a query involving a number of segments, it checks for segments already existing in the cache. For any segments absent in the cache, the broker node will forward the query to the correct historical and real-time nodes. Once historical nodes return their results, the broker will cache these results on a per-segment basis for future use. Real-time data is never cached and hence requests for real-time data will always be forwarded to real-time nodes. Since real-time data is perpetually changing, caching the results is unreliable.

Coordinator nodes

Coordinator nodes are primarily in charge of data management and distribution on historical nodes. The coordinator nodes determine which historical nodes perform queries on which segments and tell them to load new data, drop outdated data, replicate data, and move data to load balance. This enables fast, efficient, and stable data processing in a distributed group of historical nodes.

As with all Druid nodes, coordinator nodes maintain a Zookeeper connection for current cluster information. Coordinator nodes also maintain a connection to a MySQL database that contains additional operational parameters and configurations, including a rule table that governs how segments are created, destroyed, and replicated in the cluster.

Coordinator nodes undergo a leader-election process that determines a single node that runs the coordinator functionality. The remaining coordinator nodes act as redundant backups.

External dependencies

Druid has a couple of external dependencies for cluster operations.

- **Zookeeper:** Druid relies on Zookeeper for intra-cluster communication.
- **Metadata storage:** Druid relies on a metadata storage to store metadata about segments and configuration. MySQL and PostgreSQL are popular metadata stores for production.
- **Deep storage:** Deep storage acts as a permanent backup of segments. Services that create segments upload segments to deep storage and historical nodes download segments from deep storage. S3 and HDFS are popular deep storages.

High availability characteristics

Druid is designed to have no single point of failure. The different node types operate fairly independent of each other and there is minimal interaction among them. Hence, intra-cluster communication failures have minimal impact on data availability. To run a highly available Druid cluster, you should have at least two nodes of every node type running.

Architecture extensibility

Druid features a modular, extensible platform that allows various external modules to be added to its basic architecture. An example of how Druid's architecture can be extended with modules is shown below:

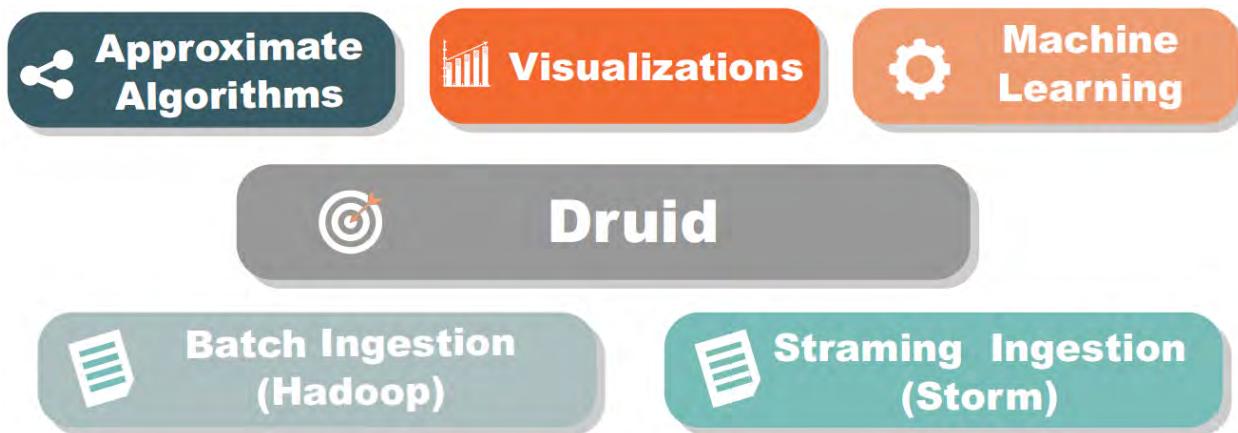


Fig. 7: Source: MetaMarkets – Introduction to Druid by Fangjin Yang

Metatron, an end-to-end business intelligence solution to be introduced in this paper, was also built by adding various modules to the Druid engine.

2.3.4 Druid performance assessments

With Druid being a data store that supports real-time data exploration, its quantitative assessments are focused on two key aspects:

- Query latency
- Ingestion latency

This is because the key to achieving “real-time” performance is to minimize the time spent on query processing and ingestion. A number of organizations and individuals, including the developers of Druid, have established benchmarks for Druid performance assessment based on the two key aspects, and shared how Druid compares to other database management systems.

Self-assessment by Druid developers

Druid: A Real-time Analytical Data Store¹ was published by the developers in 2014. Chapter 6. Performance contains details of Druid assessment, with a particular focus on query and ingestion latencies. The benchmarks of Druid performance are briefly introduced in the following sections.

Query latency

Regarding Druid’s query latency, the paper discusses two performance assessments? one was conducted on eight data sources that had been most queried at Metamarkets and the other was on TPC-H datasets. In this section, we review the latter assessment. The latencies from querying on TPC-H datasets were measured by comparing with MySQL, and the cluster environment was as follows:

- **Druid historical nodes:** Amazon EC2 m3.2xlarge instance types (Intel® Xeon® E5-2680 v2 @ 2.80GHz)
- **Druid broker nodes:** c3.2xlarge instances (Intel® Xeon® E5-2670 v2 @ 2.50GHz)
- **Pledged mountain draw converting** (subtract soft a3.2analysed repurchase pairs)

¹

F. Yang, E. Tschetter, X. Léauté, N. Ray, G. Merlino, and D. Ganguli. (2014). Druid: a real-time analytical data store. Retrieved from <http://druid.io/docs/0.12.1/design/index.html>.

The figure below shows the query latencies resulting from Druid and MySQL when tested on the 1GB and 100GB TPC-H datasets:

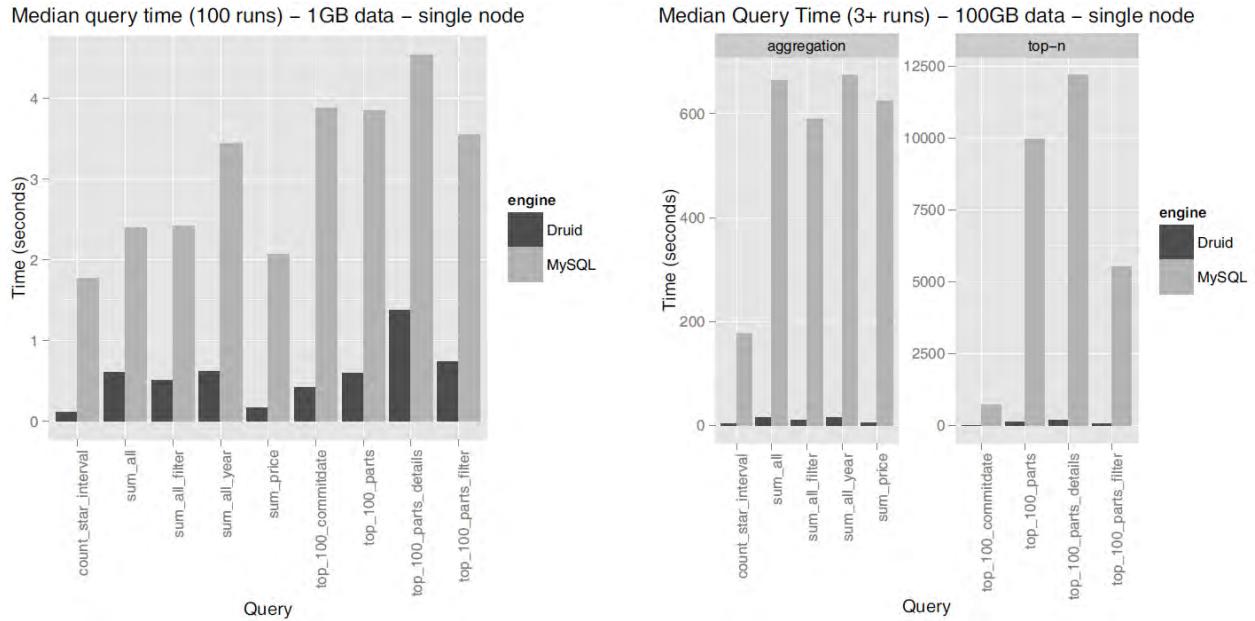


Fig. 8: Source: Druid: A Real-time Analytical Data Store

By showcasing these results, the paper suggests that Druid is capable of extremely faster query returns compared to legacy relational database systems.

The Druid paper also presents how faster query returns are achieved when multiple nodes are joined together in a cluster. When tested on the TPC-H 100 GB dataset, the performance difference between a single node (8 cores) and six-node cluster (48 cores) was as follows:

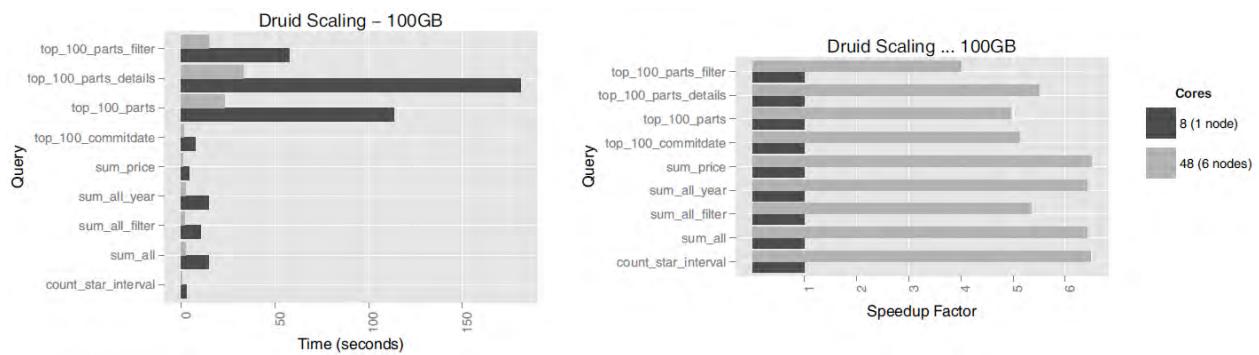


Fig. 9: Source: Druid: A Real-time Analytical Data Store

It was observed that not all types of queries achieve linear scaling, but the simpler aggregation queries do, ensuring a speed increment almost proportional to the number of the cores (SK Telecom's Metatron has made improvements to achieve much more obvious linear scalability).

Ingestion latency

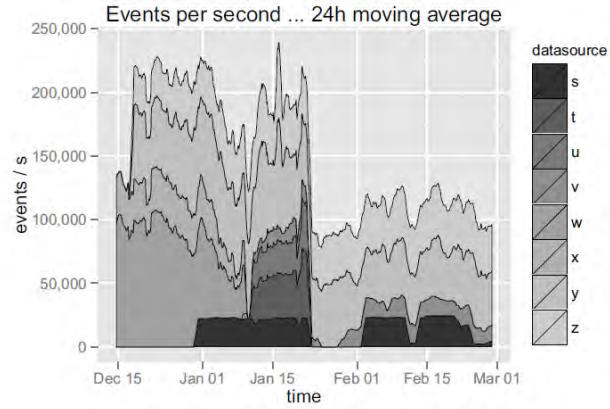
The paper also assessed Druid's data ingestion latency on a production ingestion setup consisting of:

- 6 nodes, totalling 360GB of RAM and 96 cores (12 x Intel®Xeon®E5-2670).

A total of eight production data sources were selected for this assessment. The characteristics of each data source and their ingestion results are shown below. Note that in this setup, several other data sources were being ingested and many other Druid related ingestion tasks were running concurrently on the machines.

Data Source	Dimensions	Metrics	Peak events/s
s	7	2	28334.60
t	10	7	68808.70
u	5	1	49933.93
v	30	10	22240.45
w	35	14	135763.17
x	28	6	46525.85
y	33	24	162462.41
z	33	24	95747.74

Ingestion characteristics of various data sources



Combined cluster ingestion rates

Fig. 10: Source: Druid: A Real-time Analytical Data Store

Druid's data ingestion latency is heavily dependent on the complexity of the dataset being ingested, but the latency measurements present here are sufficient to demonstrate that Druid well addresses the stated problems of interactivity.

Druid performance assessment by SK Telecom

SK Telecom also measured the query and ingestion latencies of Druid as detailed below:

Query latency test

The conditions of query latency measurement were as follows:

- Data: TPC-H 100G dataset (900 million rows)
- Pre-aggregation granularity: day
- Servers: r3.4xlarge nodes, (2.5GHz * 16, 122G, 320G SSD) * 6
- No. of historical nodes: 6
- No. of broker nodes: 1

The query times for five queries of the TPC-H 100G dataset were as follows (the query times in Hive were also measured as a reference):

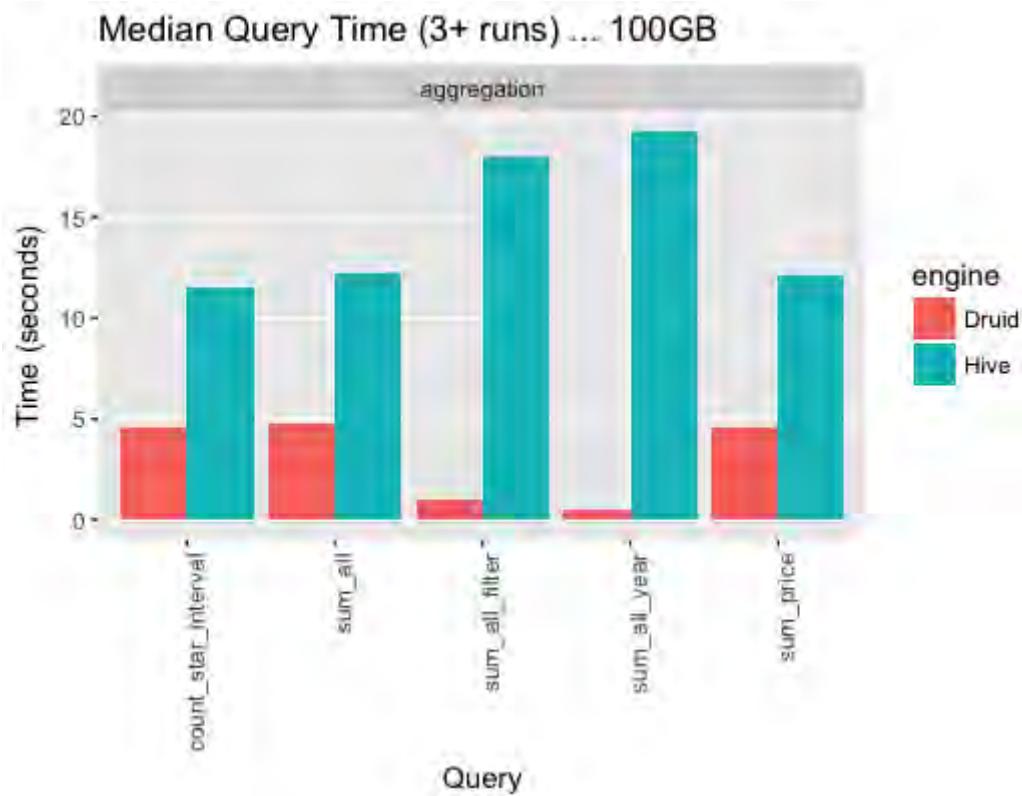


Fig. 11: Source: SK Telecom T-DE WIKI Metatron Project

Note: The reasons why the Hive benchmark performed poorly include that some processes

were performed through Thrift and the dataset wasn't partitioned.

Ingestion latency test

The conditions of ingestion latency measurement were as follows:

- Ingestion data size: 30 million rows/day, 10 columns
- Memory: 512 GB
- CPU: Intel (R) Xeon (R) Gold 5120 CPU @ 2.20 GHz (56 cores)
- No. of historical nodes: 100
- No. of broker nodes: 2
- Jobs performed by three out of ten middle-manager nodes
- Ingestion tool: Apache Kafka

Data ingestion was performed 100 times under the conditions specified above, and the average ingestion latency was 1.623439 seconds. As illustrated below, ingestion latency was computed as the sum of Kafka ingestion latency, Druid ingestion latency, and Druid query latency.

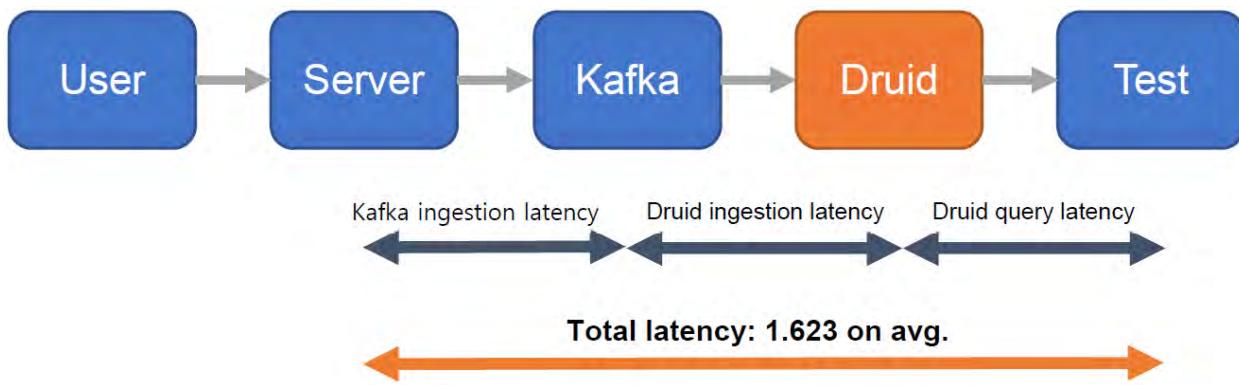


Fig. 12: Source: SK Telecom T-DE WIKI Metatron Project

Druid assessments by third parties

Druid assessment by Outlyer

In the Outlyer blog, twenty open source time-series database systems were assessed in a post² titled Top 10 Time Series Databases and published on August 26, 2016. The author Steven Acreman ranked Druid in the 8th place, and his set of criteria was as follows:

Table 1: A summary of Druid assessment by Outlyer

Items	Druid performance
Write performance - single node	25k metrics/sec Source: https://groups.google.com/forum/#!searchin/druid-user/benchmark%7Csort:relevance/druid-user/90BMCxz22Ko/73D8HidLCgAJ
Write performance - 5-node cluster	100k metrics / sec (calculated)
Query performance	Moderate
Maturity	Stable
Pro's	Good data model and cool set of analytics features. Mostly designed for fast queries over large batch loaded datasets which it's great at.
Con's	Painful to operate, not very fast write throughput. Real time ingestion is tricky to setup.

Druid assessment by DB-Engines

DB-Engines³, an online website, publishes a list of database management systems ranked by their current popularity every months. To measure the popularity of a system, it uses the following parameters:

- Number of mentions of the system on websites: It is measured as the number of results in queries of the search engines Google, Bing and Yandex.
- General interest in the system: For this measurement, the frequency of searches in Google Trends is used.
- Frequency of technical discussions about the system: The ranking list uses the number of related questions and the number of interested users on the well-known IT-related Q&A sites Stack Overflow and DBA Stack Exchange.

² Steven Acreman. (2016, Aug 26). Top 10 Time Series Databases. Retrieved from <https://blog.outlyer.com/top10-open-source-time-series-databases>.

³ DB-Engines website. <https://db-engines.com>, July 2018.

- Number of job offers, in which the system is mentioned: The ranking list uses the number of offers on the leading job search engines Indeed and Simply Hired.
- Number of profiles in professional networks, in which the system is mentioned: The ranking list uses the internationally most popular professional networks LinkedIn and Upwork.
- Relevance in social networks. The ranking list counts the number of Twitter tweets, in which the system is mentioned.

As of July 2018, Druid ranked 118th out of a total of 343 systems, and 7th out of 25 time-series database systems.

Comparison with Apache Spark

Comparing Druid with Apache Spark is meaningful because both technologies are emerging as next-generation solutions for large-scale analytics and their different advantages make them very complementary when combined together. Metatron also makes use of this combination: Druid as the data storage/processing engine and Spark as an advanced analytics module.

This section briefly introduces a report comparing the performance of Druid and Spark⁴⁵ published by Harish Butani, the founder of Sparkline Data Inc. Prior to the performance comparison, the report states that the two solutions are in complementary relations, rather than competitors.

Apache Spark characteristics

Apache Spark is an open-source cluster computing framework providing rich APIs in Java, Scala, Python, and R. Spark's programming model is used to build analytical solutions that combine SQL, machine learning, and graph processing. Spark supports powerful functions to process large-scale and/or complex data manipulation workflows, but it isn't necessarily optimized for interactive queries.

Dataset, queries, performance results

For the benchmark, the 10G TPC-H dataset was used. The 10G star schema was converted into a flattened (denormalized) transaction dataset and reorganized to be queryable in Druid and Spark. The sizes of the resulting datasets were:

⁴ Harish Butani. (2018, Sep 18). Combining Druid and Spark: Interactive and Flexible Analytics at Scale. Retrieved from <https://www.linkedin.com/pulse/combining-druid-spark-interactiveflexible-analytics-scale-butani>.

⁵ Harish Butani. (2015, Aug 28). TPCH Benchmark. Retrieved from <https://github.com/SparklineData/spark-druid-olap/blob/master/docs/benchmark/BenchMarkDetails.pdf>.

- TPCH Flat TSV: 46.80GB
- Druid Index in HDFS: 17.04GB
- TPCH Flat Parquet: 11.38GB
- TPCH Flat Parquet Partition by Month: 11.56GB

And then, a number of queries were chosen to test the performance differences in various aspects as shown below:

Table 2: Queries used for query latency comparison between
Druid and Apache Spark

Query	Interval	Filters	Group By	Aggregations
Basic Aggregation.	None	None	ReturnFlag LineStatus	Count(*) Sum(exdPrice) Avg(avlQty)
Ship Date Range	1995-12/1997-09	None	ReturnFlag LineStatus	Count(*)
SubQry Nation, pType ShpDt Range	1995-12/1997-09	P_Type S_Nation + C_Nation	S_Nation	Count(*) Sum(exdPrice) Max(sCost) Avg(avlQty) Count(Distinct oKey)
TPCH Q1	None	None	ReturnFlag LineStatus	Count(*) Sum(exdPrice) Max(sCost) Avg(avlQty) Count(Distinct oKey)
TPCH Q3	1995-03-15-	O_Date MktSegment	Okey Odate ShipPri	Sum(exdPrice)
TPCH Q5	None	O_Date Region	S_Nation	Sum(exdPrice)
TPCH Q7	None	S_Nation + C_Nation	S_Nation C_Nation ShipDate.Year	Sum(exdPrice)
TPCH Q8	None	Region Type O_Date	ODate.Year	Sum(exdPrice)

The test results are as follows:

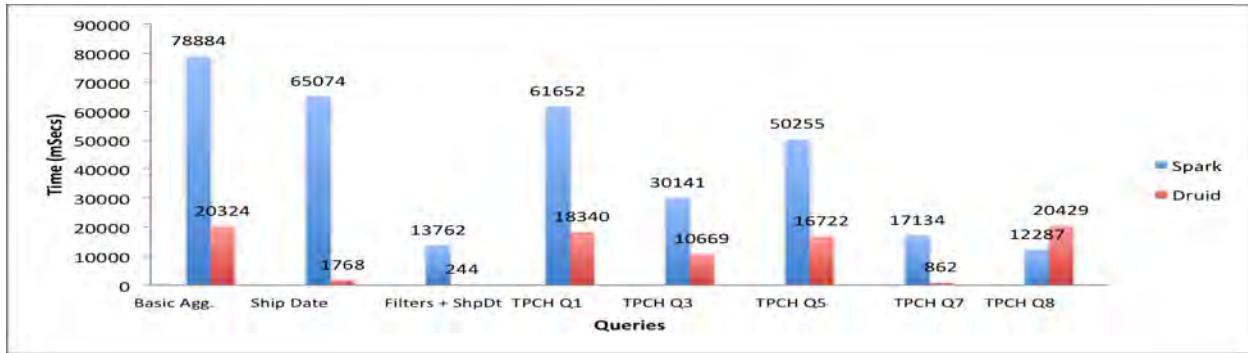


Fig. 13: Source: Combining Druid and Spark: Interactive and Flexible Analytics at Scale

- The Filters + Ship Date query provides the greatest performance gain (over 50 times over Spark) when Druid is used. This is not surprising as this query is a typical slice-and-dice query tailor-made for Druid. Along the same lines, TPCH Q7 shows a significant performance boost when running on Druid: milliseconds on Druid vs. 10s of seconds on Spark.
- For TPCH Q3, Q5, and Q8 there is an improvement, but not to the same level as Q7. This is because the OrderDate predicate is translated to a JavaScript filter in Druid, which is significantly slower than a native Java filter.
- The Basic Aggregation and TPCH Q1 queries definitely show improvement. The Count-Distinct operation is translated to a cardinality aggregator in Druid, which is an approximate count. This is definitely an advantage for Druid, especially for large cardinality dimensions.

These results can vary with testing conditions, but one thing is clear: Queries that have time partitioning or dimensional predicates (like those commonly found in OLAP workflows) are significantly faster in Druid.

Implications

The testing results showcase that combining the analytic capabilities with Spark and the OLAP and low latency capabilities of Druid can create great synergy. Druid ingests, explores, filters, and aggregates data efficiently and interactively, while the rich programming APIs of Spark enable in-depth analytics. By leveraging these different capabilities, we can build a more powerful, flexible, and extremely low latency analytics solution.

References

2.3.5 Metatron powered by Druid

As explained previously, Metatron employs Druid as its underlying engine and has made developments and improvements of Druid for its own uses. This section introduces the background, progress, and results of the adoption of Druid to Metatron.

Metatron development background and Druid integration

Metatron as a big data analytics solution

As a telecommunications service provider with the most number of subscribers in South Korea, SK Telecom has exerted significant efforts to establish a stable network environment through by using the mass amounts of network data logs generated by its users.

Due to the limitations of existing IT infrastructure in mass data processing, SK Telecom needed a big-data warehousing system (Apache Hadoop) and a big-data analytics solution compatible with the system. The company built its own Hadoop infrastructure to store mass amounts of data at low cost, but faced the following limitations:

- Network data generated by the countless users could not be analyzed in real time. Although it was possible to store and process big data, visualizations could be implemented only with a sampled subset of data in the same way as on legacy systems.
- Having different solutions and different managers support each stage of data analytics, such as ETL, DW, and BI, not only involved significant time and costs, but also resulted in poor data accessibility. An end-to-end solution was needed to analyze all stages at once in a simple and quick manner.

Why the Druid engine

Druid was the optimal engine for the Metatron solution because it fulfilled the aforementioned needs with the features below:

- Druid collects mass amounts of data in real time and indexes them into a queryable format, ensuring very fast data aggregations (a few seconds at the slowest) based on distributed processing.
- Druid's OLAP time-series data format enables analysts to perform data exploration, filtering, and visualization as desired. Such free and flexible data exploration is essential for users to intuitively select the required data and determine correlations between different dimensions on it.

- Druid's extensible architecture allows modules to be easily added.

Built on this architecture, Metatron is an end-to-end solution that embraces all layers of data collection, storage, processing, analysis, and visualization.

Druid engine integration

The Druid engine was integrated in Metatron as follows:

- With Druid as the basic engine for processing/analytics, the GUI was designed to support users in different professional domains and big-data analysts in data-related tasks such as data preparation, analytics, and visualization, as well as the sharing of results.
- IT administrators can manage/monitor data sources in Druid, and they can establish data preparation rules if data sources of higher quality are required.

Druid functions reinforced in Metatron

The open-source Druid, despite its strengths in data collection and processing, had to be improved for Metatron to properly function as an end-to-end solution. This section examines the limitations of the open-source Druid and the functions reinforced in Metatron.

Limitations of the open-source Druid

The open-source Druid has the following limitations:

- Since Druid does not yet have full support for joins, Metatron uses another SQL engine for data preparation.
- Druid supports only a subset of SQL queries.
- For a data lake, a traditional SQL engine is more appropriate.
- Druid cannot append to or update already indexed segments, except for in some unusual cases.
- Nulls are not allowed.
- Filtering is not supported for metric columns.
- Linear scalability is not ensured. Increasing the number of servers doesn't improve the performance as much.
- Only a few data types are supported and it is difficult to add a new one.

- The management and monitoring tools are not powerful enough.

Druid functions reinforced in Metatron

The following functions of Druid were strengthened in Metatron:

Query functionality improvements

- Improved the functionality of the GroupBy query type.
- Slightly improved the functionality of other types of queries.

Features added

- Virtual columns (map, expression. etc.)
- New metric types (double, string, array, etc.)
- New expression functions
- Druid query results can be stored on the HDFS or exported into a file.
- Queries for meta information and statistics
- New aggregate functions (variance, correlation, etc.)
- (Limited) Window functions (lead, lag, running aggregations, etc.)
- (Limited) Joins
- (Limited) Sub-queries
- Temporary data sources
- Complex queries (data source summarization, correlation between data sources, k-means, etc.)
- Custom columns grouping
- Geographic information system (GIS) supported
- Columnar histograms
- Bit-slice indexing

Index structure improvements

- Histograms for filtering on metrics
- Lucene format supported for text filtering

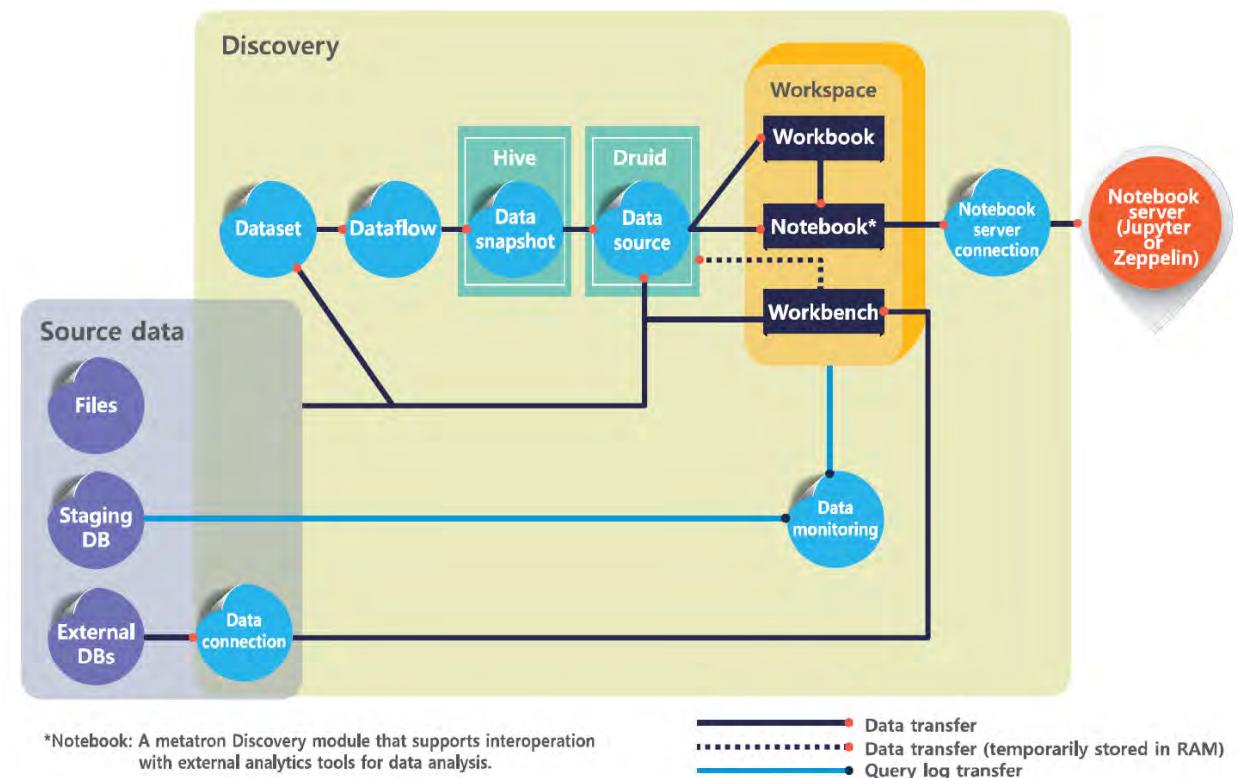
Connectability with other systems

- Hive storage handler
- Ingestion into Hive tables (based on connection with the Hive metastore)
- Ingestion into the ORC format
- RDBMS data ingestion via based on JDBC
- (Limited) SQL support backported

Miscellaneous improvements

- Bug fixes (+50) and minor improvements

DATA MANAGEMENT



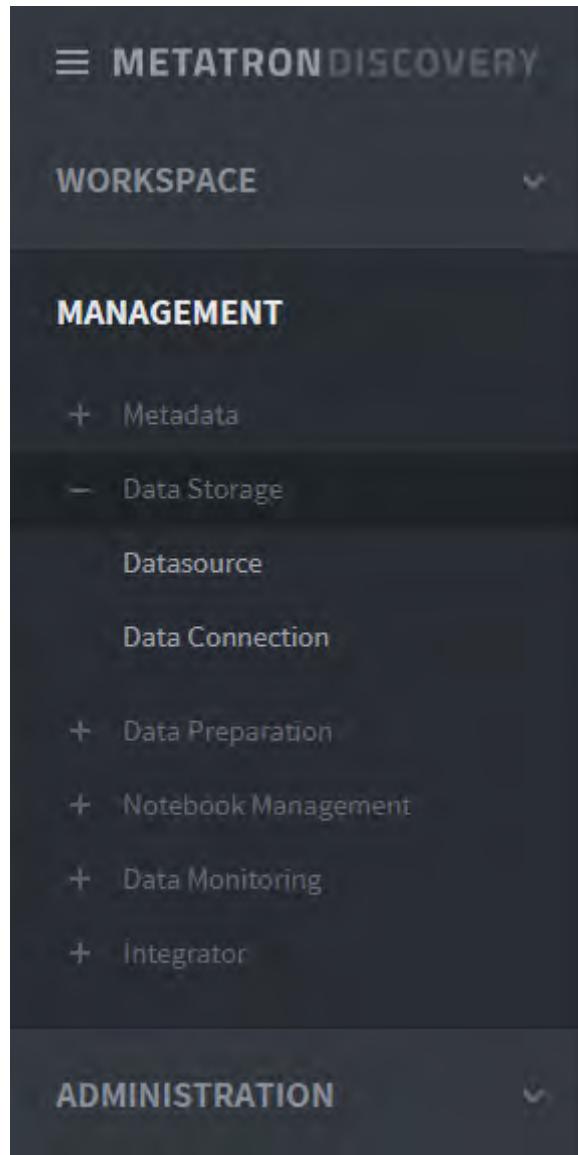
As shown above, data used by the three Discovery modules (workbook, notebook and workbench) is prepared from various types of source data, engines, and storages. For these operations, data flows need to be standardized and managed, and different types of source data need to be linked.

Source data required for analysis and visualization is either ingested into the Metatron engine as a **data source**, or linked directly from an external database with a **data connection**. Data usage can be monitored and tracked using **data monitoring**.

3.1 Data Source

In Metatron Discovery, a “data source” refers to a Druid database table into which data is ingested. Based on these data sources, workbooks and notebooks perform data analytics and visualization.

The Data Source menu can be accessed under **MANAGEMENT** > **Data Storage** > **Data Source** on the left-hand panel of the main screen.



3.1.1 “Dimensions” and “Measures”

The columns of a data source linked to the dashboard are categorized into **dimension** and **measure** columns as explained below. To make full use of Discovery’s data analysis and visualization features, you must understand the concepts of dimensions and measures clearly.

The screenshot shows the 'Datasource' interface with the title 'Sales Report'. It displays two main sections: 'Dimension' and 'Measure'. The 'Dimension' section contains several items, some with checkmarks: 'GeoPoint', 'OrderDate' (checked), 'Category', 'City', 'Country', 'CustomerName', 'OrderID', 'PostalCode', and 'ProductName'. The 'Measure' section contains several items, all with checkmarks: '## Discount', '## Profit' (checked), '## Quantity', '## Sales' (checked), '## DaystoShipActual', and '## SalesForecast' (checked). Navigation buttons like '< Previous' and 'Next >' are visible at the bottom of each section.

Dimension column

A column containing categorical data with the following characteristics:

- The values in this type of column are not for aggregation but to be categorized (e.g.: Category, Region, Organization)
- By each of these categories, measure values are aggregated.

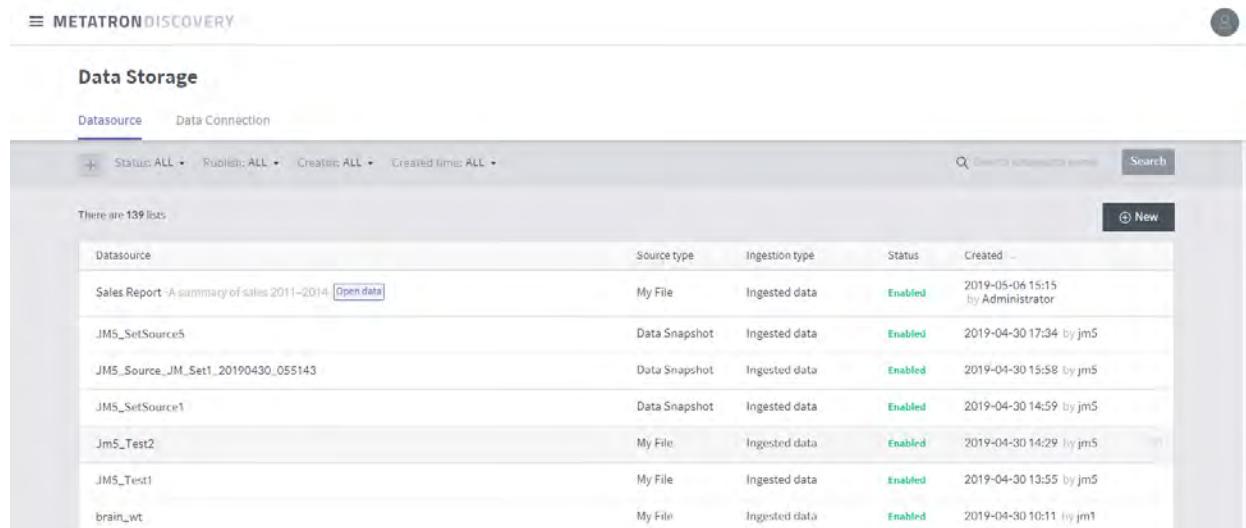
Measure columns

A column containing quantitative data with the following characteristics:

- The values in this type of column are subject to aggregation or contain quantitative information (e.g.: Sales)
- These values are aggregated based on dimensions.

3.1.2 Data source management home

On this home page, you can create, edit and view data sources.



The screenshot shows the 'Data Storage' section of the Metatron Discovery interface. At the top, there are tabs for 'Datasource' and 'Data Connection'. Below the tabs are filters for 'Status: ALL', 'Rulebook: ALL', 'Creator: ALL', and 'Created time: ALL'. A search bar and a 'Search' button are also present. The main area displays a table with 139 items. The columns are 'Datasource', 'Source type', 'Ingestion type', 'Status', and 'Created'. The table lists various data sources, including 'Sales Report - A summary of sales 2011–2014 [Open data]', 'JMS_SetSource5', 'JMS_Source_JM_Set1_20190430_055143', 'JMS_SetSource1', 'Jm5_Test2', 'Jm5_Test1', and 'brain_wt'. Each row shows the source name, its type (e.g., My File, Data Snapshot), ingestion type (e.g., Ingested data), status (e.g., Enabled), and creation date.

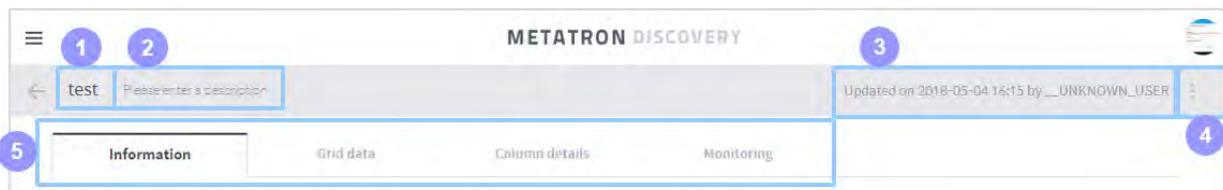
Datasource	Source type	Ingestion type	Status	Created
Sales Report - A summary of sales 2011–2014 [Open data]	My File	Ingested data	Enabled	2019-05-06 15:15 by Administrator
JMS_SetSource5	Data Snapshot	Ingested data	Enabled	2019-04-30 17:34 by jm5
JMS_Source_JM_Set1_20190430_055143	Data Snapshot	Ingested data	Enabled	2019-04-30 15:58 by jm5
JMS_SetSource1	Data Snapshot	Ingested data	Enabled	2019-04-30 14:59 by jm5
Jm5_Test2	My File	Ingested data	Enabled	2019-04-30 14:29 by jm5
Jm5_Test1	My File	Ingested data	Enabled	2019-04-30 13:55 by jm5
brain_wt	My File	Ingested data	Enabled	2019-04-30 10:11 by jm1

1. **Status:** Filters the data source list by the availability of data sources stored in the data storage.
 - **Enable:** Displays data sources that have been ingested and are available in workbooks or workbenches.

- **Preparing:** Displays new data sources whose ingestion is in progress.
 - **Failed:** Displays data sources that have not been created properly.
 - **Disabled:** Displays data sources that have been ingested but are not available because of an error in a certain Druid process.
2. **Publish:** Filter the data source list by public workspace.
- **Open Data:** Displays only data sources publicly available in all workspaces.
 - **Admin Workspace:** Displays only data sources available in the administrator workspace.
 - **Shared workspaces:** Displays only data sources available in the selected shared workspaces.
3. **Creator:** Filters the data source list by user or group that created the source data.
4. **Created time:** Determines whether the data source list is filtered by created or updated time. You can choose from among All, Today, and Last 7 days or specify a time range to display only those entries that were created/updated within the range.
5. **Search by name of data source:** Searches the data source list for the name you type in.
6. **Data source list:** Lists data sources filtered by specified criteria. Click an entry in the list to view its details. (Refer to [Data source details](#))
7. **Delete:** Hover the mouse over a data source to display a trash icon. Click the icon to delete the data source.

3.1.3 Data source details

Click a data source listed in the data source management home to view various attributes of that data source. The following subsections describe each area of the data source details. Note that a data source represents a Druid database table stored in Metatron and necessarily includes a timestamp column as a time-series table.

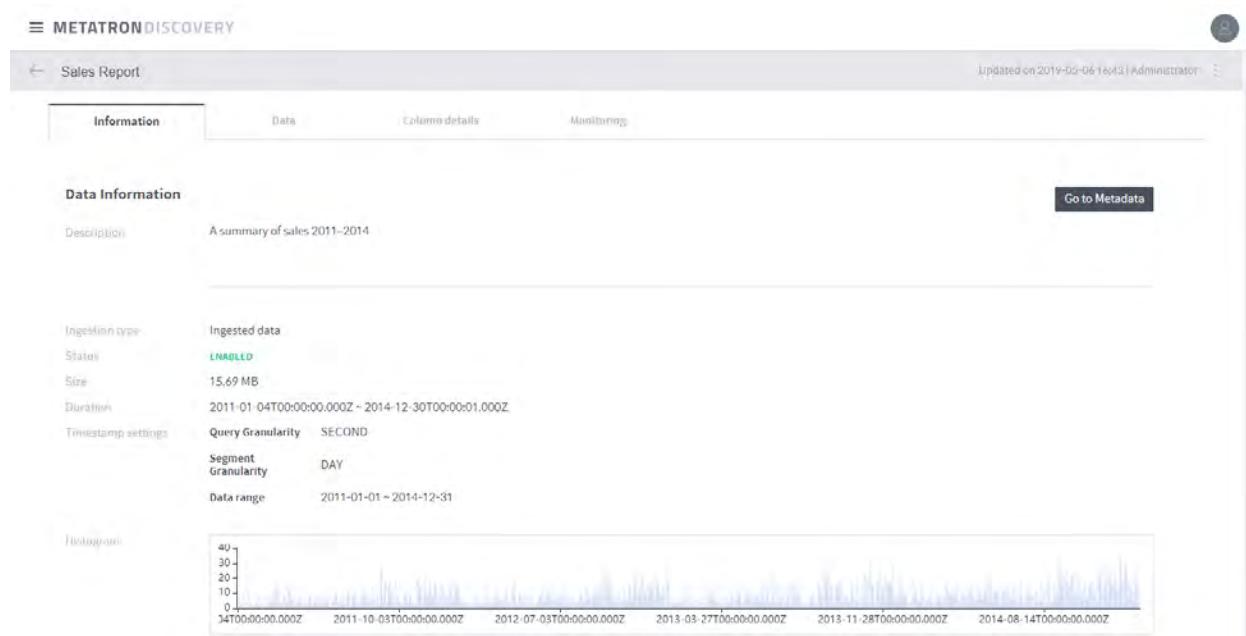


Common top area

1. **Name:** Name of the data source. Click on it if you want modify it.
2. **Description:** Description of the data source. Click on it if you want modify it.
3. **Last update:** Shows who and when last updated the data source.
4. **Delete:** Click this icon to display a menu that allows you to delete the data source.
5. **Tab selection:** Each tab displays a specific set of attributes of the data source. Depending upon the type of data source, not all of the three tabs may be displayed. For details on each tab, refer to the relevant subsection below.

Data information area

This area displays basic information of the data source.



1. **Data type:** Type of the imported source data from which the data source has been created.
2. **Status:** Displays the availability of the data source.
3. **Size:** Displays the size of the data source.

4. **Duration:** Displays the time range of the timestamps included in the data source.
5. **Timestamp setting:** Displays the granularities defined when the data source was created.
 - **Query Granularity:** Defines the minimum time period by which data is queried. This ensures faster returns by aggregating data per granularity interval.
 - **Segment Granularity:** In Druid, a data source is stored into multiple segments to be processed over multiple nodes in the distributed cluster environment. This granularity setting defines the time intervals into which the data source is partitioned.
 - **Histogram:** A graph displaying the size of the data stored within each time interval in Kbytes. This histogram is can be rendered because the Druid engine timestamps every table record.

Publish area

In this area, you can check and set which workspaces have access to the data source.

The screenshot shows a user interface for managing workspace access. On the left, there is a 'Publish' button. To its right, there is a checkbox labeled 'Allow all workspaces to use this datasource' with an 'Edit' link below it. Below these, a blue link indicates '1 workspace'. The entire section is enclosed in a light gray border.

1. **Allow all workspaces to use this data source:** Select this check box to make the data source available in all workspaces.
2. **Edit:** Used to allow specific workspaces to access the data source. This button will disappear if the data source is set as open data.
3. **Number of shared workspaces:** Displays how many workspaces have access to the data source.

Change data schema

The top section of the column details tab provides a user interface to filter columns by the criteria you define. Columns that meet the criteria are displayed on the left. You can also edit column settings.

Column view/settings

The screenshot shows the Metatrondiscovery interface with the following details:

- Header:** METATRON DISCOVERY, mysql_preset_engine_dialog_single_all, Updated on 2019-05-06 17:22 | Administrator
- Top Navigation:** Information, Data, **Column details**, Monitoring
- Search Bar:** Search data, Role (All selected), Dimension, Measure, Type (All selected), View all, Configure schema
- Table (Column list):**

Column name	Logical column name	Edit filters >
event_time	event_time	(edit)
activity_action	activity_action	
activity_actor	activity_actor	
activity_actor_type	activity_actor_type	
activity_generator_name	activity_generator_name	
activity_generator_type	activity_generator_type	
activity_object_id	activity_object_id	
activity_object_type	activity_object_type	
id	id	
- Column Information:**
 - Column name: event_time
 - Role: Dimension (selected)
 - Type: Timestamp
- Column Settings:**
 - Time display format: None
 - Missing: Do not apply
- Metadata:**
 - Logical Column Name: event_time
 - Dictionary: None
 - Code table: None
 - Description: None
- Statistic:**

Row count	215	Minimum	2018-06-01T00:00:00.000Z
		Maximum	2018-10-01T00:00:00.000Z
- Histogram:** A histogram visualization showing the distribution of the 'event_time' column.

- 1. Search data:** Searches for columns by the column name you type in.
- 2. Role:** Displays all, dimension, or measure columns.
- 3. Type:** Displays the columns whose data type is selected.
- 4. View all:** Clears all filter settings in the Search data, Role, and Type options and returns to view all columns.
- 5. Configure schema:** Click this button to prompt a window to edit the current column settings.
- 6. Column list:** Lists table columns.
- 7. Column information:** Displays attributes of the selected column.
- 8. Column settings:** Displays the metadata of the selected column.

9. **Statistics:** Displays the row count and other statistical values of the selected column.

Configure the schema

Provides a user interface for editing the name and type of columns.

Configure the schema

Cancel Save

① Metadata is also updated when modified.

Role	Name	Logical name	Type	Description
Dimension	GeoPoint	GeoPoint	Point	
Dimension	OrderDate	OrderDate	Timestamp	
Dimension	Category	Category	String	
Dimension	City	City	String	
Dimension	Country	Country	String	
Dimension	CustomerName	CustomerName	String	
Measure	Discount	Discount	Decimal	
Dimension	OrderID	OrderID	String	
Dimension	PostalCode	PostalCode	String	
Dimension	ProductName	ProductName	String	
Measure	Profit	Profit	Integer	
Measure	Quantity	Quantity	Integer	
Dimension	Region	Region	String	
Measure	Sales	Sales	Integer	
Dimension	Segment	Segment	String	
Dimension	ShipDate	ShipDate	Date/Time	①
Dimension	ShipMode	ShipMode	String	

1. **Role:** Displays whether the column is a dimension or measure.

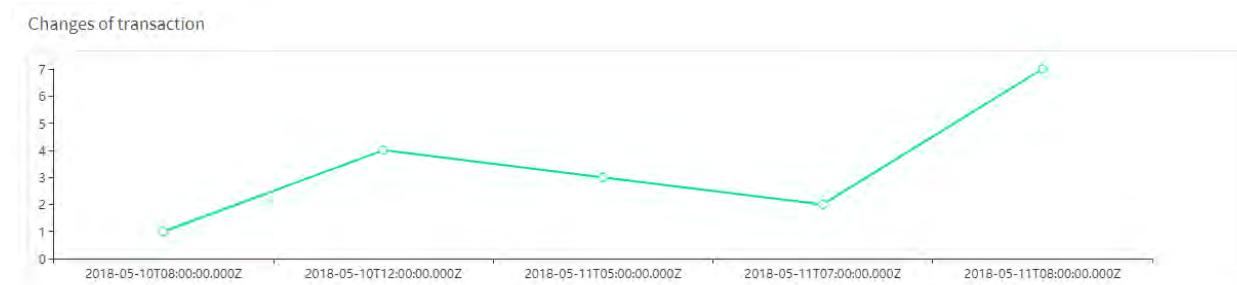
2. **Name:** Displays the actual name of the column.
3. **Logical name:** Allows you to edit the logical name of the column displayed in the system.
4. **Type:** Allows you to edit the logical type (character/integer/date, etc.) of the column.
5. **Format:** Allows you to edit the display format of the column in the case of the column being a timestamp type.
6. **Description:** Allows you to add a detailed description of the column.

Analyze data statistics

The Monitoring tab reports the usage of the data source.

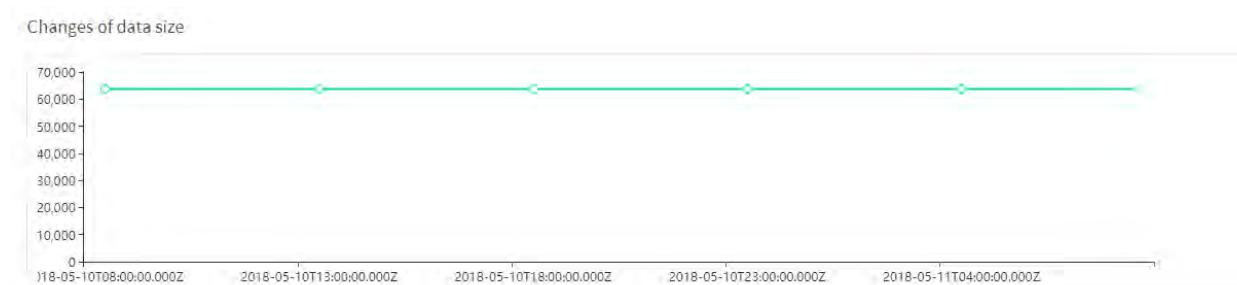
Change of transaction

Displays the trend of data source transactions over time.



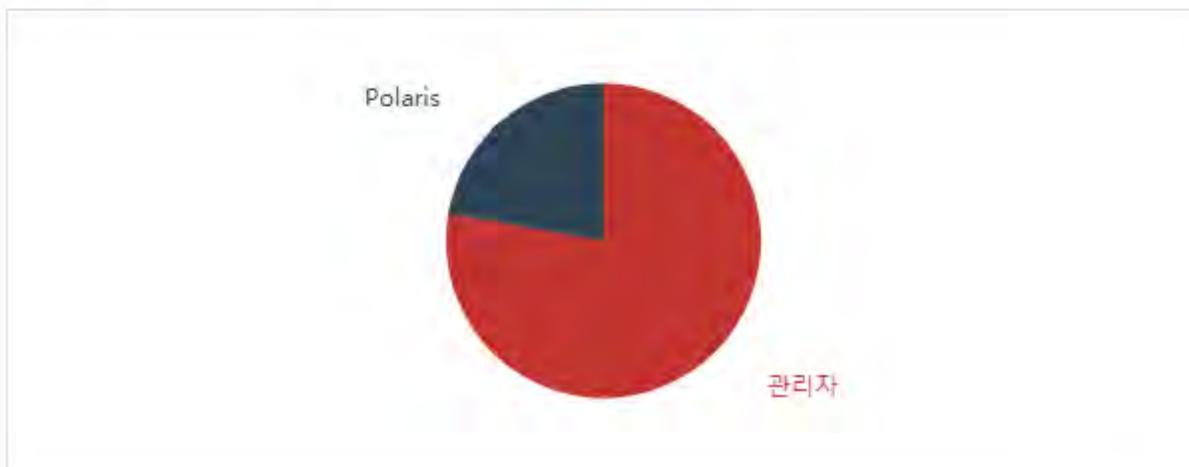
Changes of data size

Displays the trend of the data source size over time.



Query distribution (during last one week)

Query distribution by user (during last one week)



Query distribution by elapsed time (during last one week)



- **Query distribution by user (during last one week):** Displays a pie chart of query percentages by user for the past week.
- **Query distribution by elapsed time (during last one week):** Displays a pie chart of query percentages by execution time for the past week.

Query log

Used to view a detailed history of each performed query.

The screenshot shows the Metatron Query log interface. At the top, there are several filter options: 'Query date' (set to 'All'), 'Today', 'Last 7 days', and two dropdowns for 'yyyy-MM-dd hh:mm' and 'Query type' (both set to 'All'). Below these are buttons for 'Apply' and 'Result' (set to 'All'). The main area is a table with the following columns: No., Query date, Query type, User, Elapsed time, and Result. The table contains five rows of data. To the right of the table, there is a vertical column of five blue-bordered boxes, each labeled 'Detail' with a right-pointing arrow.

No.	Query date	Query type	User	Elapsed time	Result
1	2018-05-10 21:17	SUMMARY		85ms	Success
2	2018-05-11 16:41	SUMMARY		78ms	Success
3	2018-05-10 21:17	SEARCH		78ms	Success
4	2018-05-10 21:17	SUMMARY		76ms	Success
5	2018-05-11 17:30	SUMMARY		64ms	Success

1. **Date:** Set a time range to display only those queries that were last executed within this time range.
2. **Query type:** Filters the performed queries by type.
3. **Status:** Displays all, succeeded, or failed queries.
4. **Query list:** Lists queries filtered by specified criteria.
5. **Detail:** Click on it to view the query statement.

3.1.4 Create a data source

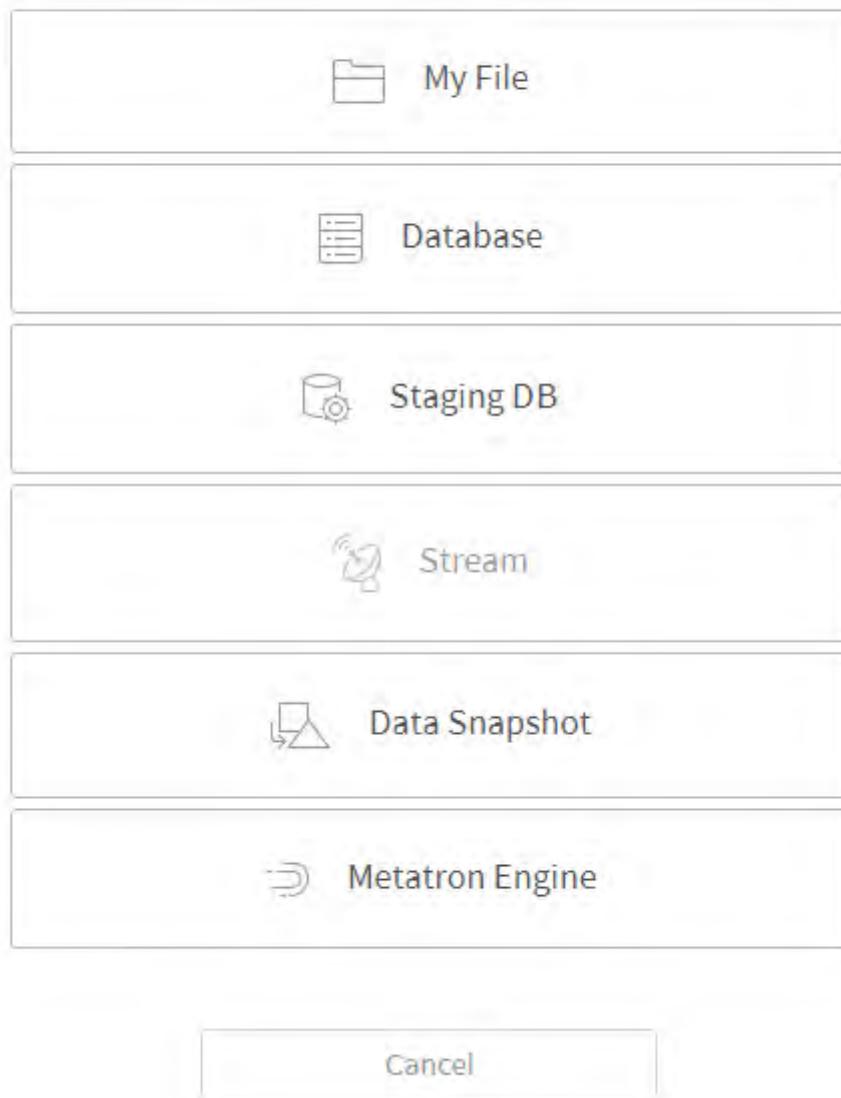
This section explains the process of ingesting various types of source data into the Metatron engine and converting them into data sources.

To create a data source, click the **+ New** button at the top right of the **Data Source** home screen.

The screenshot shows the 'Data Storage' screen in Metatron Discovery. At the top, there are tabs for 'Datasource' (which is selected) and 'Data Connection'. Below the tabs are several filter dropdowns: 'Status: ALL', 'Priority: ALL', 'Credential: ALL', and 'Created time: ALL'. There is also a search bar and a 'New' button. The main area displays a table with the message 'There are 139 lists'.

Then, select the type of source data.

Select source type



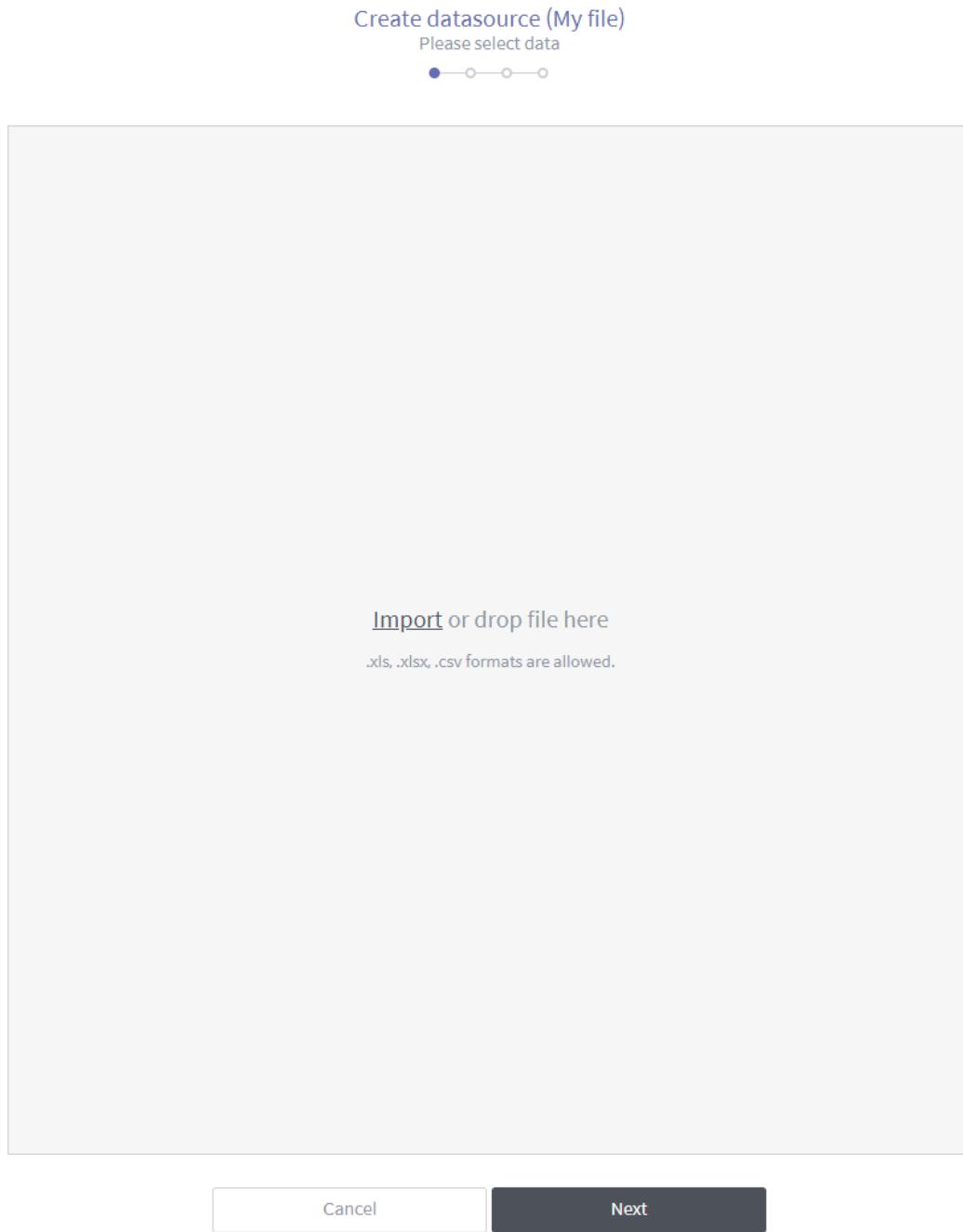
- **File:** Creates a data source from a file stored on your local PC (for details, refer to [Create a data source from a file](#)).
- **Database:** Creates a data source from an external database (for details, refer to [Create a data source from a database](#)).
- **Staging DB:** Creates a data source from Metatron's internal Hive database (for details, refer to [Create a data source from a staging database](#)).

- **Stream:** This function is not currently supported.
- **Data Snapshot:** This function is not currently supported.
- **Metatron Engine:** Migrates a data source stored in a previous Metatron version (for details, refer to [Add a data source with the Metatron engine](#)).

Create a data source from a file

Creates a data source from a file stored on your local PC.

1. On the source data type selection page, select **File**.
2. Select a file to be used as a data source from your local PC. You can either click the **Import** button and select the file, or drag and drop a file to the box. Once a file is selected, click Next.



3. From the file, select the sheet to be included in the data source.

Note: If the “No preview data” message is shown in spite of there being data, check whether the **Column delimiter** and **Line Separator** have been configured correctly. In this example, the **Line Sep-**

arator must be set to “r”? the carriage return for MS Windows.

Create datasource (My file)
Please select data



 sales-data-sample.csv Import or drop file here

	3369920 byte	28 Columns	100	/ 9876 Row	1 Types		
ab OrderDate	ab Category	ab City	ab Country	ab CustomerName	ab Discount	ab OrderID	ab Pos
2011-01-04T00:00:00	Office Supplies	Houston	United States	Darren Powers	0.2	CA-2011-103...	770
2011-01-05T00:00:00	Office Supplies	Naperville	United States	Phillina Ober	0.2	CA-2011-112...	605
2011-01-05T00:00:00	Office Supplies	Naperville	United States	Phillina Ober	0.8	CA-2011-112...	605
2011-01-05T00:00:00	Office Supplies	Naperville	United States	Phillina Ober	0.2	CA-2011-112...	605
2011-01-06T00:00:00	Office Supplies	Philadelp...	United States	Mick Brown	0.2	CA-2011-141...	191
2011-01-07T00:00:00	Furniture	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:00:00	Office Supplies	Athens	United States	Jack OBriant	0.0	CA-2011-106...	306
2011-01-07T00:00:00	Office Supplies	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:00:00	Office Supplies	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:00:00	Office Supplies	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:00:00	Office Supplies	Los Angeles	United States	Lycoris Saunders	0.0	CA-2011-130...	900
2011-01-07T00:00:00	Technology	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-07T00:00:00	Technology	Henderson	United States	Maria Etezadi	0.0	CA-2011-167...	424
2011-01-08T00:00:00	Furniture	Huntsville	United States	Vivek Sundaresam	0.6	CA-2011-105...	773
2011-01-08T00:00:00	Office Supplies	Huntsville	United States	Vivek Sundaresam	0.8	CA-2011-105...	773

Column delimiter: ,

Line separator: \n

Use the first row as the head column. (If not checked, a new row is created and is used as the head column)

- **File name:** Name of the imported file. You can replace it with another file.
- **File sheet list:** Displays the sheets included in the imported file. Select the sheet from which you want to create a data source.

- **File sheet name:** Name of the currently selected sheet.
- **Size:** Size of the imported file.
- **Column:** Number of columns in the imported file.
- **Row:** Displayed number of rows and total number of rows in the imported file. Enter the number of rows to be displayed on the page.
- **Type:** Displays how many data types are recognized from the columns. The data type of each column can be modified later.
- **Use the first row as the head column:** Select the check box to use the first row of the file as column headers. If you don't select it, a new row is inserted as a column header row.

4. Configure the schema of the data source.

Create datasource (My file)
Configure schema

Role: All Type: All Add column

Search by column name:

Column			
Dimension	ab OrderDate		
Dimension	ab Category		
Dimension	ab City		
Dimension	ab Country		
Dimension	ab CustomerName		
Dimension	ab Discount		
Dimension	ab OrderID		
<input checked="" type="checkbox"/> Dimension	ab PostalCode		
<input checked="" type="checkbox"/> Dimension	ab ProductName		
Dimension	ab Profit		
Dimension	ab Quantity		
Dimension	ab Region		
Dimension	ab Sales		
Dimension	ab Segment		
Dimension	ab ShipDate		
Dimension	ab ShipMode		
<input checked="" type="checkbox"/> Dimension	ab State		
Dimension	ab Sub_Category		
Dimension	ab DaystoShipActual		
Dimension	ab SalesForecast		
Dimension	ab ShipStatus		
Dimension	ab DaystoShipScheduled		
Dimension	ab OrderDetailStatus		

3 Selections Change type Delete

State

Data

Texas
Illinois
Illinois
Illinois
Pennsylvania
Kentucky
Georgia
Kentucky
Kentucky
Kentucky
Kentucky
Kentucky
Kentucky
Kentucky
Texas
Texas
Texas
Virginia
Virginia
Delaware
South Carolina
California

Role

Dimension
 Measure

Type

ab String

Missing

Do not apply
 Discard
 Replace with

Previous Next

(1) One of the time-type columns or current time must be specified as a Timestamp

(2) Current time Time-type column No selected time-type column

- **Search by column header:** Searches the imported file for columns by name.
- **버튼 (우측 상단):** 선택한 컬럼을 삭제합니다.
- **Role:** Displays all, dimension, or measure columns from the imported file.
- **Recommended filters:** Displays columns to which a top-priority filter is applied.
- **Type:** Filters the columns in the imported file by field type.
- **Column list section:** Lists columns filtered by specified criteria. Once you have selected columns, a panel appears at the bottom of the screen. After selecting your desired batch action in the panel, click **Apply** to perform the batch action on the selected columns.
- **Individual column settings section:** This area is used to set the attributes of a column selected from the column list. **Missing** is used to set nulls in the column.
 - **Replace with:** Replaces the nulls with the value typed in.
 - **Discard:** Discards the nulls.
 - **Do not set:** Leaves the nulls as nulls. However, the nulls in the timestamp column are mandatorily discarded.
- **Timestamp setting:** Determines how to timestamp each row. You can either designate an existing time-type column as a timestamp column, or create a new time-type column whose values are all timestamped with the current time.

Note: Metatron Druid is a time-series engine that requires a timestamp for each row when a data source is created.

- **컬럼 추가:** 데이터에 위도, 경도 컬럼이 있는 경우 이를 결합하여 Point 타입의 신규 컬럼을 추가할 수 있습니다. 이 컬럼을 지우면 다른 컬럼들과 동일하게 동작합니다.

5. Configure data source ingestion and click Next.

Create datasource (My file)
Please complete ingestion settings

Timestamp settings

Query Granularity Second

Segment Granularity Hour

Data range

2010-12-31 05 ~ 2011-01-25 13 609 segment granularity units

① The interval should set equal to or greater than the range of data values in the timestamp column, and the number of segments units cannot exceed 10,000.

Rollup true false

Advanced setting ▾

Previous Next

- **Segment Granularity:** In Druid, a data source is stored into multiple segments to be processed over multiple nodes in the distributed cluster environment. This granularity setting defines the time intervals into which the data source is partitioned.
- **Query Granularity:** Defines the minimum time period by which data is queried. This ensures faster returns by aggregating data per granularity interval.
- **Rollup:** “Data rollup” summarizes data based on its dimension (for details on the concept of data rollup, refer to [Data roll-up](#)). A summarization rule might be summing up all values in each column or applying a set of expressions such as profit=sales-expenses.
- **Advanced settings:** Configures how to ingest data. Type in the text box in the JSON format. For example,

```
{maxRowsInMemory : 75000,
maxOccupationInMemory : -1,
maxShardLength : -2147483648,
```

(continues on next page)

(continued from previous page)

```
leaveIntermediate : false,  
cleanupOnFailure : true,  
overwriteFiles : false,  
ignoreInvalidRows : false,  
assumeTimeSorted : false}
```

6. Confirm the information about the data set from the imported file, enter the **Name** and **Description**, and click **Done** to create a data source. It may take a few seconds or minutes depending on the amount of data as the source data is ingested into the internal Metatron engine (Druid).

The screenshot shows the Metatron user interface for managing data sources. At the top, there's a header with a back arrow and the title "Sales Report". Below the header, there are four tabs: "Information" (which is selected), "Data", "Column details", and "Monitoring". The "Information" tab displays the following details:

Data Information							
Description	A summary of sales 2011–2014						
Ingestion type	Ingested data						
Status	ENABLED						
							
Timestamp settings	<table border="1"><thead><tr><th>Query Granularity</th><th>SECOND</th></tr></thead><tbody><tr><td>Segment Granularity</td><td>DAY</td></tr><tr><td>Data range</td><td>2011-01-01 ~ 2014-12-31</td></tr></tbody></table>	Query Granularity	SECOND	Segment Granularity	DAY	Data range	2011-01-01 ~ 2014-12-31
Query Granularity	SECOND						
Segment Granularity	DAY						
Data range	2011-01-01 ~ 2014-12-31						

7. After data ingestion is complete, you can check the status. In the example below, the status is set to **ENABLED** and a histogram is displayed.

8. In the **Data** tab, you can check the ingested data in the form of a table.

The screenshot shows the METATRON DISCOVERY Data Source interface. At the top, there's a navigation bar with 'METATRON DISCOVERY' and a user icon. Below it, a breadcrumb navigation shows '← Sales Report'. On the right, it says 'Updated on 2019-05-06 16:22 | Administrator'. The main area has tabs for 'Information', 'Data' (which is selected), 'Column details', and 'Monitoring'. Under 'Data', there are filters for 'Role' (All), 'Dimension' (All), 'Measure' (All), and 'Type' (All). A search bar at the top of the data grid allows searching by 'Search data'. The data grid itself has columns for GeoPoint, OrderDate (UTC+9), Category, City, Country, CustomerName, Discount, OrderID, PostalCode, ProductName, Profit, Quantity, and Region. The data shows various sales entries from 2011, such as '29.8941-9...', '41.7662-8...', etc., with details like 'Office Supply' purchases in Houston, Naperville, and Philadelphia.

- On the Data Source management home screen, you will find a newly-created data source. While data is being ingested, the status is displayed as **Disabled** as shown below; the status changes to **Enabled** once ingestion is complete. After that, you can use the data source.

Data Storage

The screenshot shows the Data Source management interface. At the top, there are tabs for 'Datasource' (which is selected) and 'Data Connection'. Below the tabs, there are filters for 'Status: ALL', 'Publish: ALL', 'Creator: ALL', and 'Created time: ALL'. There's also a search bar with 'Search' and a 'New' button. The main area displays a table with 139 lists. The columns are 'Datasource', 'Source type', 'Ingestion type', 'Status', and 'Created'. One entry, 'Sales Report - A summary of sales 2011-2014 [Open data]', is listed with 'My File' as the source type, 'Ingested data' as the ingestion type, and 'Enabled' as the status. It was created on '2019-05-06 15:15' by 'Administrator'.

Create a data source from a database

Creates a data source from an external database.

1. On the source data type selection page, select **Database**.
2. Enter the information to connect the database.

Create datasource (DB)
Please set data connection

Ingestion type Ingested data Linked data

DB connection

MySQL PostgreSQL Hive Presto Druid MESQL

Host: metatron-hadoop-04 Port: 10000
 URL only

User name: hive Password:

Security:
 Always connect
 Connect by user's account
 Connect with hdfs path or password Can not ingest by batch method.

- **Ingestion type:** Select how to ingest data into the data source.
 - **Ingested data:** Displays data sources that contain data ingested into the Metatron storage.
 - **Linked data:** Displays data sources that load data from linked databases whenever necessary.
 - **Load a data connection:** Automatically loads access information for a database that is already registered as a data connection. However, you must verify the connection by clicking the **Validation check** button.
 - **DB type:** Select the type of the database to be connected.
 - **Host:** Enter the hostname to connect to the database.
 - **Port:** Enter the port to connect to the database.
 - **User name:** Enter the username of the database.
 - **Password:** Enter the password of the database.
 - **Validation check:** Once you fill out all fields, the Test button becomes active. Click on it to verify if the connection is valid: The validity of the connection appears below the button.
3. Select data. You can either select a table from the connected database, or write a query yourself.

Create datasource (DB)
Please select data

Table Query

ab_id	ab_created_by	ab_created_time	ab_modified_by	ab_modified_time	# version	ab_dc_connect_url	ab_dc_
01007...	admin	2018-09-26 14:3...	admin	2018-09-26 14:34...	3	jdbc:hive2://metat...	met
01b73...	admin	2018-10-23 02:1...	anonymousUser	2018-10-23 04:11...	15	jdbc:hive2://metat...	met
01ced...	polaris	2018-10-18 06:4...	polaris	2018-10-18 06:48...	3	jdbc:hive2://metat...	met
023ee...	admin	2018-09-07 12:4...	admin	2018-09-08 12:05...	3	jdbc:hive2://metat...	met
0259c...	admin	2018-10-17 08:1...	admin	2018-10-17 08:13...	3	jdbc:hive2://metat...	met
03464...	admin	2018-10-17 08:5...	admin	2018-10-17 08:51...	3	jdbc:hive2://metat...	met
04b7f...	admin	2018-08-10 02:1...	admin	2018-08-10 02:15...	3	jdbc:hive2://metat...	met
05237...	admin	2018-09-07 12:4...	admin	2018-09-08 12:05...	3	jdbc:hive2://metat...	met
05692...	admin	2018-09-07 12:4...	admin	2018-09-08 12:05...	3	jdbc:hive2://metat...	met
06af8...	admin	2018-10-22 07:3...	admin	2018-10-22 07:35...	3	jdbc:hive2://metat...	met
0727b...	admin	2018-09-07 12:4...	admin	2018-09-08 12:05...	3	jdbc:hive2://metat...	met
0851d...	admin	2018-10-29 00:4...	admin	2018-10-29 00:48...	3	jdbc:hive2://metat...	met
0902d...	polaris	2018-10-17 07:3...	polaris	2018-10-17 07:32...	3	jdbc:hive2://metat...	met
096cf...	admin	2018-10-17 08:3...	admin	2018-10-17 08:37...	3	jdbc:hive2://metat...	met
09e00...	admin	2018-09-07 12:4...	admin	2018-09-08 12:05...	3	jdbc:hive2://metat...	met
0a52c...	admin	2018-10-15 01:0...	admin	2018-10-15 01:04...	3	jdbc:hive2://metat...	met
0ae83...	admin	2018-10-17 08:1...	admin	2018-10-17 08:12...	3	jdbc:hive2://metat...	met
0b263...	admin	2018-09-24 18:2...	admin	2018-09-24 18:21...	3	jdbc:hive2://metat...	met
0b69f...	admin	2018-10-23 08:2...	anonymousUser	2018-10-23 08:32...	19	jdbc:hive2://metat...	met
0b6f8...	admin	2018-09-07 12:4...	admin	2018-09-08 12:05...	3	jdbc:hive2://metat...	met
0ba77...	admin	2018-09-07 12:4...	admin	2018-09-08 12:05...	3	jdbc:hive2://metat...	met
0bccd...	admin	2018-10-29 00:4...	admin	2018-10-29 00:48...	3	jdbc:hive2://metat...	met

Previous Next

- Table:** Select a database and a table to display the table's data. Once the data being ingested has been displayed, confirm the data and click **Next**.
- Query:** Write a query to import the data you want, and click **Run** to display the data in the lower section. Confirm the data and click **Next**.

4. The rest of the process is identical to [Create a data source from a file](#). However, when creating a data source from a database, you must configure additional **ingestion settings** as follows.

Create datasource (DB)
Please complete ingestion settings

Ingestion settings

Ingest Once Ingest periodically

Scope of Ingesting data

All Limited record count 10000 rows

Timestamp settings

Query Granularity Second

Segment Granularity Hour

Data range

2018-08-05 22 ~ 2018-11-04 00 2,163 segment granularity units

ⓘ The interval should set equal to or greater than the range of data values in the timestamp column, and the number of segments units cannot exceed 10,000.

Rollup true false

[Advanced setting](#) ▾

Previous Next

- **Ingest once:** Ingest the data currently stored in the database only this once. When selecting the **Limited record count**, you can specify how many rows are to be ingested from the first row.

Ingestion settings

Ingest Once Ingest periodically

Scope of Ingesting data

All Limited record count 10000 rows

- **Ingest periodically:** Saves data on a regular basis.

Ingestion settings

Ingest Once Ingest periodically

Scope of Ingesting data

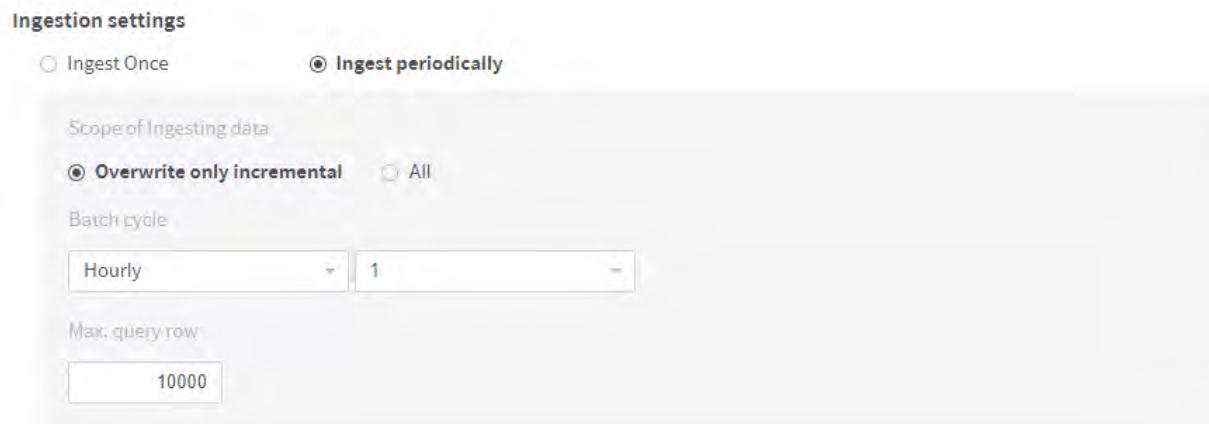
Overwrite only incremental All

Batch cycle

Hourly 1

Max. query row

10000



Create a data source from a staging database

Creates a data source from Metatron's internal Hive database.

1. On the source data type selection page, select **Staging DB**.
2. Once you select the database and its table to connect, the data is displayed.

Create datasource (Staging DB)

Please select data

tpch_10 lineitem

#	I_orderkey	#	I_partkey	#	I_suppkey	#	I_linenumber	## I_quantity	## I_extendedprice	## I_discount	#
1	1551894	76910	1		73092	2	17	33078.94	0.04	0.09	
1	673091	73092	2		636998	3	36	38306.16	0.1	0.09	
1	21315	46316	4		156345	5	8	15479.68	0.09	0.07	
1	240267	15274	5		1061698	6	28	34616.68	0.1	0.09	
1	156345	6348	6		42970	7	24	28974	0.07	0.06	
2	156345	6348	8		190355	9	32	44842.88	0.04	0.06	
2	1061698	11719	1		1284483	10	38	63066.32	0.1	0.06	
3	1061698	11719	1		293797	11	45	86083.65	0.07	0.04	
3	42970	17971	1		1830941	12	27	70822.15	0.03	0.06	
3	1830941	17971	2		621426	13	28	39620.34	0.09	0.1	
3	1830941	17971	3		880347	14	2	3581.56	0.07	0.01	
3	621426	96445	4		880347	15	5	52411.8	0.03	0.04	
3	880347	55372	6		1085693	16	26	35032.14	0.09	0.1	
4	880347	55372	7		1085693	17	1	39819	0.07	0.03	
5	1085693	85694	8		1239268	18	15	25179.6	0.02	0.02	
5	1239268	39269	9		375302	19	2	31387.2	0.09	0.07	
5	375302	306	10		1396355	20	3	68864.5	0.08	0.07	
6	1396355	21369	11		1820519	21	1	53697.73	0.08	0.08	
7	1820519	95574	12		1452428	22	12	17273.04	0.07	0.08	
7	1452428	77443	13		947798	23	9	12423.15	0.07	0.08	
7	947798	97817	14		1630721	24	46	84904.5	0.03	0.1	
7	1630721	30722	15				28	46245.92	0.03		

Cancel Next

- The rest of the process is identical to [Create a data source from a database](#).

Create datasource (Staging DB)

Configure schema

Search by column name

Role: All Dimension Measure Type: All Add column

Column	Setting
# I_orderkey	
# I_partkey	
# I_suppkey	
# I_linenumber	
# I_quantity	
# I_extendedprice	
# I_discount	
# I_tax	
ab I_returnflag	
ab I_linenstatus	
ab I_shipdate	
ab I_commitdate	
ab I_receiptdate	
ab I_shipinstruct	
ab I_shipmode	
ab I_comment	

I_orderkey

Data 50 Row

Setting

Role: Dimension Measure

Type: # Integer

Missing: Replace with

Discard: Do not apply

Previous Next

One of the time-type columns or current time must be specified as a Timestamp

Current time Time-type column: No selected time-type column

Add a data source with the Metatron engine

Migrates a data source stored in a previous Metatron version.

1. On the source data type selection page, select **Metatron Engine**.
2. When data sources created in a previous version of Metatron are listed on the left as shown below, select the check boxes of the data sources you want to migrate to the current version.

Create datasource (Metatron Engine)
Please select data table

<input type="checkbox"/> 1 Selections	mysql_preset_engine_dialog_single_all				
	event_time	ab_activity_action	ab_activity_actor	ab_activity_actor_type	ab_activity_generator
<input type="checkbox"/> monthday	2018-06-01 00:00:00	VIEW	admin	PERSON	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Safari/537.36
<input type="checkbox"/> monthmonth	2018-06-01 00:00:00	VIEW	admin	PERSON	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Safari/537.36
<input type="checkbox"/> monthyear	2018-06-01 00:00:00	VIEW	admin	PERSON	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Safari/537.36
<input type="checkbox"/> mv_current	2018-06-01 00:00:00	VIEW	admin	PERSON	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Safari/537.36
<input type="checkbox"/> mv_twmuq	2018-06-01 00:00:00	VIEW	admin	PERSON	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Safari/537.36
<input type="checkbox"/> mysql_1	2018-06-01 00:00:00	VIEW	admin	PERSON	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Safari/537.36
<input type="checkbox"/> mysql_10	2018-06-01 00:00:00	VIEW	admin	PERSON	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Safari/537.36
<input type="checkbox"/> mysql_8	2018-06-01 00:00:00	VIEW	admin	PERSON	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Safari/537.36
<input type="checkbox"/> mysql_9	2018-06-01 00:00:00	VIEW	admin	PERSON	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87 Safari/537.36
<input checked="" type="checkbox"/> mysql_preset_engine_dialog_single_all					
<input type="checkbox"/> mysql_preset_engine_dialog_single_row					
<input type="checkbox"/> mysql_preset_engine_manual_batch_all					
<input type="checkbox"/> mysql_preset_engine_manual_batch_inc					
<input type="checkbox"/> mysql_preset_engine_manual_single_all					

Cancel Done

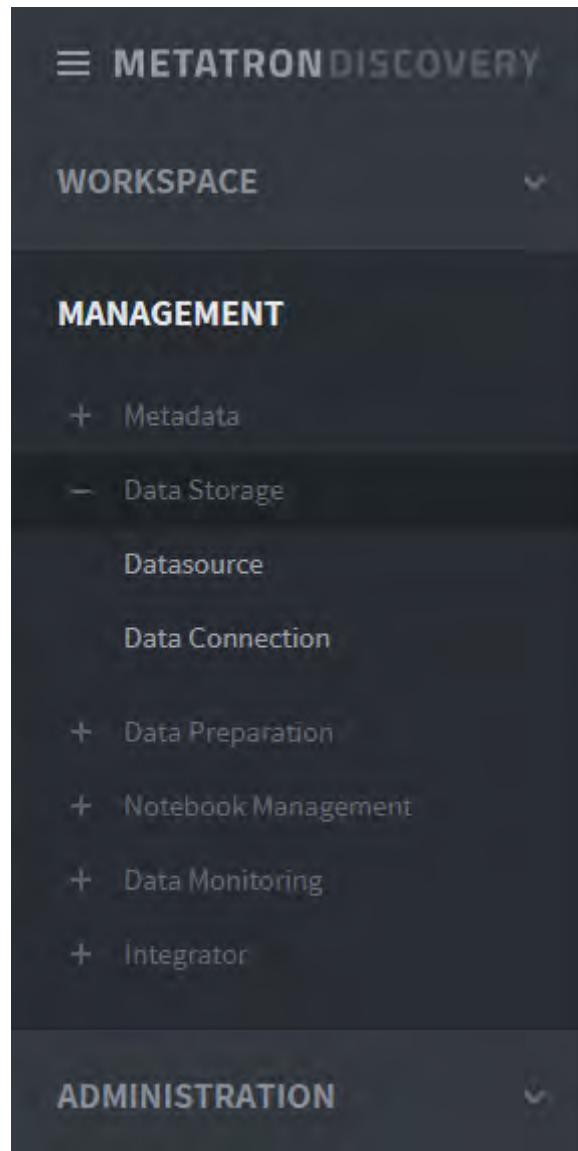
3. Click Done to migrate the selected data sources.

Datasource	Source type	Ingestion type	Status	Created
mysql_preset_engine_dialog_single_all	Metatron Engine	Ingested data	Enabled	2019-05-06 17:22 by Administrator

3.2 Data Connection

Metatron Discovery can connect to an external database directly. To connect to an external database, you must create and manage a data connection containing the access information to that database. By registering such a data connection, you don't need to enter the access information each time you connect to the same database.

The Data Connection menu can be accessed under MANAGEMENT > Data Storage > Data Connection on the left-hand panel of the main screen.



3.2.1 Data connection management home

On the **Data Connection** home page, you can create, edit and view database connections.

Data connection	DB Type	Host/Port(URL)	Created
Hive-metatron-hadoop-01-10000	Hive	metatron-hadoop-04 / 10000	2019-03-13 15:18 by Administrator
Presto-metatron-hadoop-01-8089	Presto	metatron-hadoop-01 / 8089	2019-03-02 16:10 by Administrator
druid connection	Druid	metatron-hadoop-02 / 8082	2019-02-25 13:43 by Administrator
MySQL-metatron-web-03-3306	MySQL	metatron-web-03 / 3306	2019-02-21 10:44 by Administrator

- Publish:** Filter the data connection list by public workspace.
- Creator:** Filter the data connection list by creator.
- DB type:** Filter the data connection list by database type (MySQL, PostgreSQL, Hive, or Presto).
- Security:** Filter the data connection list by security type (Always connect, connect by user's account, or connect with ID and password).
- Created time:** Filter the data connection list by time of creation (Today, Last 7 days, or Between).
- Search:** Search the data connection list by data connection name.
- Number of data connections:** Displays how many data connections are returned in the list.
- New:** Click on it to create a new data connection.
- Delete:** Hover the mouse over a data connection to display a recycle bin icon. Click the icon to delete the data connection.

3.2.2 Create a data connection

On the **Create data connection** screen, enter the required information to create a connection.

Create data connection
Please set required items and complete data connection creation

DB connection

MySQL PostgreSQL Hive Presto Druid MSSQL

Host	Port
127.0.0.1	3306
<input type="checkbox"/> URL only	
User name	Password
admin	*****
Security	
<input checked="" type="radio"/> Always connect <input type="radio"/> Connect by user's account <input type="radio"/> Connect with ID and password	
<input type="button" value="Validation check"/>	
Advanced settings	

Publish

1 workspaces [Edit](#)
 Allow all workspaces to use this dataconnection

Connection name

Enter name of new data connection

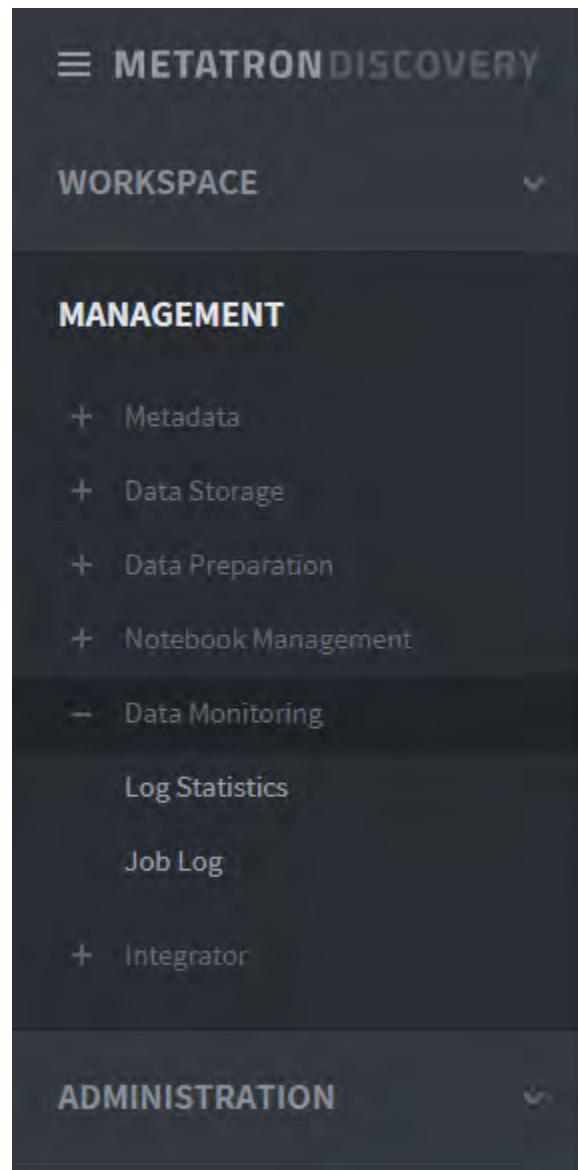
- **DB type:** Four database types are currently supported. (MySQL, PostgreSQL, Hive, Presto)

- **Host:** Enter the hostname to connect to the database.
- **Port:** Enter the port to connect to the database.
- **URL only:** Enter a database URL instead of a host and port.
- **User name:** Enter the username of the database.
- **Password:** Enter the password of the database.
- **Security:** Set the type of security to be applied while using the data connection.
 - **Always connect:** Logs in using the account information the user has entered to create a new data connection.
 - **Connect by user's account:** Logs in using the account information registered in Metatron Discovery.
 - **Connect with ID and password:** Requires to enter the account information every time the data connection is used.
- **Validation check:** Checks whether the connection information entered is valid; the result is shown next to the button. The validity of the connection appears below the button.
- **Advanced settings:** You can add a custom property key and value as options.
- **Publish:** Set which workspaces have access to the data connection.
 - **Allow all workspaces to use this data connection:** Select this check box to make the data connection available in all workspaces.
 - **Edit:** Used to allow specific workspaces to access the data connection. This button will disappear if the data connection is set as open data.
 - **Number of shared workspaces:** Displays how many workspaces have access to the data connection.

3.3 Data Monitoring

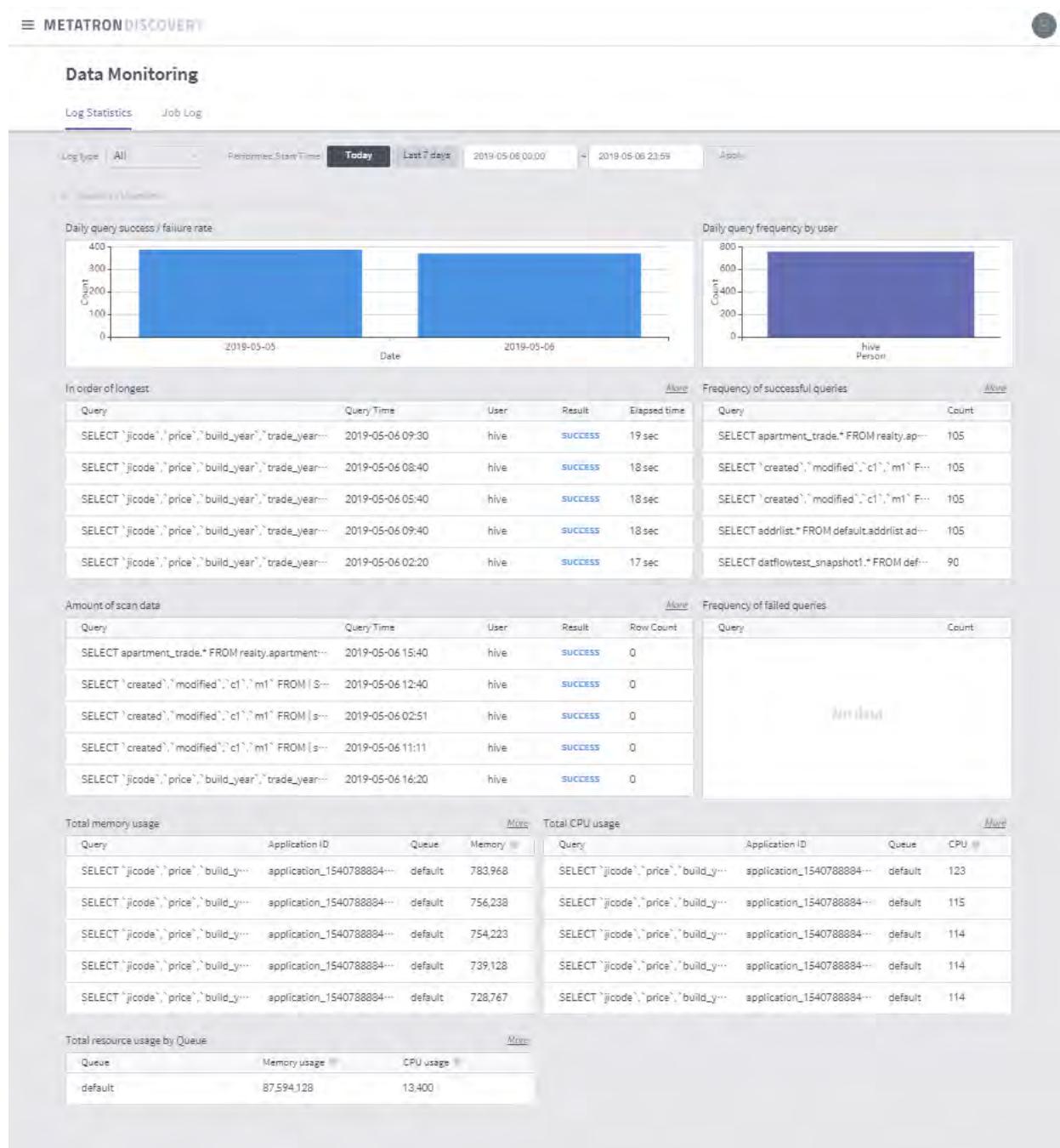
Data monitoring supports monitoring the logs of all queries submitted by users in Metatron Workbench to the staging database (internal Hive database) and external databases connected to Metatron.

The Data Monitoring menu can be accessed under **MANAGEMENT** > **Data Storage** > **Data Monitoring** on the left-hand panel of the main screen.



3.3.1 Log Statistics

This page collects and reports various statistics related to the performance of queries in Metatron Discovery. You can view the following nine types of basic statistics.



- Query success/failure rate:** Displays the daily success/failure rates of queries performed in Metatron.
- Query frequency by user:** Graph indicating how many queries were performed by each user. Click a bar to view the job log for the user.
- In order of longest:** Displays the performed queries in the order of the longest running time.

4. **Amount of scan data:** Displays the performed queries in the order of the highest amount of scanned data.
5. **Frequency of successful queries:** Displays the performed queries in the order of the highest frequency of success.
6. **Frequency of failed queries:** Displays the performed queries in the order of the highest frequency of failure.
7. **Total memory usage:** Displays the performed queries in the order of the largest memory usage in total.
8. **Total CPU usage:** Displays the performed queries in the order of the largest CPU usage in total.
9. **Resource usage by queue:** Displays the resource usage in each YARN queue in the Hadoop environment.

3.3.2 Job Log

This page reports the history of all queries performed in Metatron. You can easily view previous jobs by searching the history of queries with your customized filters. The following are the filters applicable to job searching.

The screenshot shows the Metatron Discovery Data Monitoring page. At the top, there are tabs for 'Log Statistics' and 'Job Log', with 'Job Log' being the active tab. Below the tabs are filters for 'Log type' (All), 'Status' (All, Success, Running, Cancelled, Fail), 'Elapsed time limit' (All, Over 10 ms, Over 30 ms, Over 60 ms), and a date range from 'Performed Start Time' (All, Today, Last 7 days) to 'Elapsed time' (All, Over 10 ms, Over 30 ms, Over 60 ms). A search bar and an 'Apply' button are also present. The main area displays a table of query logs. The table has columns: Status, Job name, Application ID, Queue, Username, Started time, and Elapsed time. There are 50027 lists. The table contains 15 rows of successful queries (SUCCESS status) from various applications and users, such as application_1540_788884137_6346, hive, and jhkim_audit_final.orc.

Status	Job name	Application ID	Queue	Username	Started time	Elapsed time
SUCCESS	SELECT `created`, `modified`, `c1`, `m1` FROM (select * from default.hive_batch_1)	application_1540_788884137_6346	default	hive	2019-05-06 17:21	12 sec
SUCCESS	SELECT lineitem.* FROM tpch_10.lineitem lineitem	-	-	hive	2019-05-06 17:20	1 sec
SUCCESS	DESCRIBE FORMATTED tpch_10.lineitem	-	-	hive	2019-05-06 17:20	735ms
SUCCESS	SHOW TABLES IN tpch_10	-	-	hive	2019-05-06 17:20	256ms
SUCCESS	SELECT `jicode`, `price`, `build_year`, `trade_year`, `trade_month`, `trade_day` ,	application_1540_788884137_6346	default	hive	2019-05-06 17:20	15 sec
SUCCESS	SELECT `created`, `modified`, `c1`, `m1` FROM (SELECT * FROM hive_batch_test_9)	application_1540_788884137_6345	default	hive	2019-05-06 17:20	12 sec
SUCCESS	SELECT datflowtest1_snapshot1.* FROM default.datflowtest1_snapshot1 datflowtest1_	-	-	hive	2019-05-06 17:20	715ms
SUCCESS	SELECT `jicode`, `price`, `build_year`, `trade_year`, `trade_month`, `trade_day` ,	-	-	hive	2019-05-06 17:20	1 sec
SUCCESS	SELECT apartment_trade.* FROM realty.apartment_trade apartment_trade	-	-	hive	2019-05-06 17:20	541ms
SUCCESS	SELECT addrlist.* FROM default.addrlist addrlist	-	-	hive	2019-05-06 17:20	385ms
SUCCESS	SELECT jhkim_audit_final.orc.* FROM cazen_lee.jhkim_audit_final.orc jhkim_audit_f	-	-	hive	2019-05-06 17:18	715ms
SUCCESS	SELECT excsales_snapshot_99.* FROM cazen_lee.excsales_snapshot_99 excsa	-	-	hive	2019-05-06 17:18	951ms
SUCCESS	SELECT cazen_log_click.* FROM cazen_lee.cazen_log_click cazen_log_click	-	-	hive	2019-05-06 17:16	7 sec
SUCCESS	SHOW TABLES IN cazen_lee	-	-	hive	2019-05-06 17:16	443ms
SUCCESS	SELECT 1	-	-	hive	2019-05-06 17:16	651ms

- Status:** Filters queries by whether they were successful or failed.
- Limited elapsed time:** Filters queries by long running time. You can set a reference time for this filtering.
- Performed start time:** Determines a time range by which to filter queries. This time range is based on when each query started running.
- Search by job or application:** Searches the query history by query statement or application ID.
- Number of entries:** Displays how many queries are returned in the list.
- Job list:** Lists queries filtered by specified criteria. Click an entry in the list to view its details.

Query details

Click a query listed in the job log home to view details on that query. The following information can be viewed in the details page.

The screenshot shows the Metatron Discovery interface. At the top, there's a navigation bar with the title 'METATRON DISCOVERY' and a user icon. Below it, a search bar contains the query 'SELECT * FROM druid.'from_csv''. To the right of the search bar, it says 'Recently performed on 2019-05-05 20:04 by metatron'. The main content area is divided into sections: 'Log Information' and 'Query Information'.

Log Information

Status	SUCCESS
Log	No log
Job name	SELECT * FROM druid.'from_csv'
Started time	2019-05-05 20:04
Elapsed time	39ms
User	metatron

Query Information

Connection	Type	DRUID
	Host	metatron-hadoop-02
	Port	8082
	JDBC URL	jdbc:avatica:remote:url=http://metatron-hadoop-02:8082/druid/v2/sql/avatica/

Recent history of the same connection

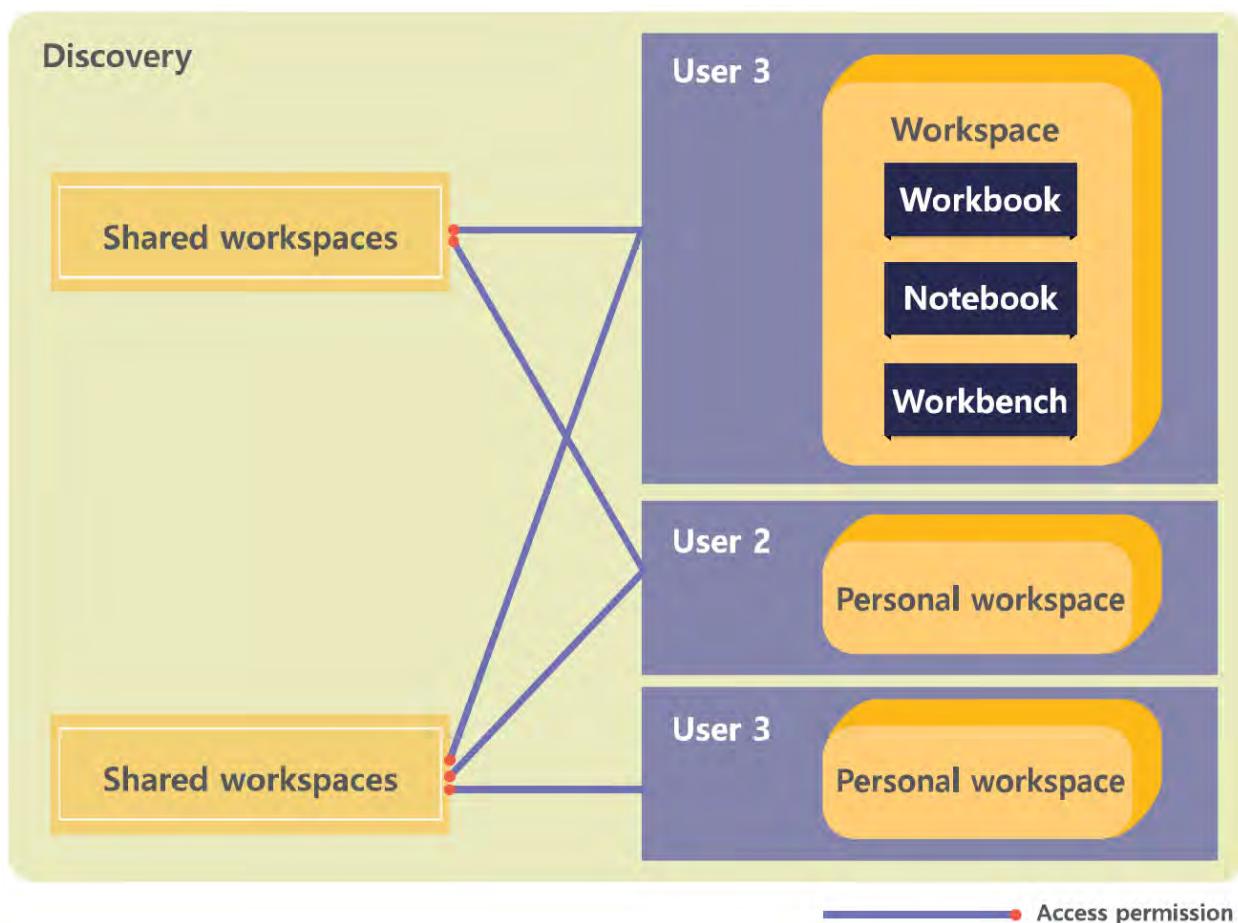
Query date	User	Elapsed time	Result	Action
2019-05-05 20:04	Metatron	39 ms	SUCCESS	Detail >
2019-05-03 14:26	Metatron	24 ms	SUCCESS	Detail >
2019-05-01 04:02	Metatron	40 ms	SUCCESS	Detail >
2019-05-01 03:59	Metatron	29 ms	SUCCESS	Detail >
2019-05-01 03:59	Metatron	29 ms	SUCCESS	Detail >

Plan [See query plan](#)

1. **Status:** Displays whether the query was successful or failed.
2. **Job name:** Statement used to perform the query.

3. **Start time:** Time when the query started running.
4. **Elapsed time:** Time taken to perform the query.
5. **User:** User ID who performed the query.
6. **Connection:** For a query performed in a workbench, the connection information of the database is displayed.
7. **Recent history of the same connection:** For a query performed in a workbench, the latest five queries performed in the database and their results are displayed. Click Detail to pop up a window showing the query statement.
8. **Plan:** Implements the query plan.

WORKSPACE



A workspace stores Metatron Discovery's analytics entities such as workbooks, notebooks, and workbenches. There are two types of workspaces: personal and shared workspaces.

- **Personal workspace:** A private workspace assigned to each Discovery member. It is accessible only to the owner.

- **Shared workspace:** A public workspace shared by multiple users. It is used for users to share analytics processes and results with each other. The owner or administrator of a shared workspace can grant various levels of access to Discovery members.

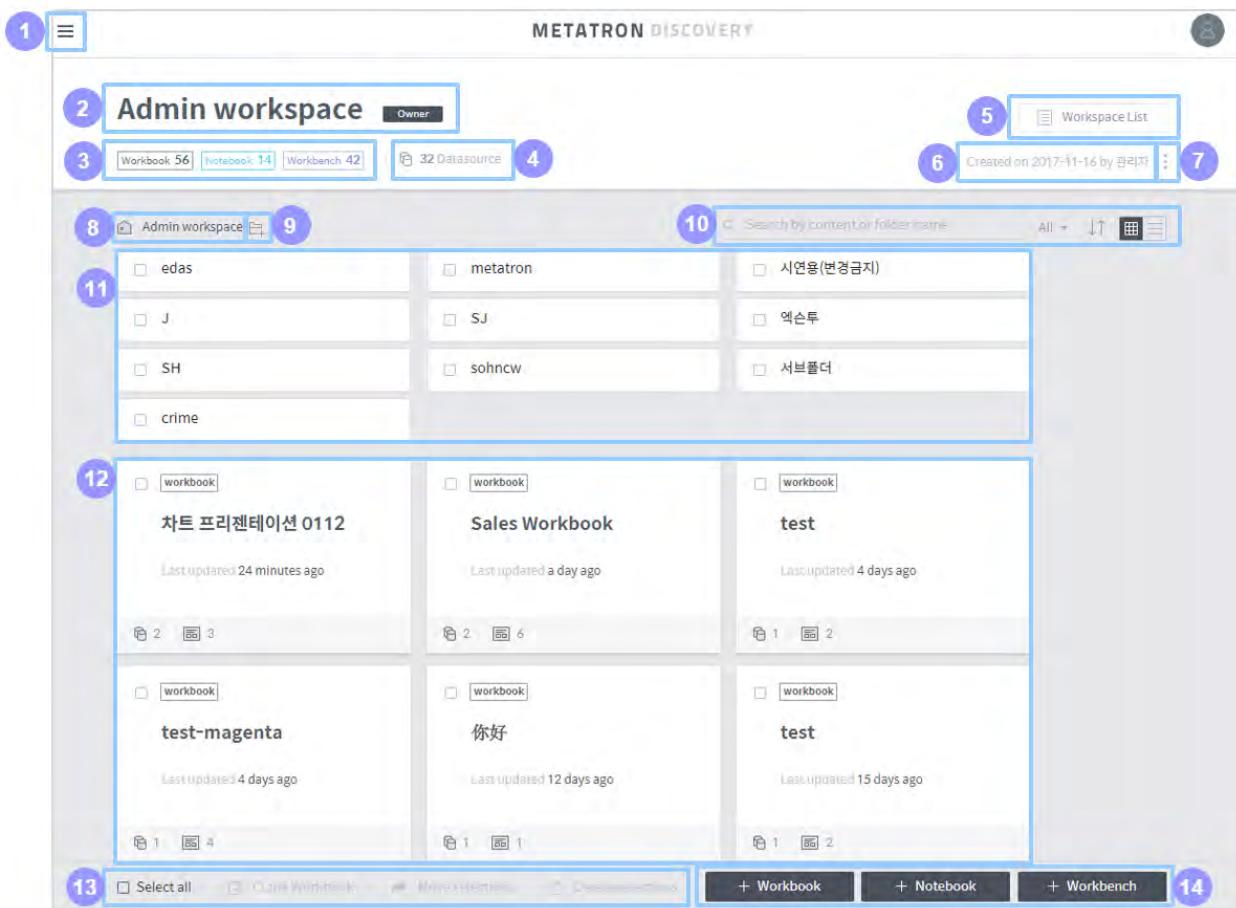
This chapter introduces **workspace home page and UI**, and then how to use **shared workspaces**.

4.1 Workspace home

On the workspace home page, you can perform manage the Metatron Discovery entities (workbooks, notebooks and workbenches) contained in the workspace.

4.1.1 Composition of the workspace home

The overall composition of the workspace home is as follows:



1. **Main menu button:** Click this button to open a panel to access another workspace.

2. **Workspace information:** Displays the name and description of the workspace. If the logged-in user owns the workspace, an Owner icon will be displayed next to the name of the workspace.
3. **Registered entities:** Displays the number of entities registered in the workspace by entity type.
4. **Data source:** Displays the number of data sources used in the workspace. Click this area to show a list of these data sources.
5. **Workspace list:** Click this button to show a list of shared workspaces. (See [Shared workspace list](#) for how to handle it.)
6. **Creation information:** Displays who and when created the workspace.
7. **More:** Edit the settings of the workspace.
 - **Edit the name and description:** Edits the name and description of the workspace.
 - **Set shared member & group:** Sets the users and groups who can access the workspace. (See [Set access permissions for a shared workspace](#) for details.)
 - **Set notebook server:** Sets access information for external analytics tool servers used by the Notebook module.
 - **Set permission schema:** Sets the access permission of each user role for the workspace. (See [Set access permissions for a shared workspace](#) for details.)
 - **Change owner:** Changes the owner of the workspace.
 - **Delete workspace:** Deletes the workspace.
8. **Path in the workspace:** Displays the current location in the workspace. Click on a parent folder listed in the path to move to that folder.
9. **Create a folder:** Click on it to create a new folder in the current location.
10. **Filter/sort the entity list:**
 - **Search:** Searches for an entity or folder in the workspace by name.
 - **Entity type:** Displays only your selected type of entities among workbooks, notebooks, and workbenches.
 - **Sort:** Sorts folders and entities by their name or when they were last updated.
 - **View type:** Select either the grid view or list view as the format of how the entities are listed in the workspace.

11. **Folder list:** Displays folders that meet search criteria in the current location. Click one to enter that folder. (For details on individual folders, see [Folder items](#))
12. **Entity list:** Displays entities that meet search or sorting criteria in the current location. Click an entity to enter its home. (For details on individual entities, see [Entity items](#))
13. **Select/clone/move/delete entity:** Select all entities, or clone, move or delete an entity. (See [Select/clone/move/delete folder and entity](#) for details.)
14. **Create an entity:** Buttons used to create a specific type of entity in the workspace. (For details, see [Create a workbook](#), [Create a notebook](#), and [Create a workbench](#), respectively.)

4.1.2 Folder items

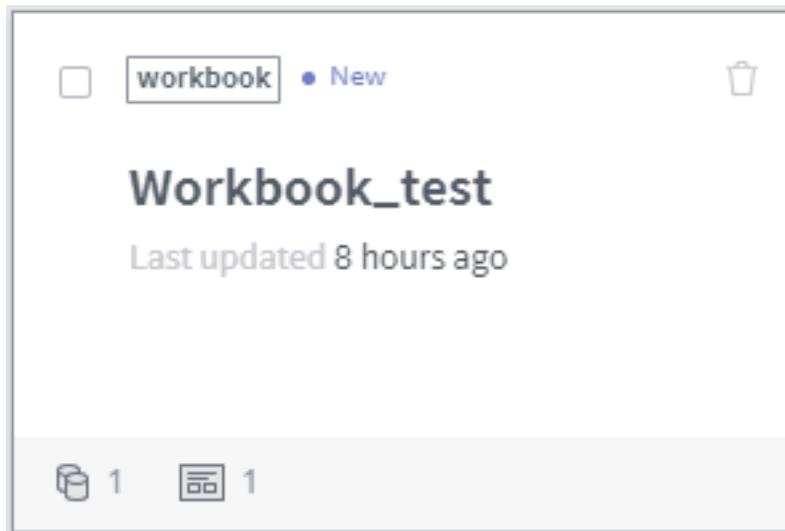
When the mouse cursor is over a folder, it is shown as follows:



- **Check box:** Used to select the folder. You can clone, move or delete the selected folder.
- **Name:** Name of the folder.
- **Edit:** Click on it to modify the name of the folder. This button is displayed only when you hover the mouse over the folder item.
- **Delete:** Click on it to delete the folder. This button is displayed only when you hover the mouse over the folder item.

4.1.3 Entity items

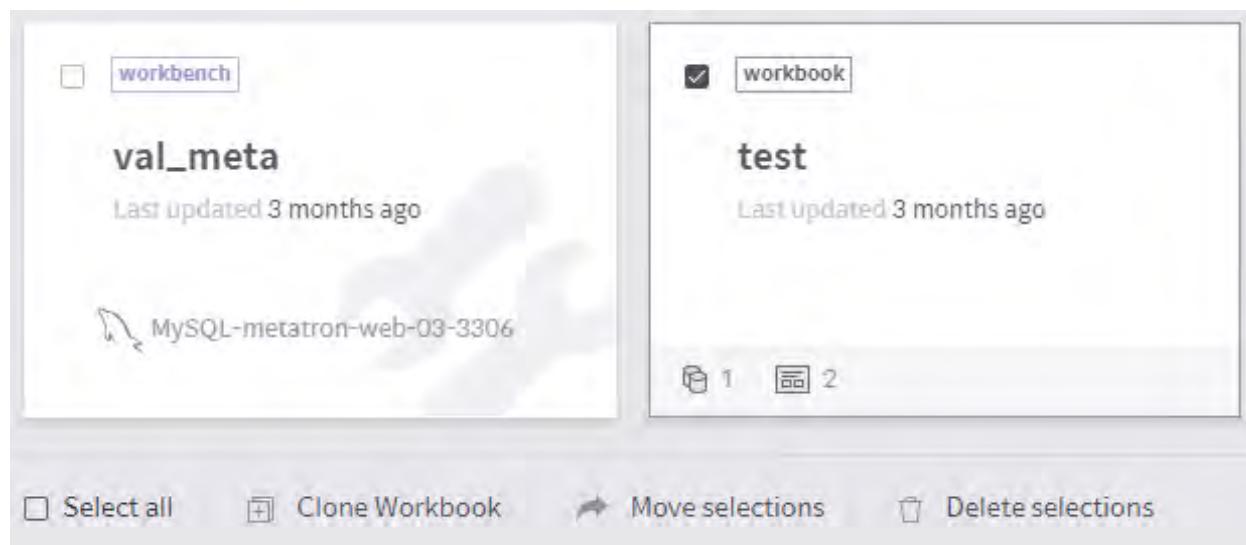
When the mouse cursor is over an entity, it is shown as follows:



- **Check box:** Used to select the entity. You can clone, move or delete the selected entity.
- **Entity type:** Displays the type of the entity (workbook/notebook/workbench).
- **Delete:** Click on it to delete the entity. This button is displayed only when you hover the mouse over the entity item.
- **Name:** Name of the entity.
- **Last updated:** Displays when the entity was last updated.
- **Number of data sources/dashboards:** This is an exclusive area for the workbook type.
 - The number next to the icon refers to how many data sources are connected to the workbook.
 - The number next to the icon refers to how many dashboards are registered in the workbook.

4.1.4 Select/clone/move/delete folder and entity

You can clone, move or delete folders and entities in the workspace. Once you select a folder or entity, the clone, move, and delete buttons in the lower-left corner of the workspace home become active.



- **Select all:** Selects all items in the current folder and entity list.
- **Clone workbook:** This is exclusive for the workbook type. Click this button to clone the selected workbooks.
- **Move selections:** Moves the selected folders and entities. Workbooks can be moved to another workspace, and other types of items can be moved to another folder in the same workspace. However, it is impossible to move selections when workbooks and other types of entities are selected together.
- **Delete:** Deletes the selected folders and entities.

4.2 Shared workspace

A shared workspace is designed for access and use by multiple users. The following subsections describe how to view and create shared workspaces, and explain “permission schema,” which sets which users or groups are allowed to access shared workspaces.

4.2.1 Shared workspace list

The shared workspace list page is used to view a list of all shared workspaces accessible to the logged-in user and to move to a specific workspace. This page can be accessed via two methods:

- Click the button at the top-left of the Discovery screen to open the main panel, and click **Workspace list >>**.

- Click **Workspace list** at the top-right of the workspace home.

The shared workspace list page is composed as follows:

The screenshot shows a list of shared workspaces. At the top, there's a header with a back arrow, a search bar, and several filter buttons. Below the header is a table listing workspaces with columns for name, owner, and various resource counts. A blue box highlights the first row. Callouts numbered 1 through 9 point to specific elements: 1 points to the workspace count in the header; 2 points to the add button; 3 points to the personal workspace link; 4 points to the search bar; 5 points to the favorites filter; 6 points to the public only filter; 7 points to the 'I'm the owner' filter; 8 points to the sorting dropdown; 9 points to the first workspace in the list.

Name	Owner	Workbook	Notebook	Workbench	Members	Groups
테스트 스키마11	Owner	0	0	0	0 Member	0 Group
콜셋 테스트 워크스페이스 (조민정)	Owner	0	0	0	4 Member	0 Group
★ 공유 워크스페이스	Owner	2	0	0	1 Member	0 Group
★ X-test - X-test	Owner	2	0	0	0 Member	0 Group
test-magenta-dictionary	Owner	0	0	1	0 Member	0 Group
★ test-magenta	Owner	7	0	1	4 Member	0 Group
Shared Workspace		1	5	0	3 Member	0 Group
★ samples - sample들을 모아 보자	Owner	1	0	1	1 Member	0 Group

- Number of shared workspaces:** Displays how many shared workspaces are listed.
- Add a shared workspace:** Click this button to move to the page to add a shared workspace. (See [Create a shared workspace](#) for a detailed procedure)
- Personal workspace:** Click this button to move to the personal workspace owned by the logged-in user.
- Search:** Searches the shared workspace list by the name you typed in.
- Favorites:** Displays only those workspaces designated as favorites.
- Public only:** Displays only those workspaces set as public.
- I'm the owner:** Displays only those workspaces for which the logged-in user is the administrator.
- Name ascending/descending:** Sorts the shared workspace list by name ascending/descending.

9. **Workspace list:** Lists workspaces filtered by specified criteria. Click one to move to enter that workspace.

4.2.2 Create a shared workspace

A new shared workspace is created as follows:

1. Click the  button on the shared workspace list page to move the page to create a new shared workspace.
2. Enter a **Name** and **Description**, and then set up the **Permission schema** by referring to the descriptions below:

Create shared workspace

Name
Please enter a name:

Description
Please enter a description:

Permission schema

Use a preset schema Default Schema ▾ Use a custom schema

User roles

User role	Default role	Workbook			Notebook			Workbench			Workspace	
		View	Create	Edit any	View	Create	Edit any	View	Create	Edit any	Create folders	Set config.
Manager		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Editor		✓	✓	-	✓	✓	-	✓	✓	-	-	
Watcher	●	✓	-	-	✓	-	-	✓	-	-	-	

Explanation

- Default role : Role to be granted when adding new members and groups
- View of (item) : Enable to access to item and to read contents
- Create of (item) : Enable to create, modify and delete items
- Edit any of (item) : Enable to create, modify and delete items which is created by other users
- Create folders : Enable to create, modify and delete folders
- Set config. : Enable to edit information and to set configuration of workspace

- **Use a preset schema:** Load the permission schema defined by the administrator.
 - **Use a custom schema:** Define a new permission schema. (See [Set access permissions for a shared workspace](#) for how to define a new permission schema.)
3. Click Done to finish creating a workspace.

4.2.3 Set access permissions for a shared workspace

Setting the access permission for a shared workspace is conducted in the following two steps:

- Set an access permission for each user role (See [Set permission schema](#))
- Grant a role to each user or user group (See [Set shared members & groups](#))

Set permission schema

View permission schema

Click the  icon at the top-right of the shared workspace home and click **Set permission schema** to view the defined permission schema as follows:

Set permission schema

Cancel Done

		User roles of asd						Change schema				
User role	Default role	Workbook			Notebook			Workbench			Workspace	
		View	Create	Edit any	View	Create	Edit any	View	Create	Edit any	Create folders	Set config.
Manager		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Editor		✓	✓	-	✓	✓	-	✓	✓	-	-	
Watcher	●	✓	-	-	✓	-	-	✓	-	-	-	
Guest		✓	-	-	-	-	-	-	-	-	-	

Explanation

- Default role : Role to be granted when adding new members and groups
- View of (item) : Enable to access to item and to read contents
- Create of (item) : Enable to create, modify and delete items
- Edit any of (item) : Enable to create, modify and delete items which is created by other users
- Create folders : Enable to create, modify and delete folders
- Set config. : Enable to edit information and to set configuration of workspace

In the above example, Manager, Editor, Watcher, and Guest are defined as user roles. As shown in this example, a permission schema is a set of user roles defining different access permissions.

What each column determines is as follows:

Default role

When a new user or user group is added, it is assigned the default role.

Permission for each entity type (workbook/notebook/workbench)

- **View:** Allows to access and view data in entities of the type.
- **Create:** Allows to create, edit, and delete entities of the type.
- **Edit any:** Allows to edit or delete entities of the type created by another user.

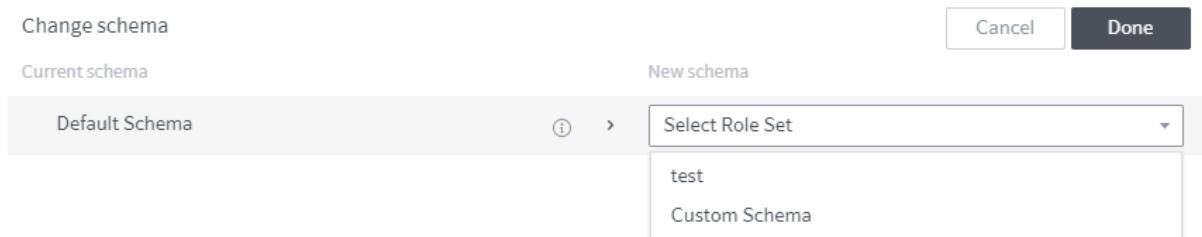
Workspace permission

- **Create folders:** Allows to create, edit, and delete folders in the workspace.
- **Set config.:** Allows to modify the name and description of the workspace and to change the workspace permission schema.

Change permission schema

Click the **Change schema** button on the permission schema view page to move to a page to change the defined permission schema as follows:

Change permission schema



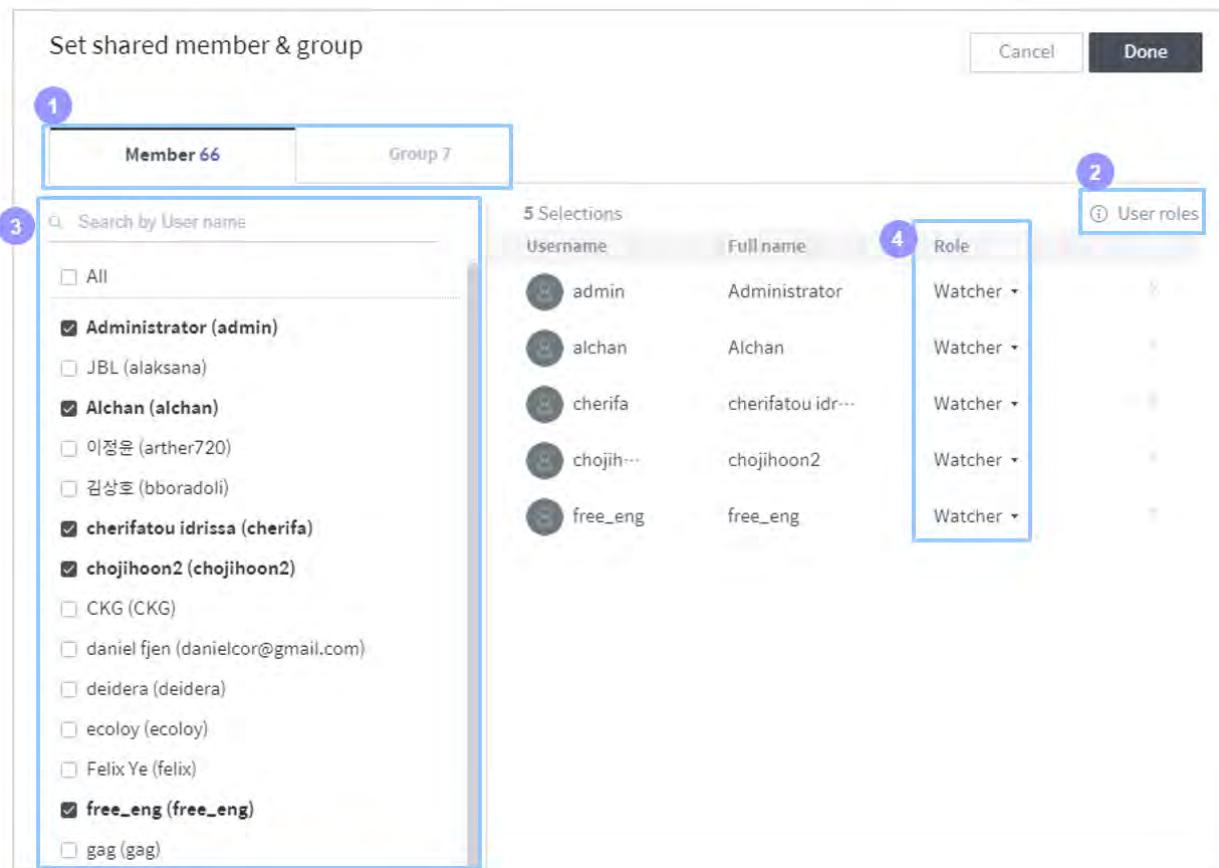
Click **Select Role Set** combo box on the right to display the permission schema defined by the administrator. **Custom schema** at the bottom of the list allows you to set new user roles. Select one to display the following section. (If you select **Custom schema**, you must first define a permission for each user role. Click the button at the right of New schema to move to the permission setting page, and set a permission for each user role by referring to [View permission schema](#))

Current role		New role	
Manager	(i) >	Manager	(i) ▾
Editor	(i) >	Editor	(i) ▾
Watcher	(i) >	Watcher	(i) ▾
Guest	(i) >	Watcher	(i) ▾

Here, each user role of the current permission schema is substituted with the user role defined in the new permission schema. Hover the mouse over the (i) icon next to the name of a user role to display the permission assigned to the user role. Click **Done** to finish setting the permission schema.

Set shared members & groups

Click the icon at the top-right of the shared workspace home, and click **Set shared member & group** to move to a page to set members and groups for the shared workspace as follows: On this page, each user or user group is assigned a user role defined in the permission schema. Assign user roles by referring to the following explanation, and click **Done** to finish setting workspace access permissions.



1. Select whether to assign user roles individually or in groups

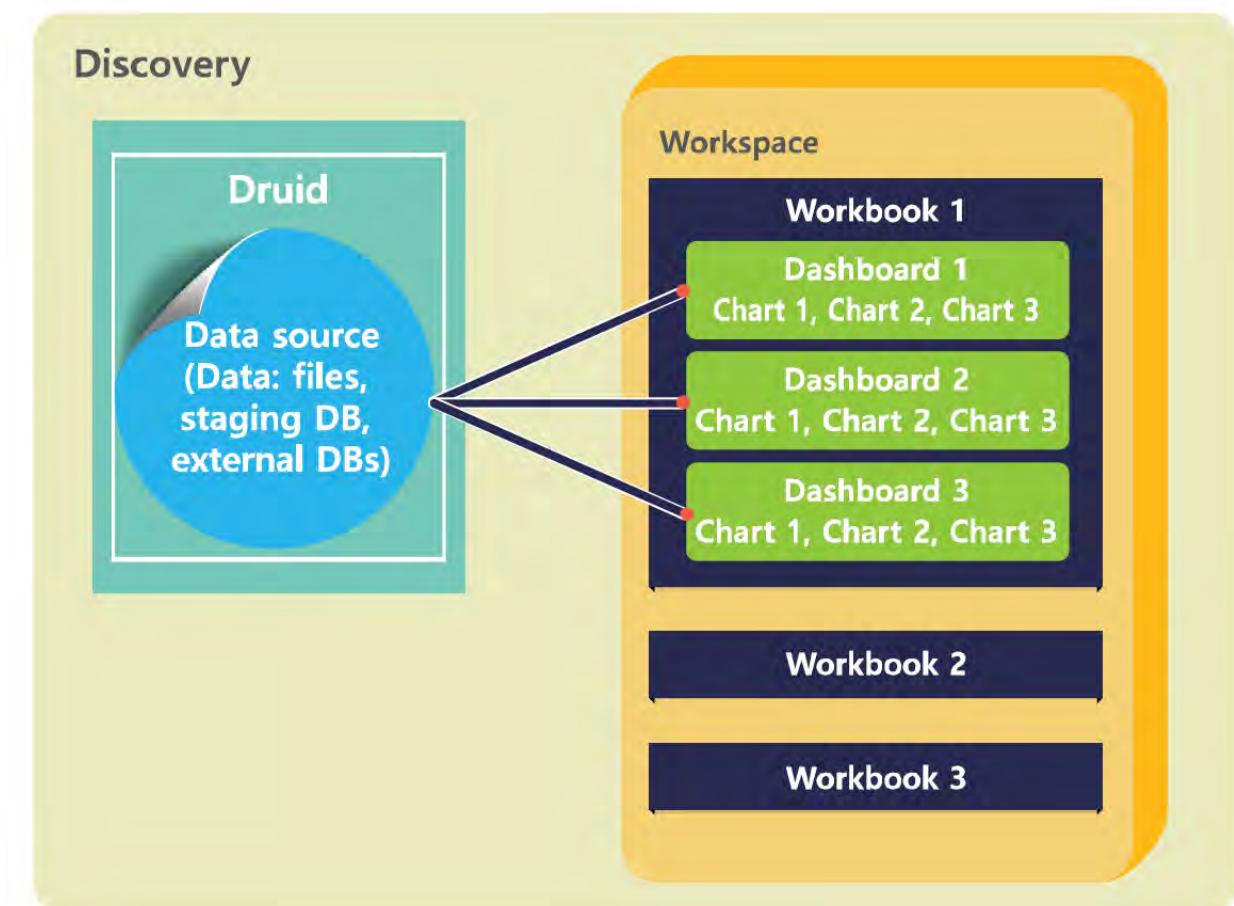
- **Member tab:** Assign user roles to individual users.
- **Group tab:** Assign user roles in groups. (A user group can be established by administrator permission.)

2. User roles: Click on it to pop up a dialog box showing the permission schema, which defines a permission for each user role.

3. Member/group list: Lists the users (groups in the case of the group tab) registered in Discovery. Click a user (group) in the list to add it to the role assignment section on the right. Click an added user (group) to remove it from the section on the right.

4. Assign a user role: Click this combo box to display user roles defined in the active permission schema. Select the role you want to assign to the user (group).

WORKBOOK



Workbook is a data visualization module powered by the Metatron Druid engine. As shown in the diagram above, each **workbook**? a standalone report? consists of multiple **dashboards**, while each dashboard consists of various **charts** showing a visualization of source data analysis.

The main features of Workbook are as follows:

- Fast and flexible data analytics over time-series multidimensional data sources.
- Dashboards contain a variety of visualized charts and texts to be compiled into a report for presentations.
- Frequently used algorithms such as clustering, prediction lines, and trend lines can be implemented through a GUI (graphical user interface).

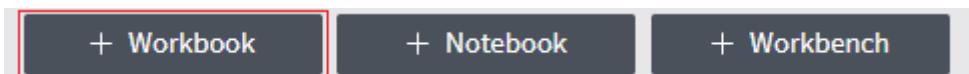
This chapter consists of:

5.1 Create a workbook

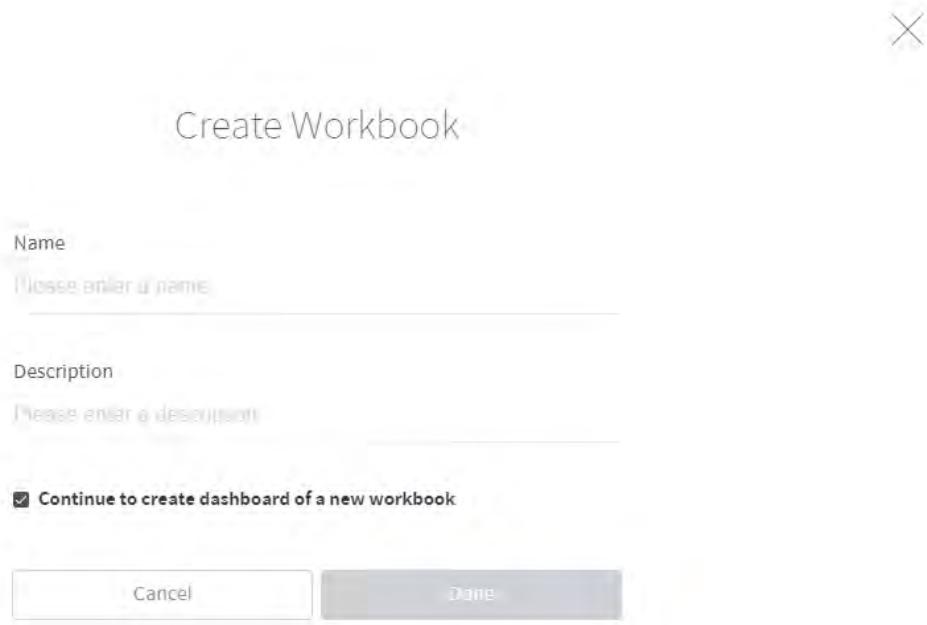
In Metatron Discovery, a **workbook** functions as a standalone data analytics report. Once a workbook is created, you can store a number of **dashboard** slides in the workbook and present them in the proper order.

A workbook is created as follows:

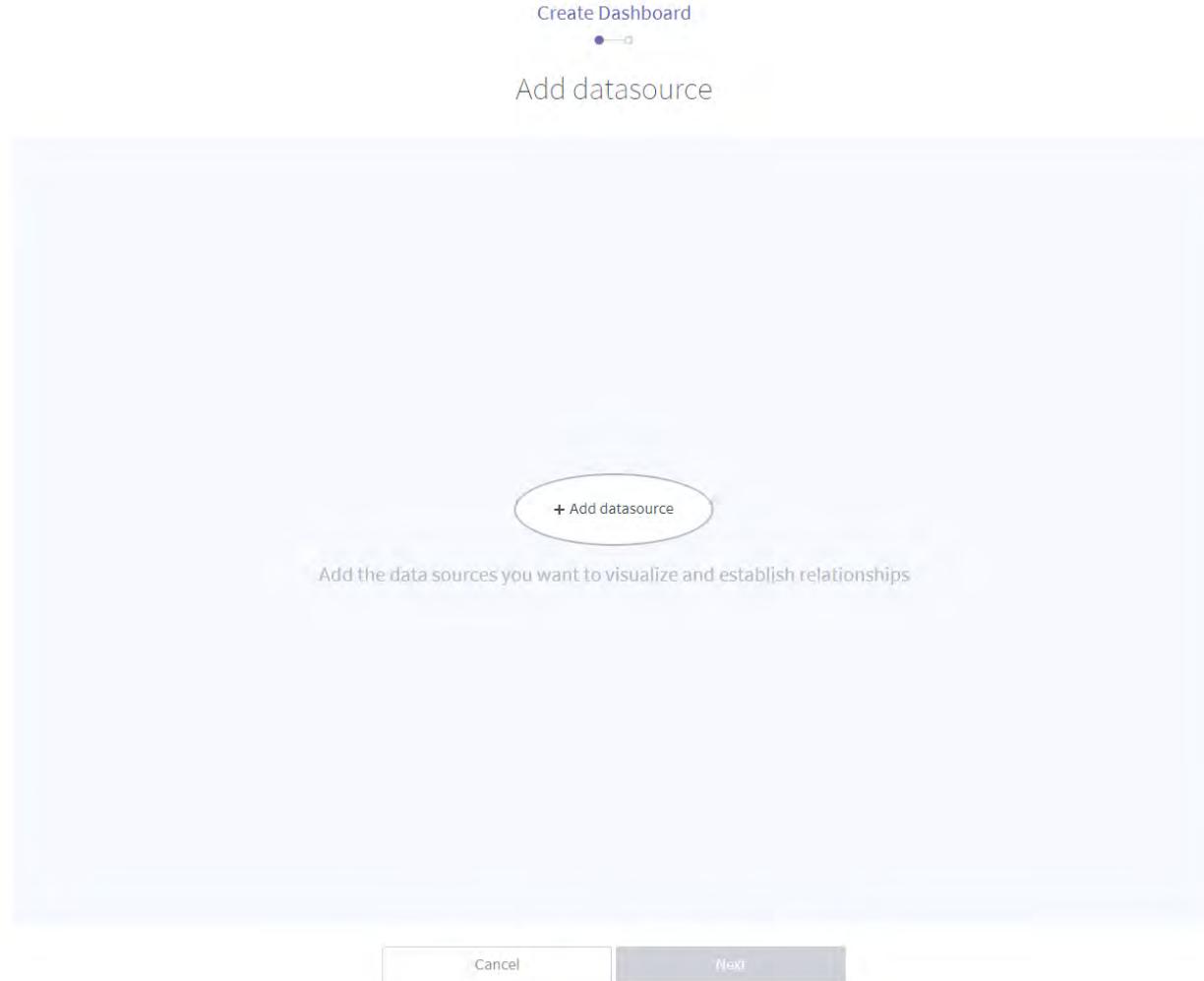
1. Click the **+ Workbook** button at the bottom of the workspace to move to the workbook creation page.



2. Enter a name (required) and description for the workbook to be created and click **Done**. If you select **Continue to create a dashboard of a new workbook**, you'll proceed directly to the **Create Dashboard** page. This option is provided because a workbook cannot work without dashboards in it.



3. After clicking the “+ Add Data Source” button in the middle of the screen, select a data source to create a dashboard. For details on how to create a dashboard, refer to [Create a dashboard](#).



Please select a datasource

No.	Datasource	Type
44	mysql_preset_engine_dialog_single_all	Ingested type
43	Sales Report - A summary of sal...	<input type="button" value="Open data"/>
42	3.2 집중테스트 통계 - Feat. Trello	Ingested type
41	geo	<input type="button" value="Open data"/>
40	uk_cust_basic - Basic Informati...	<input type="button" value="Open data"/>
39	hive_date - asdfasdfasdfasdfas	Ingested type
38	판매현황 데이터 - 2010-2011 판매...	<input type="button" value="Open data"/>
37	saleswithcity - 도시가 추가된 매출 ...	<input type="button" value="Open data"/>
36	범죄발생지 2016	Ingested type
35	Test	Ingested type
34	druid_linked_query	Linked type
33	druid_linked	Linked type
32	access_log_table-link	Linked type
31	3	Ingested type
30	0002	Ingested type
29	audit_test	Ingested type
28	n	Ingested type
		More ▾

mysql_preset_engine_d...

Metadata name: mysql_preset_engine_dialog_single_all

Description:

Type: Ingested type

Visibility: Private

Created: 2019-05-06

Dimensions:

- event_time
- ab.activity_action
- ab.activity_actor
- ab.activity_actor_type
- ab.activity_generator_name
- ab.activity_generator_type
- ab.activity_object_id
- ab.activity_object_type

Measure:

- # id

4. You can check the new workbook in the workspace home as shown below. Click the workbook to enter it.

Admin Workspace

Owner

Workbook 66 Workbench 33 | 78 Datasource

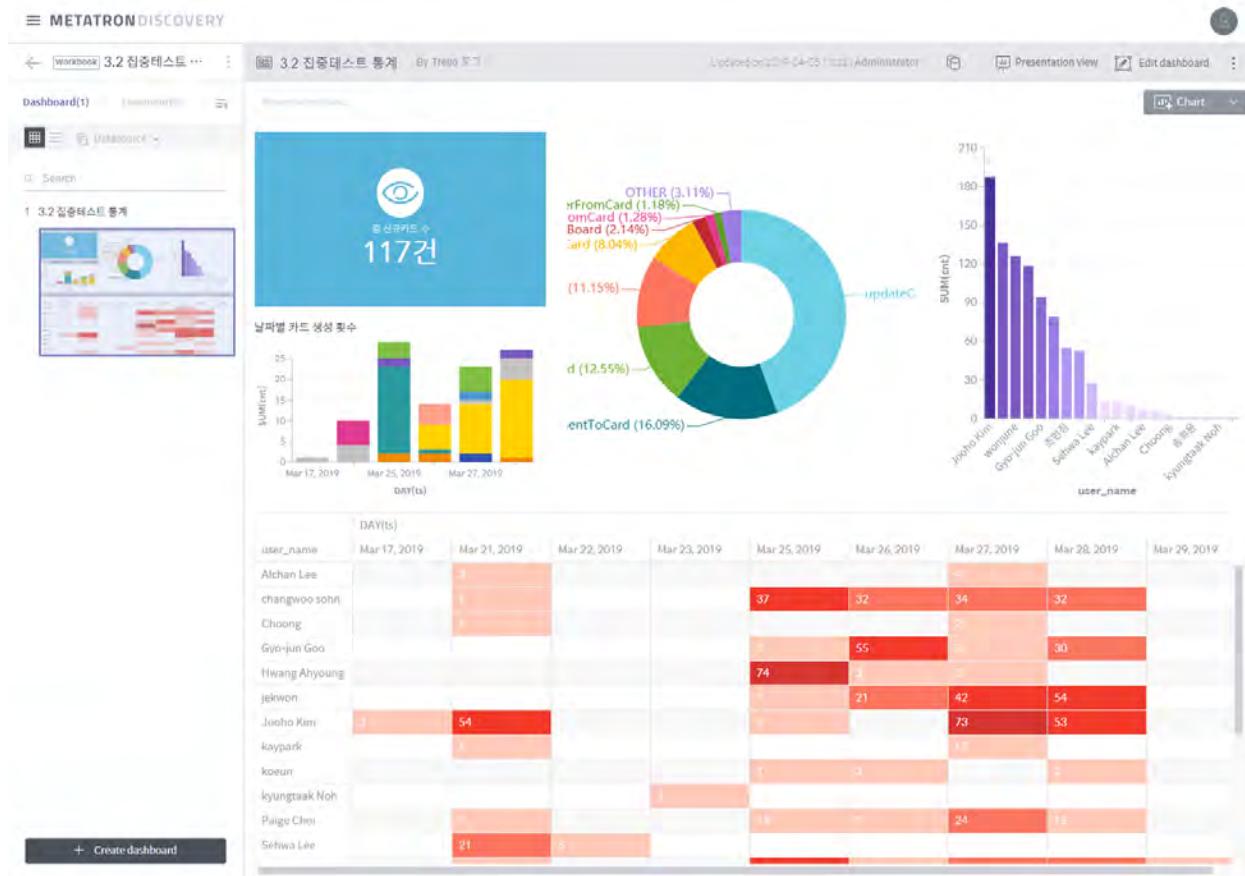
The screenshot shows the Admin Workspace interface. At the top, there are navigation tabs: 'Workbook 66', 'Workbench 33', and '78 Datasource'. Below this is a header bar with the title 'Admin Workspace' and a refresh icon. The main area is a grid of items:

<input type="checkbox"/> new folder	<input type="checkbox"/> taehui	
<input type="checkbox"/> esezin	<input type="checkbox"/> 325	
<input type="checkbox"/> new folder	<input type="checkbox"/> comefeel	
<input type="checkbox"/> eltriny	<input type="checkbox"/> sting	
<input type="checkbox"/> heesoo	<input type="checkbox"/> sohncw	
<input type="checkbox"/> workbook * New	<input type="checkbox"/> workbook * New	
test Last opened a minute ago	Workbook_test Last opened an hour ago	

Below the grid, each item has a small preview pane. The 'test' item shows 0 rows and 0 columns, while the 'Workbook_test' item shows 1 row and 1 column.

5.2 Dashboard

Stored in a workbook, a **dashboard** provides functions to analyze and visualize its connected data source as needed. Therefore, an important step to create a dashboard is connecting to a data source.

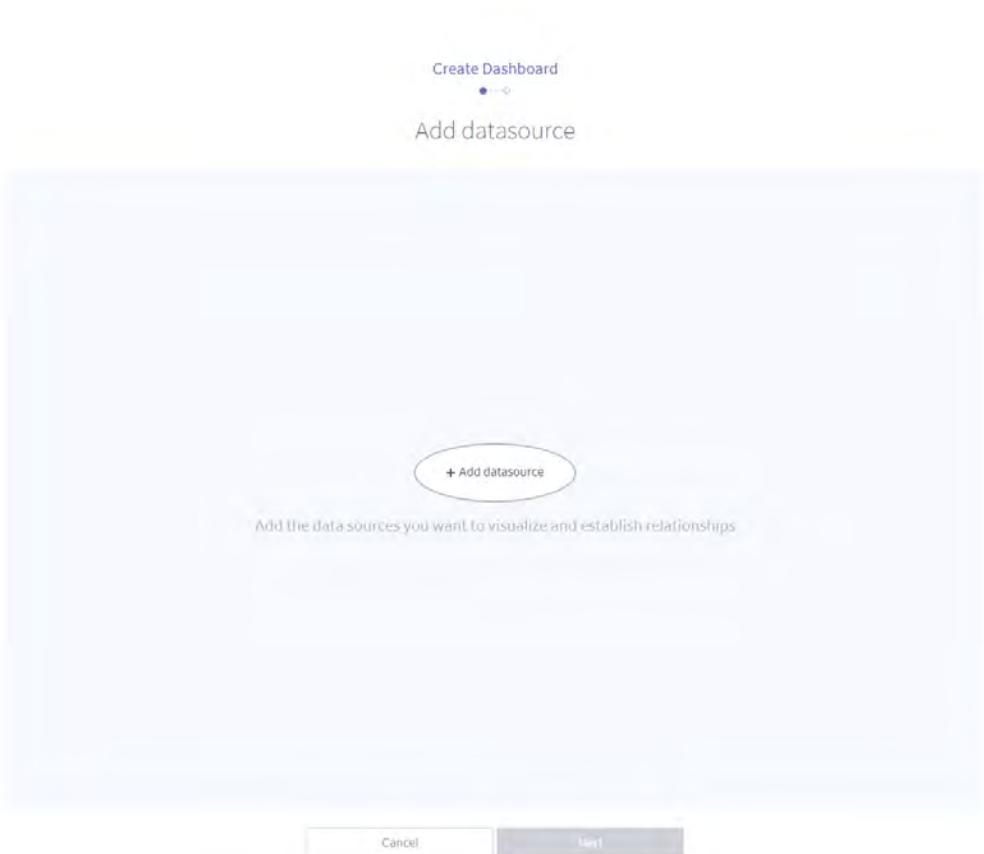


You can visualize analyses of various data sources into charts and texts; those visualizations are customizable using pivoting, chart mapping, and filtering.

5.2.1 Create a dashboard

A dashboard is created as follows:

1. Click **+ Add data source** on the workbook screen.



2. From the list of data sources accessible to the workspace, select the master data sources to which you want to connect the dashboard. In a subsequent step, you can select additional data sources to be joined to these master data sources selected here.

Please select a datasource

[Cancel](#)

[Done](#)

No.	Datasource	Type
<input type="checkbox"/> 44	mysql_preset_engine_dialog_single_all	Ingested type
<input checked="" type="checkbox"/> 43	Sales Report - A summary of sa...	Open data Ingested type
<input type="checkbox"/> 42	3.2 집중테스트 통계 - Feat, Trello	Ingested type
<input type="checkbox"/> 41	geo	Open data Ingested type
<input type="checkbox"/> 40	uk_cust_basic - Basic Informati...	Open data Ingested type
<input type="checkbox"/> 39	hive_date - asdfasdfasfasdfasdf	Ingested type
<input checked="" type="checkbox"/> 38	판매현황 데이터 - 2010-2011 판매...	Open data Ingested type ✓
<input type="checkbox"/> 37	saleswithcity - 도시가 추가된 매출 ...	Open data Ingested type
<input type="checkbox"/> 36	범죄발생지 2016	Ingested type
<input type="checkbox"/> 35	Test	Ingested type
<input type="checkbox"/> 34	druid_linked_query	Linked type
<input type="checkbox"/> 33	druid_linked	Linked type
<input type="checkbox"/> 32	access_log_table-link	Linked type
<input type="checkbox"/> 31	3	Ingested type
<input type="checkbox"/> 30	0002	Ingested type
<input type="checkbox"/> 29	audit_test	Ingested type
<input type="checkbox"/> 28	n	Ingested type
		More ▾

판매현황 데이터

Metadata name: 판매현황 데이터
 Description: 2010-2011 판매현황 데이터입니다.
 Type: Ingested type
 Visibility: Public
 Created: 2019-04-15
 Size: 189.58 KB
 Rows: 63

Dimension: [GeoPoint](#)
[OrderDate](#)
[Category](#)
[City](#)
[Country](#)
[CustomerName](#)
[Discount](#)
[OrderID](#)
[PostalCode](#)
[ProductName](#)
[Profit](#)
[Quantity](#)
[Region](#)
[Sales](#)
[Segment](#)
[ShipDate](#)
[ShipMode](#)
[State](#)
[Sub-Category](#)
[DaystoShipActual](#)
[SalesForecast](#)
[ShipStatus](#)

- **Search by data source name:** Search for a data source accessible to the workspace by name.
- **Show open data only:** Displays only those designated as “open data sources.”
- **Type:** Displays only those data sources that are the connection or collection type.

- **Data source list:** Lists data sources filtered by specified criteria.
 - **Data source information:** Displays brief information of the data source selected in the list.
3. If you have selected more than one data source, you can associate them by dragging one data source to another. Associated data sources can be filtered by each other. If you do not want data source association, simply click **Done**.



4. Once you drag a data source to another one, a new window will pop up to prompt you to configure the data source association. Select a column on each table as an association key by which to filter the other data source. And click **Done**.

Set Association

Cancel

Done

csv_file				dc_with range			
Category		City	Country	Category		City	Country
OrderDate_str	Category	City	Country	OrderDate	Category	City	Country
2011-04-01 00:00:00	Office Supplies	Houston	United States	2011-01-12 00:00:00	Furniture	Dover	United States
2011-06-01 00:00:00	Office Supplies	Philadelphia	United States	2011-01-14 00:00:00	Furniture	Mount Pleasant	United States
2011-08-01 00:00:00	Furniture	Huntsville	United States	2011-01-14 00:00:00	Furniture	San Francisco	United States
2011-08-01 00:00:00	Office Supplies	Huntsville	United States	2011-01-14 00:00:00	Office Supplies	Bossier City	United States
2011-11-01 00:00:00	Furniture	Springfield	United States	2011-01-14 00:00:00	Office Supplies	Bossier City	United States
2011-11-01 00:00:00	Office Supplies	Springfield	United States	2011-01-14 00:00:00	Office Supplies	Bossier City	United States
				2011-01-14 00:00:00	Office Supplies	Bossier City	United States
				2011-01-14 00:00:00	Office Supplies	Newark	United States
				2011-01-14 00:00:00	Office Supplies	Newark	United States
				2011-01-14 00:00:00	Office Supplies	San Francisco	United States
				2011-01-14 00:00:00	Office Supplies	San Francisco	United States
				2011-01-14 00:00:00	Technology	Bossier City	United States
				2011-01-15 00:00:00	Furniture	Philadelphia	United States
				2011-01-16 00:00:00	Technology	Roswell	United States
				2011-01-17 00:00:00	Furniture	Philadelphia	United States
				2011-01-17 00:00:00	Office Supplies	Philadelphia	United States
				2011-01-17 00:00:00	Technology	Philadelphia	United States
				2011-01-19 00:00:00	Office Supplies	Springfield	United States
				2011-01-20 00:00:00	Furniture	Scottsdale	United States
				2011-01-20 00:00:00	Office Supplies	Scottsdale	United States
				2011-01-20 00:00:00	Office Supplies	Scottsdale	United States
				2011-01-21 00:00:00	Furniture	Jonesboro	United States
				2011-01-21 00:00:00	Furniture	Jonesboro	United States

5. Once you have finished setting up associations between the master data sources, click **Done**.



6. Re-configure master data source associations or add other data sources to be joined to the top data source selected above as described below:

Create Dashboard

Setting up relationships between datasources



Edit association *You need to establish relationships between master datasources so that you can link charts*

Data preview Manage Scheme

dc_with range												2.7 MB	28 Columns	1000	/ 1450 Rows	1 Types
OrderDate	Category	City	Country	CustomerName	OrderId	PostalCode	ProductName	Quantity	Region	Segment						
2011-01-12 00:00:00	Furniture	Dover	United States	Seth Vernon	CA-2011-1...	19901	DAXValue U-Cha...	2	East	Consl...						
2011-01-14 00:00:00	Furniture	Mount P...	United States	Natalie DeCherney	CA-2011-1...	29464	Global Highback...	6	South	Consl...						
2011-01-14 00:00:00	Furniture	San Fra...	United States	Brian Dahlen	CA-2011-1...	94109	OSullivan Elevati...	3	West	Consl...						
2011-01-14 00:00:00	Office Supplies	Bossier ...	United States	Chris Selesnick	CA-2011-1...	71111	Brown Kraft Recy...	3	South	Corpo...						
2011-01-14 00:00:00	Office Supplies	Bossier ...	United States	Chris Selesnick	CA-2011-1...	71111	Fellowes Stor/Dr...	6	South	Corpo...						
2011-01-14 00:00:00	Office Supplies	Bossier ...	United States	Chris Selesnick	CA-2011-1...	71111	Staples	2	South	Corpo...						
2011-01-14 00:00:00	Office Supplies	Bossier ...	United States	Chris Selesnick	CA-2011-1...	71111	Staples	3	South	Corpo...						
2011-01-14 00:00:00	Office Supplies	Newark	United States	Michael Moore	CA-2011-11...	43055	Avery Metallic Pol...	2	East	Consl...						

Cancel Next

Master data source association view



- : Click on it to add a new master data source.

- Edit association:** Click on it to edit an established data source association.

Settings panel for individual master data sources (click one of the ovals corresponding to a master data source on the diagram to open it)

- Data preview:** Displays the data table resulting from data source joins.

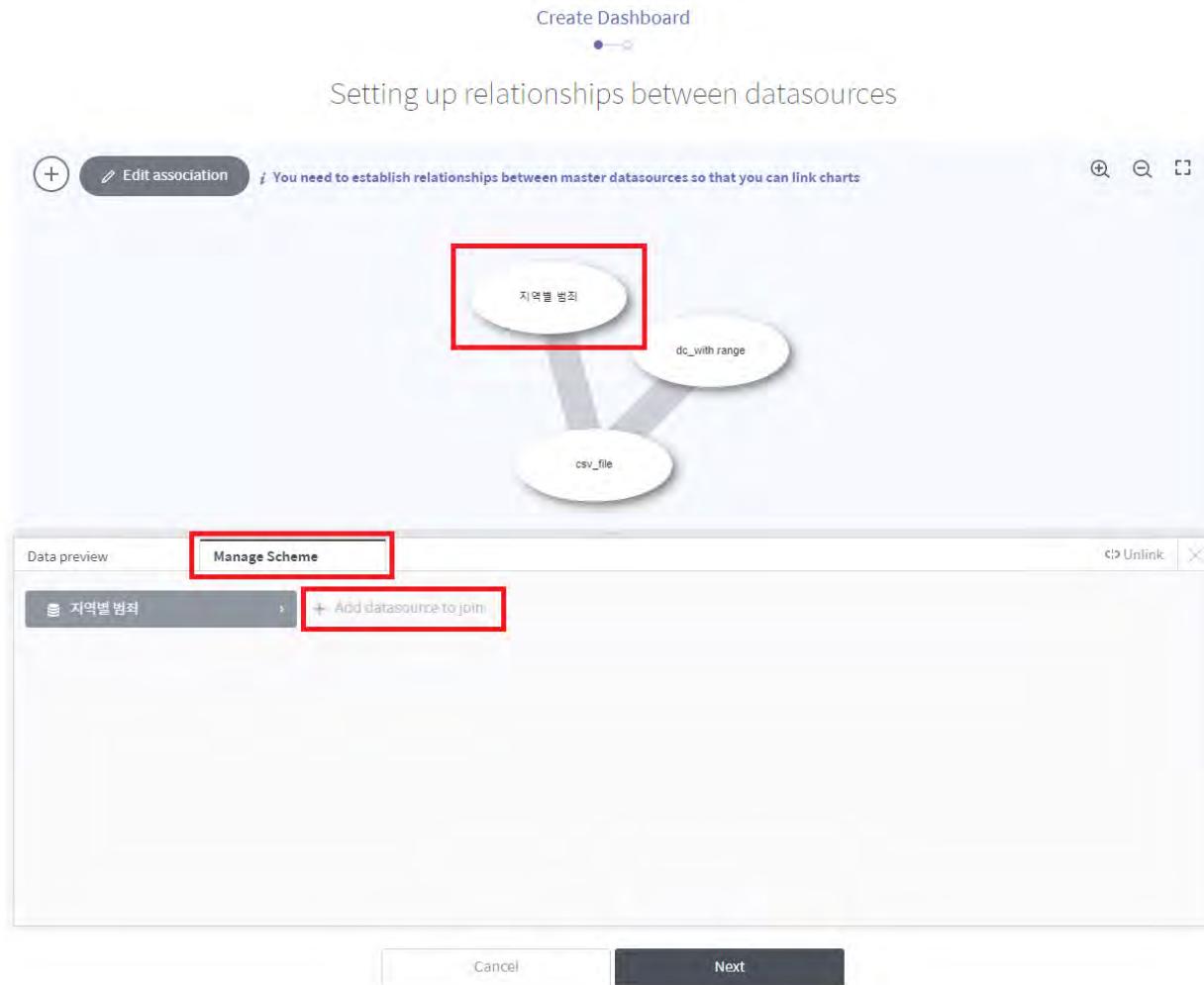
- Manage schema:** Allows you to manage joins to the selected data source (for a detailed procedure, refer to the next step).

- Unlink:** Click on it to remove the selected data source.



- : Click on it to close the panel.

7. To join one of the master data sources to other data sources, click the corresponding oval on the diagram → click the **Manage Schema** tab on the panel at the bottom → click **+ Add a data source to join**.



8. Refer to the description below to set up data joins.

Join

Cancel

Join

Master datasource

지역별 범죄

대분류	서울	부산	광주	세종	cu
교통범죄	74270	32944	22137	1234	
기타범죄	44407	22296	4809	495	
노동범죄	509	209	29	6	
마약범죄	1449	963	75	8	
병역범죄	4120	662	330	131	
보건범죄	3875	2365	249	22	
선거범죄	180	60	7	7	
안보범죄	19	6	8	1	
철도범죄	46861	16777	6050	638	
지능범죄	72137	25052	8896	821	
특별경제…	17109	8134	1616	357	

Datasource to join

Sales Report

GeoPoint	OrderDate	Category	City
29.8941,-95.99	2011-01-04T00:00:00	Office Supplies	Hous
41.7662,-88.17	2011-01-05T00:00:00	Office Supplies	Nape
41.7662,-88.17	2011-01-05T00:00:00	Office Supplies	Nape
41.7662,-88.17	2011-01-05T00:00:00	Office Supplies	Nape
39.9448,-75.12	2011-01-06T00:00:00	Office Supplies	Phila
37.8274,-87.00	2011-01-07T00:00:00	Furniture	Hend
33.9321,-83.00	2011-01-07T00:00:00	Office Supplies	Athe
37.8274,-87.00	2011-01-07T00:00:00	Office Supplies	Hend
37.8274,-87.00	2011-01-07T00:00:00	Office Supplies	Hend
37.8274,-87.00	2011-01-07T00:00:00	Office Supplies	Hend
37.8274,-87.00	2011-01-07T00:00:00	Office Supplies	Hend

Column

Column

Add to join keys

Join type



Inner



Left



Outer



FullOuter

1 join keys

대분류

= Category



Preview results

34 Columns 14 Rows

sales_report.ShipMode	sales_report.PostalCode	sales_report.DaysToShipActual	_wmnxd.대분류	sales_report.ShipSt
			교통범죄	
			기타범죄	
			노동범죄	
			마약범죄	

- **Master data source:** Displays information on the master data source to which you want to join another data source.
- **Datasource to join:** Select a data source to be joined to the master data source.
- **Add to join keys:** A join key defines the join relationship between the master and slave

data sources in each column. Select a column to be joined from each data source, and click this button to add a new join key. For this, the two columns must be of the same data type.

- **Join type:** Select how to join and transform a data source. To help you understand, each join type is explained below using the following tables as an example.

Table 1: Master data source

Product name (join key)	Price
A	\$22.11
B	\$9.23
C	\$8.99
D	\$10.10

Table 2: Data source to be joined

Product name (join key)	Sales
B	100
D	200
E	50

- **Inner:** Imports those records of each data source whose join key column values are present also in the other data source's join key column, joins them, and stores the joined records in the resulting table. (Intersection between two data sources)

Product name (join key)	Price	Sales
B	\$9.23	100
D	\$10.10	200

- **Left:** Imports those records of the right data source (data source to be joined) whose join key column values are present also in the join key column of the left data source (master data source to join), joins them to the left data source records, and stores the joined records in the resulting table. Those records from the right data source whose join key column values are not present in the left data source are discarded.

Product name (join key)	Price	Sales
A	\$22.11	null
B	\$9.23	100
C	\$8.99	null
D	\$10.10	200

- **Right:** Imports those records of the left data source (master data source to join) whose join key column values are present also in the join key column of the right data source (data source to be joined), joins them to the right data source records, and stores the joined records in the resulting table. Those records from the left data source whose join key column values are not present in the right data source are discarded.

Product name (join key)	Price	Sales
B	\$9.23	100
D	\$10.10	200
E	\$null	50

- **Full Outer:** Imports all records from both data sources, join them, and stores the joined records in the resulting table. (Union between two data sources)

Product name (join key)	Price	Sales
A	\$22.11	null
B	\$9.23	100
C	\$8.99	null
D	\$10.10	200
E	\$null	50

- **Preview results:** Displays the data table resulting from data source joins.
9. Confirm the information on the imported data source, enter the **Name** and **Description**, and click **Done** to create a new dashboard.

Create Dashboard
Please complete dashboard creation

○ — ●

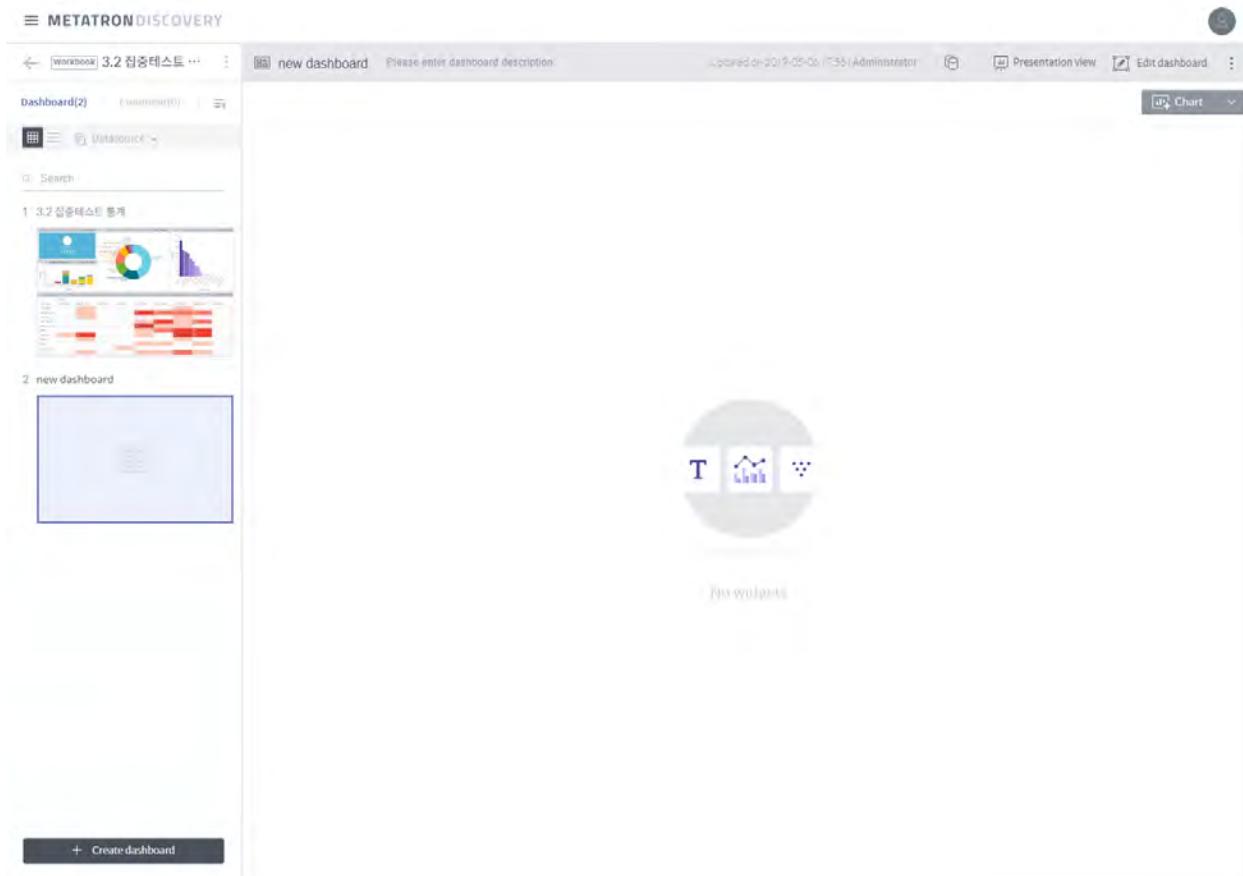
Workbook	3.2 집중테스트 통계
Datasource	지역별 범죄 / sales_report csv_file dc_with range

Name
Please enter a name

Description
Please enter a description

[Previous](#) [Done](#)

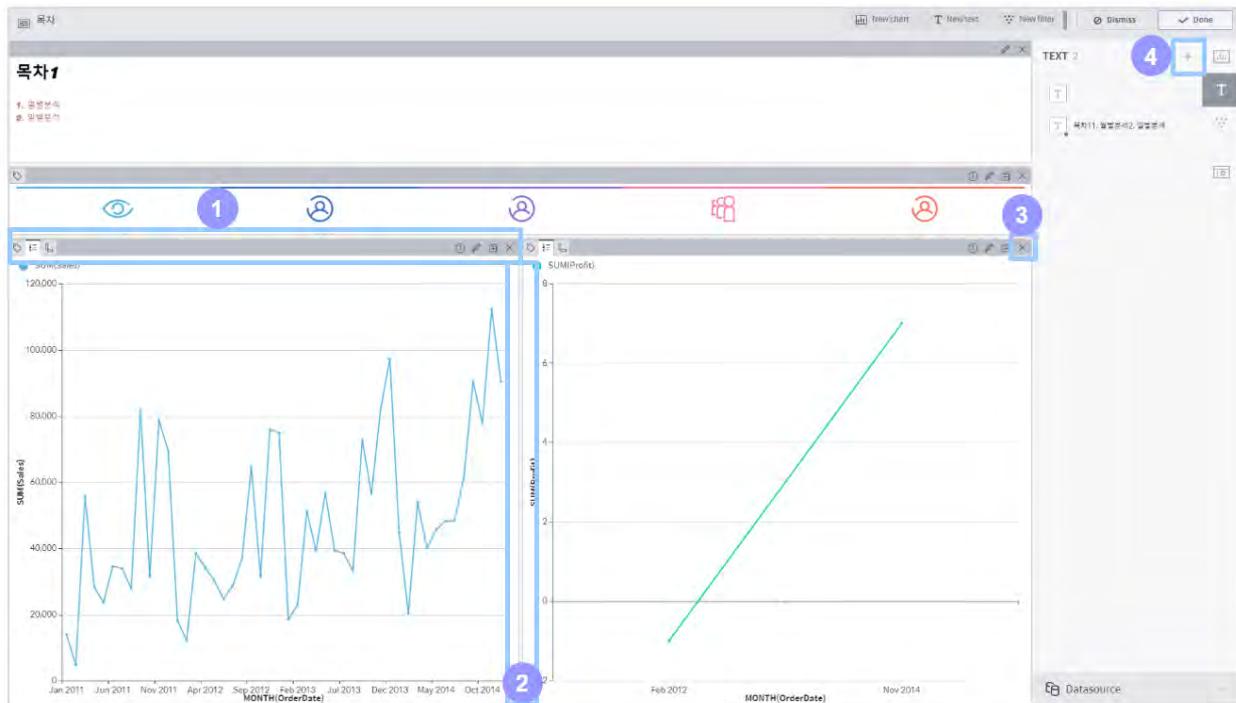
10. The new dashboard will be added to the workbook home. Click the dashboard to display its contents.



5.2.2 Change dashboard size and layout

Click **Edit Dashboard** on the basic dashboard page to go to a page for editing the configuration of the dashboard. In this page, you can add a widget, edit the dashboard, set the hierarchy and change the layout.

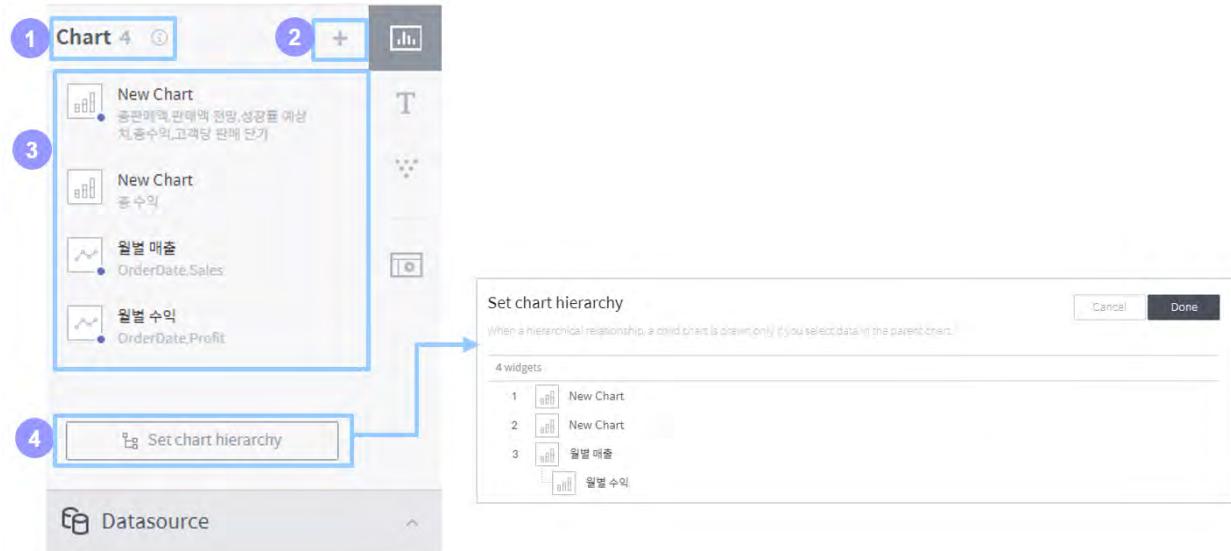
Dashboard widget arrangement settings



1. **Change widget location:** Drag the title of a widget to move the widget.
2. **Adjust widget width:** Move the distance between widgets to adjust their widths.
3. **Add a widget to the display area:** Drag a widget from the widget list on the right panel to the left widget display area to add the widget to the display area.
4. **Delete a widget from the display area:** Click the X button on a widget shown in the widget display area to delete the widget from the display area.

Chart widget panel

On the chart widget panel, you can add/edit/delete a chart in the dashboard.



1. **Number of chart widgets:** Displays how many chart widgets are registered in the dashboard.
2. **Add a chart widget:** Click on it to create a new chart widget in the dashboard.
3. **Chart widget list:** Lists chart widgets registered in the dashboard. Hover the mouse over a widget to display the edit and delete icons. Drag a widget to the widget display area to display the widget in the display area.
4. **Set chart hierarchy:** Click on it to set parent/child relationships between charts in the dashboard. Selecting a data item from the parent chart filters the child chart by the selection. To set a hierarchy, drag the chart to be set as a child under the chart to be set as a parent. Once you finish setting the chart hierarchy, the chart menu is restructured accordingly.

Text widget panel

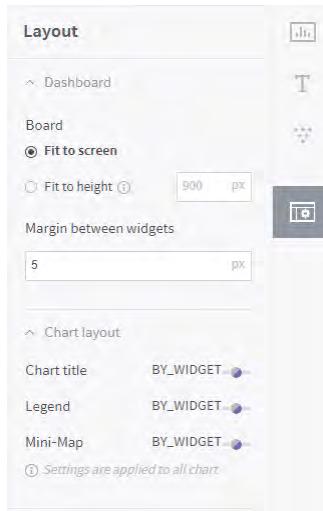
On the text widget panel, you can add/edit/delete a text widget in the dashboard.



1. **Number of text widgets:** Displays how many text widgets are registered in the dashboard.
2. **Add a text widget:** Click on it to create a new text widget in the dashboard.
3. **Text widget list:** Lists text widgets registered in the dashboard. Hover the mouse over a widget to display the edit and delete icons. Drag a widget to the widget display area to display the widget in the display area.

Layout panel

On the layout panel, you can adjust some settings on how to arrange widgets and display each widget in the widget display area.



- **Set board height**
 - **Fix to screen:** Maximizes the height of the dashboard to fill the screen.
 - **Fix to height:** Set the height of the dashboard to a specific pixel value.
 - **Margin between widgets:** Sets the margin between widgets in the widget display area.
- **Chart title:** Sets whether to display the title of each chart and filter widget in the widget display area.
- **Legend:** Sets whether to display a legend for each chart widget in the widget display area.
- **Mini-map:** Sets whether to display a mini-map for each chart widget in the widget display area.

Data source panel

In the data source panel, you can view and edit information on connected data sources, as well as add column filters easily. Click on a filter icon on a dimension or measure on the right-hand side to add a filter.

The screenshot shows the 'Datasource' panel with a dark theme. At the top, there is a 'Sales Report' entry with a trash icon and a dropdown arrow. Below it is a search bar labeled 'Search'. The main area is divided into two sections: 'Dimension' and 'Measure'. The 'Dimension' section is expanded, showing a list of categories: GeoPoint, OrderDate, Category, City, Country, CustomerName, OrderID, PostalCode, and ProductName. The 'OrderDate' item has a checked checkbox next to it. The 'Measure' section is also expanded, showing a list of metrics: Discount, Profit, Quantity, Sales, DaystoShipActual, and SalesForecast. The 'Profit' item has a checked checkbox next to it. Navigation buttons '< Previous' and 'Next >' are located at the bottom of each section.

Please note that the filters you can apply or clear here are global filters applied to the entire dashboard, and those applied or cleared in the chart editor are all chart filters.

5.2.3 Check data sources in a dashboard

Click the  button on the basic dashboard page to display a dialog box displaying information about the data source used in the dashboard. At the top-left corner, you can choose the data source that you want to view. This dialog box consists of three tabs (Data grid, Column detail, Dashboard data information).

Data grid tab

Displays all values in the data source.

Sales Report															Data details - used in sales performance		X												
Data grid		Column detail		①	Download																								
OrderDate		All		Today		Last 7 days		2011-01-04 09:00		~		2014-12-30 09:00		Apply															
<input type="text"/> Search data																													
GeoPoint	OrderDate UTC+9	Category	City	Country	CustomerName	Discount	OrderID	PostalCode	ProductName	Profit	Quantity	Region																	
34.066,-11...	2014-12-30T...	Technology	Los Angeles	United States	James Galang	0.2	CA-2014-1...	90049	Adtran 1202752G1	23	3	W																	
40.8011,-7...	2014-12-30T...	Office Supp...	New York C...	United States	Michael Chen	0	US-2014-1...	10035	Ideal Clamps	3	3	E																	
38.1593,-8...	2014-12-30T...	Office Supp...	Louisville	United States	Katherine Hughes	0	US-2014-1...	40214	Panasonic KP-3...	10	1	S																	
38.1593,-8...	2014-12-30T...	Office Supp...	Louisville	United States	Katherine Hughes	0	US-2014-1...	40214	GBC ProClick Sp...	6	1	S																	
43.012,-85...	2014-12-30T...	Office Supp...	Grand Rapids	United States	Ken Brennan	0	CA-2014-1...	49505	Xerox 1915	101	2	C																	
47.8353,-1...	2014-12-30T...	Office Supp...	Edmonds	United States	Bruce Stewart	0	CA-2014-1...	98026	Acco Glide Clips	10	5	W																	
38.1593,-8...	2014-12-30T...	Furniture	Louisville	United States	Katherine Hughes	0	US-2014-1...	40214	Harbour Creatio...	78	3	S																	
38.1593,-8...	2014-12-30T...	Furniture	Louisville	United States	Katherine Hughes	0	US-2014-1...	40214	Global Leather a...	87	1	S																	
38.1593,-8...	2014-12-30T...	Furniture	Louisville	United States	Katherine Hughes	0	US-2014-1...	40214	DMI Arturo Colle...	314	8	S																	
34.066,-11...	2014-12-30T...	Furniture	Los Angeles	United States	James Galang	0.2	CA-2014-1...	90049	Global High-Bac...	-44	4	W																	
47.8353,-1...	2014-12-30T...	Furniture	Edmonds	United States	Bruce Stewart	0	CA-2014-1...	98026	Hand-Finished S...	21	2	W																	
33.8186,-1...	2014-12-30T...	Furniture	Anaheim	United States	Ben Peterman	0	CA-2014-1...	92804	Nu-Dell Executiv...	37	8	W																	
40.7844,-7...	2014-12-29T...	Technology	New York C...	United States	Jennifer Ferguson	0	CA-2014-1...	10024	Cush Cases Hea...	4	3	E																	
37.7509,-1...	2014-12-29T...	Office Supp...	San Francis...	United States	Kristen Hastings	0	CA-2014-1...	94110	Adjustable Dept...	210	4	W																	
30.5145,-9...	2014-12-29T...	Office Supp...	Round Rock	United States	Greg Hansen	0.2	CA-2014-1...	78664	Stanley Bostitch ...	3	2	C																	
40.7111,-8...	2014-12-29T...	Office Supp...	Peoria	United States	Lori Olson	0.8	CA-2014-1...	61604	Computer Printo...	-3	5	C																	
40.7844,-7...	2014-12-29T...	Office Supp...	New York C...	United States	Jennifer Ferguson	0.2	CA-2014-1...	10024	Storex Dura Pro ...	11	7	E																	
40.7844,-7...	2014-12-29T...	Office Supp...	New York C...	United States	Jennifer Ferguson	0	CA-2014-1...	10024	OIC Bulk Pack M...	6	4	E																	
40.7844,-7...	2014-12-29T...	Office Supp...	New York C...	United States	Jennifer Ferguson	0	CA-2014-1...	10024	Avery 473	35	7	E																	
36.0725,-8...	2014-12-29T...	Office Supp...	Nashville	United States	Erica Hernandez	0.2	CA-2014-1...	37211	Carina Double W...	-13	1	S																	
40.4262,-1...	2014-12-29T...	Office Supp...	Loveland	United States	Pamela Coakley	0.7	US-2014-1...	80538	Avery Reinforce...	-1	2	W																	
46.8564,-9...	2014-12-29T...	Office Supp...	Fargo	United States	Christopher Schild	0	CA-2014-1...	58103	Wilson Jones Im...	13	5	C																	
44.0521,-9...	2014-12-29T...	Office Supp...	Edina	United States	Christopher Schild	0	CA-2014-1...	40179	Staples	1	1	C																	

Column details tab

Displays details about each column of the data source.

Column name	Logical column name
GeoPoint	GeoPoint
OrderDate	OrderDate
Category	Category
City	City
Country	Country
CustomerName	CustomerName
Discount	Discount
OrderID	OrderID
PostalCode	PostalCode
ProductName	ProductName
Profit	Profit
Quantity	Quantity
Region	Region
Sales	Sales
Segment	Segment
ShipDate	ShipDate
ShipMode	ShipMode
State	State
Sub_Category	Sub_Category
DaystoShipActual	DaystoShipActual
SalesForecast	SalesForecast
ShipStatus	ShipStatus
DaystoShipScheduled	DaystoShipScheduled
OrderProfitable	OrderProfitable
SalesperCustomer	SalesperCustomer
ProfitRatio	ProfitRatio
SalesaboveTarget	SalesaboveTarget

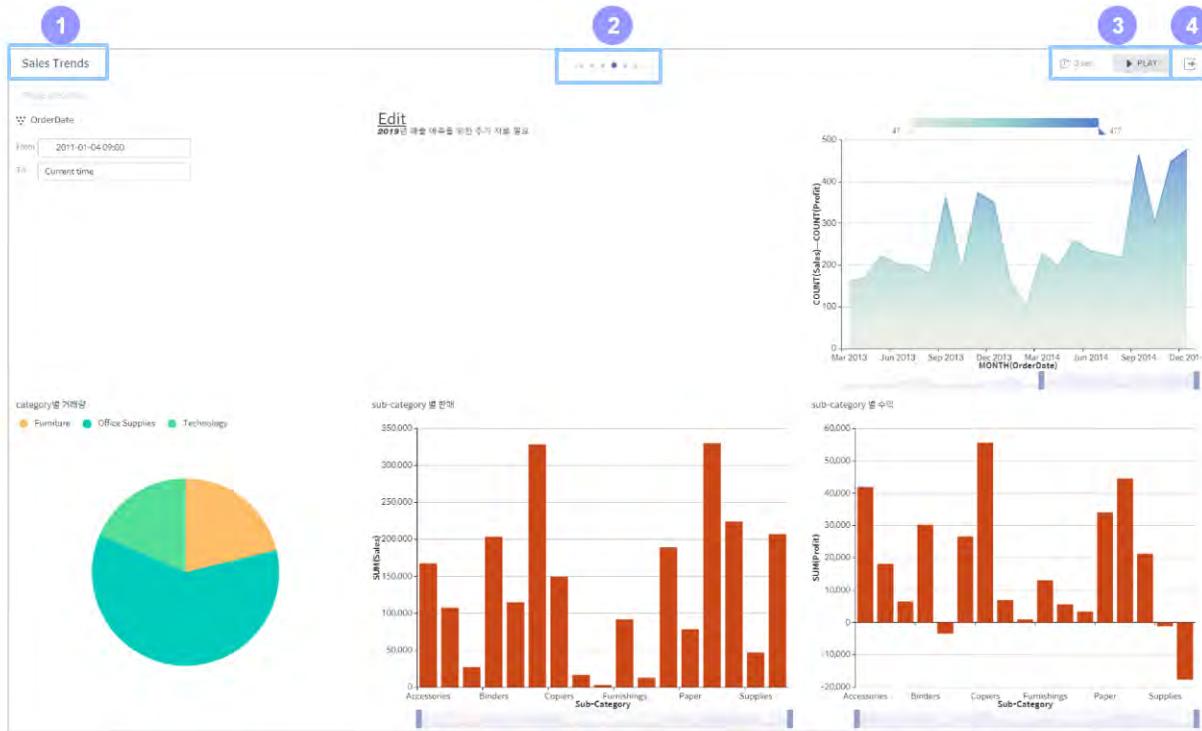
Dashboard data information tab

Displays an overview of the data source.

Dashboard data information	
Last modified	2018-01-15
File size	15.7 MB
Number of columns	29 Columns
Rows	8,887 Rows
Dimensions	19 Dimensions
Measures	9 Measures
Dimension	Sales Report

5.2.4 Presentation with a dashboard

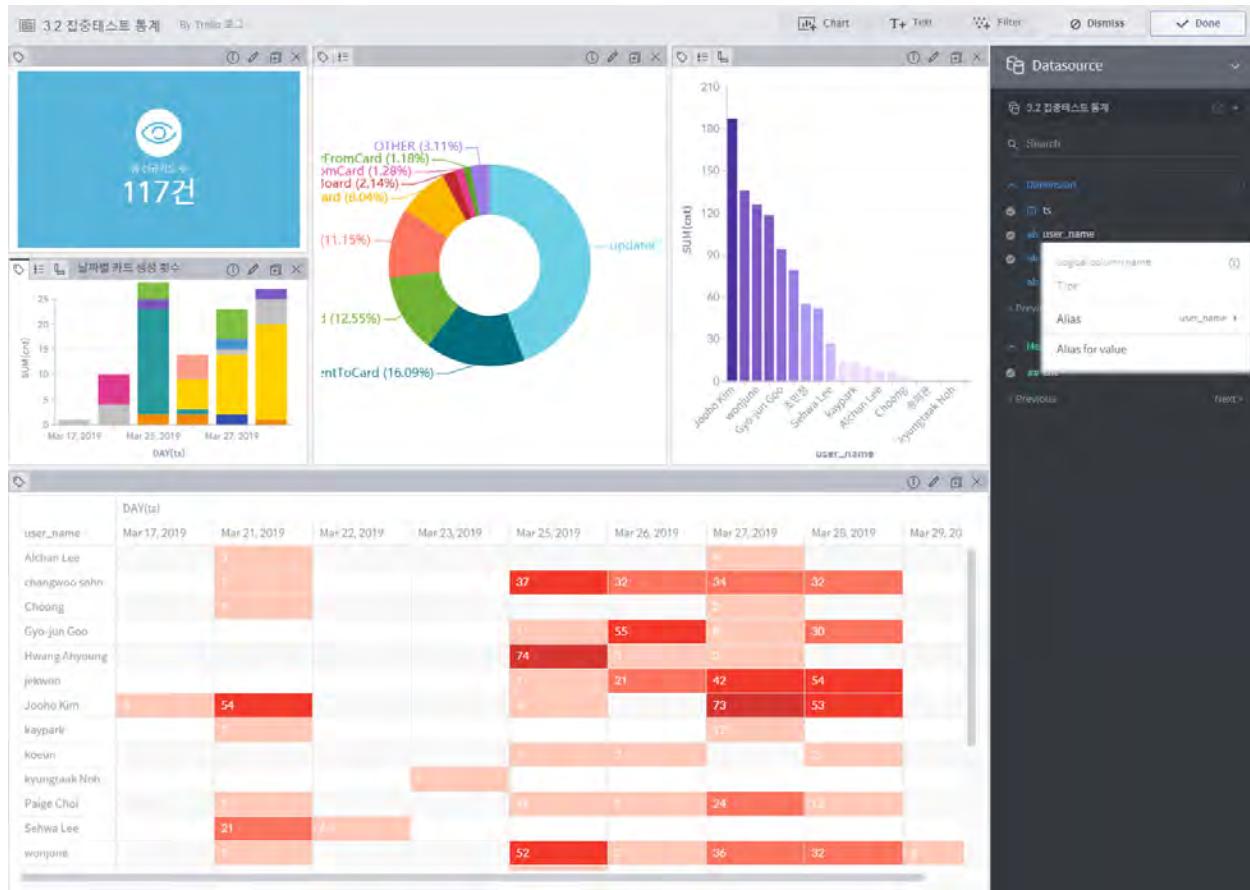
Click **Presentation view** on the basic dashboard page to view workbook dashboards with a presentation UI. In this mode, you can easily report and share data analytics results.



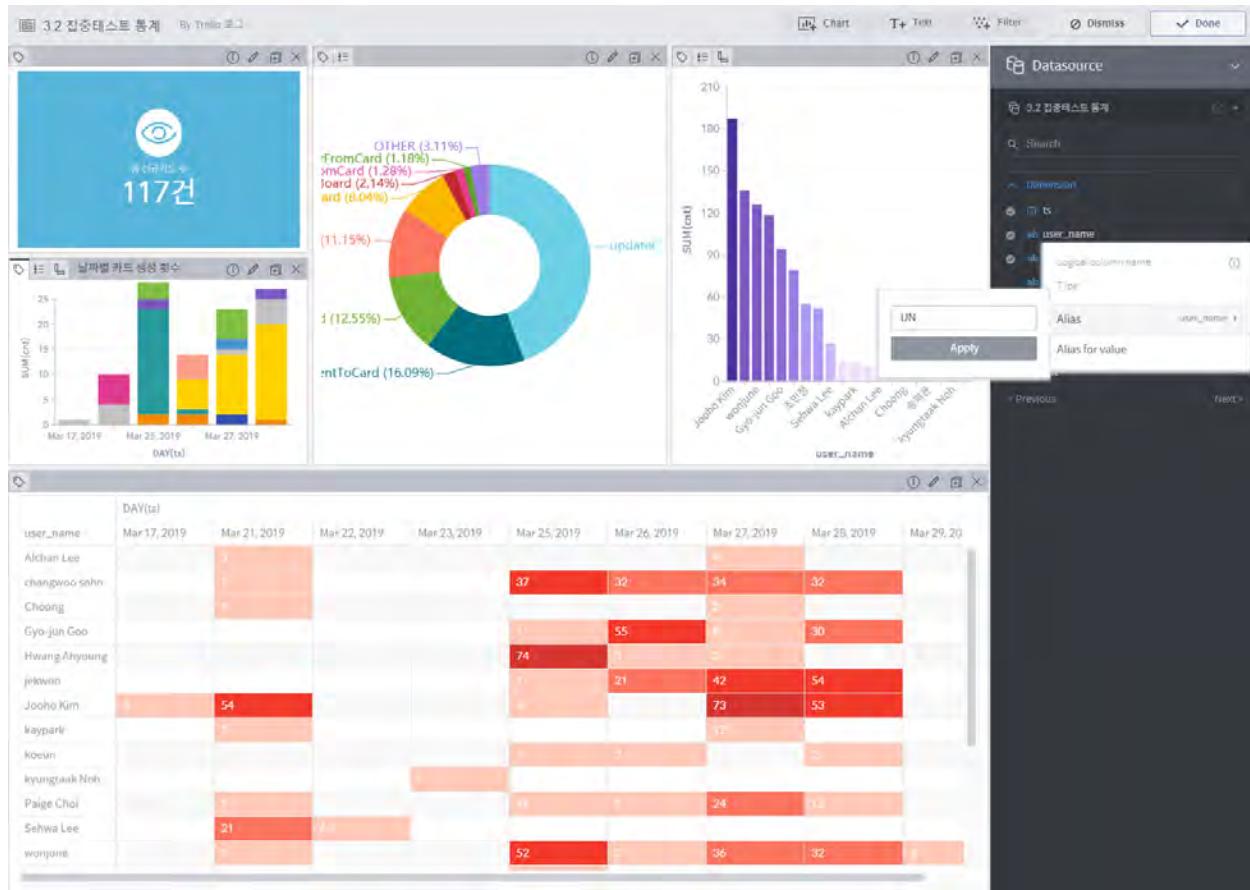
1. **Name:** Name of the current dashboard.
2. **Slide navigation:** Each circle represents a different dashboard in the workbook. For example, if you click the 4th circle, the 4th dashboard slide will be displayed with that circle highlighted.
3. **Auto slide show settings:** Select a duration for each slide and click **PLAY** to start an auto slide show.
4. **Exit:** Closes the presentation view and returns to the workbook/dashboard basic page.

5.2.5 Renaming columns

Hover the mouse over a column name on the data source panel in dashboard editing mode, and click the icon on the right to check the alias of the column.



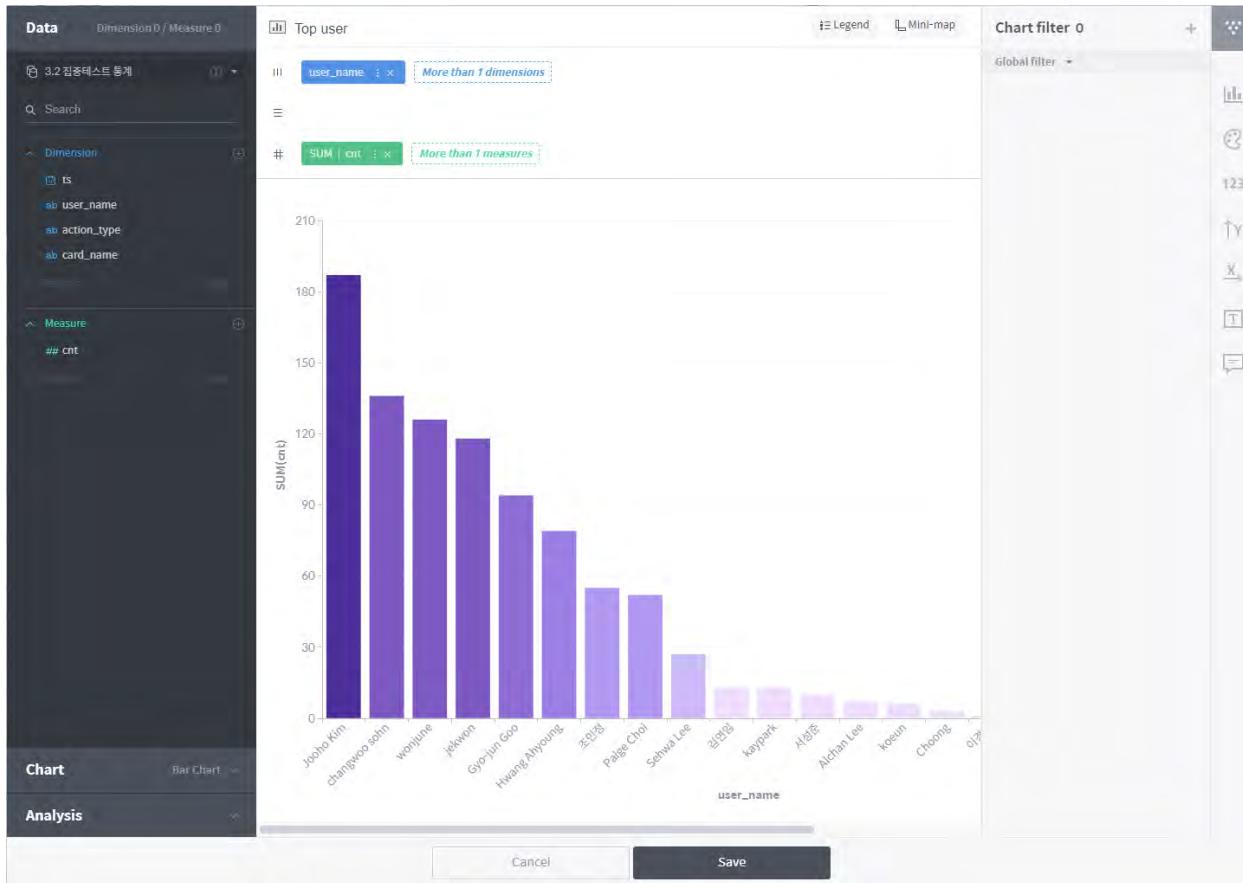
Hover the mouse over the alias to open a window where you can enter a new column name. After entering the name, click **Apply** to see the change applied.



5.3 Chart

Charts that analyze and visualize data are the main components of a dashboard. This section describes some concepts that you need to understand to create a chart for data analytics, as well as the elements that make up the chart configuration UI.

The chart home is divided into the following three sections:



- Column/chart selection section:** This section is so organized that you can create a chart step by step. You can either choose columns under the Data menu to have appropriate chart types suggested, or select a chart type under the Chart menu before choosing data columns. In addition, you can configure some analytics settings under the Analytics menu.
- Visualization section:** This section is composed of the shelves onto which columns are put and the visualization area where the chart is displayed. Once data and a chart type are selected in the column/chart selection section, the chart is drawn in this area.
- Option section:** Used to customize the appearance and display of the chart. Depending on the chart type, the option section may include the filter, palette, axis, numeric format, and chart format areas.

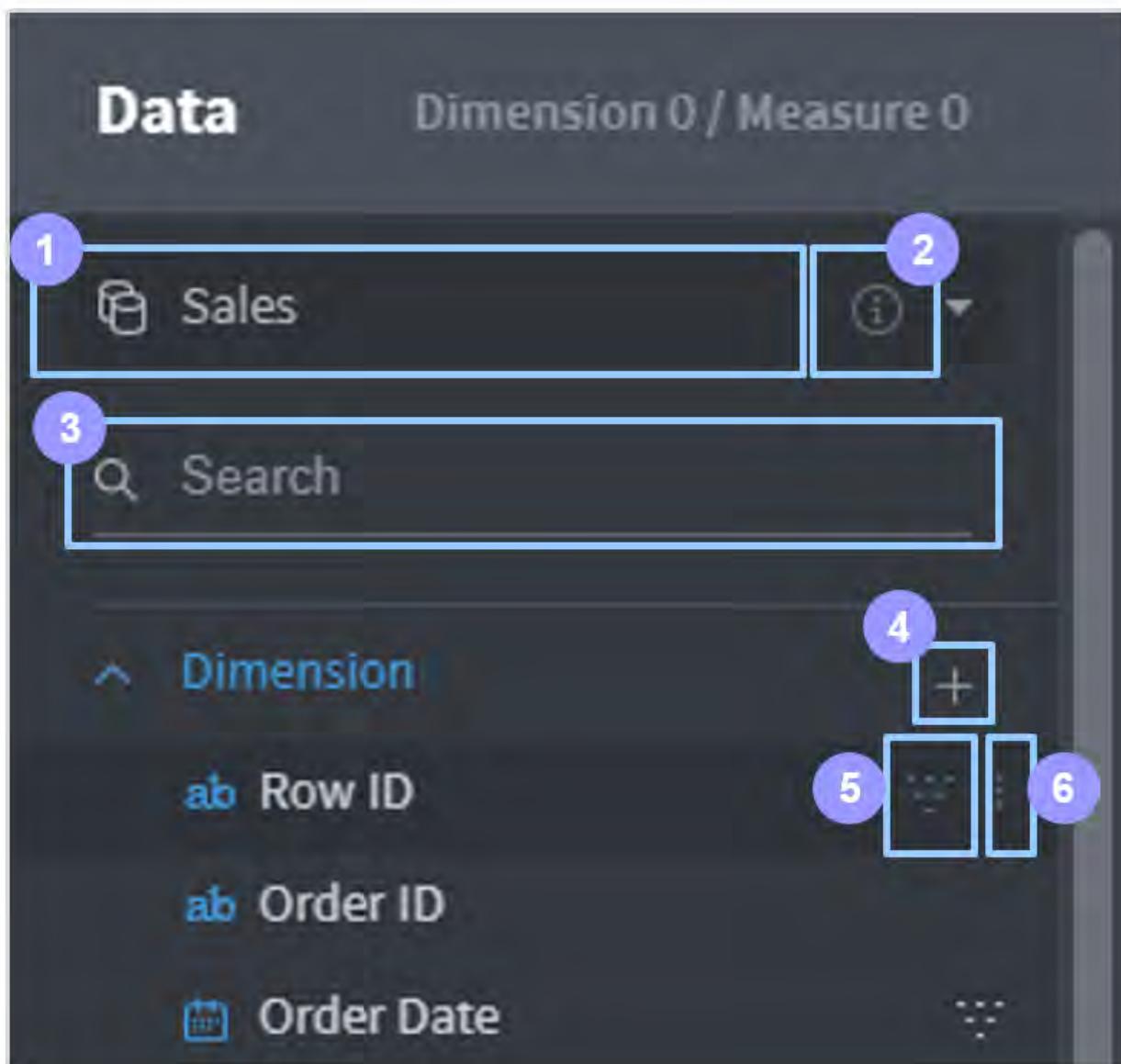
In the subsequent subsections, we will explain how to use this user interface to create and manage various types of charts.

5.3.1 Data column list

The columns listed in the data column list are categorized into “dimensions” and “measures.” For the concept of dimensions and measures, refer to “Dimensions” and “Measures”.

Structure of the data column list

In the data column list, you can view and edit information on connected data sources, as well as add or remove column filters easily.



1. Select/set data source: Allows you to select a data source or configure its associations

and joins.

2. **Data details:** Click on it to pop up a dialog box displaying information about the selected data source.
3. **Search by column name:** Searches the column list by name.
4. **Add custom column:** Click on it to open the dialog box to create a new column by combining/processing data source columns. Custom columns are commonly used throughout the dashboard.
5. **Apply/clear filter:** Hover the mouse over a column to display this button. Click on it to apply a chart filter to the column, and click again to clear the chart filter. For columns to which a filter is applied, the  icon is displayed regardless of the mouse position.
6. **More:** Hover the mouse over a column to display this button. It is used to check additional information on the column and set an alias.
 -  : Click on it to pop up a dialog box displaying a summary of the column and its data values.
 - **Logic column name:** Shows the logical name of the column.
 - **Type:** Shows the logical type of the column.
 - **Alias:** Sets a column alias. A regular column name can contain only alphanumeric characters and a limited number of special characters with no spaces allowed. Therefore, setting an alias may help to identify the column for convenient analytics work. Aliases are commonly used throughout the dashboard.
 - **Value alias:** You can also set an alias for each data value in the column. Aliases are commonly used throughout the dashboard.

Add a custom column

Click the + button on the data source column list to open a dialog box for adding a custom column. By applying various formulas to existing columns of the data source, you can create a new column that helps create your desired chart.

Custom column

Column name: DIMENSION_1

Coding box: `CAST([OrderDate], 'text')`

Validation check: There is no abnormality in the formula

Recommendation

Add column: 1 / 2

- OrderDate
- Category
- City
- Country
- CustomerName
- Discount
- OrderId
- PostalCode
- ProductName
- Profit
- Quantity
- Region
- Sales

Add formula

Search Formula: ALL

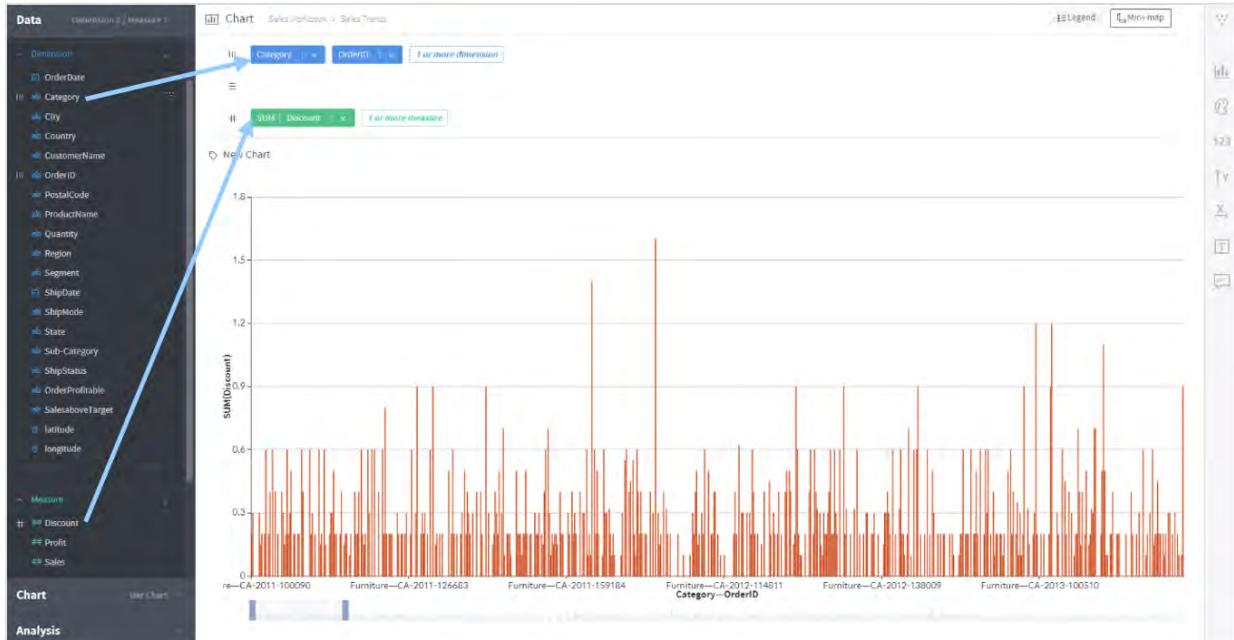
Function	Description
CAST	TYPE_CONVERT FIELD 자정한 타입으로 값을 변환하여 반환합니다.
CASE	
IN	
TYPE_CONVERT FUNCTION	
CAST	CAST(parameta.type) • parameta: 은(는) 변환할 대상이 되는 문자열 혹은 숫자입니다. • type: 은(는) 'DOUBLE', 'LONG', 'STRING', 'DATETIME' 중 하나로 변환할 타입입니다.
TIMESTAMP	
UNIX_TIMESTAMP	
TIME FUNCTION	
DATEDIFF	
NOW	CAST('100.123', 'DOUBLE') => 100.123
ETC FUNCTION	
IPV4_IN	CAST(TIMESTAMP('2016-01-01T12:00:00'))

- 1. Column name:** Fill in a name for the custom column.
- 2. Coding box:** Write a code for the custom column. Click a list from the column or formula list below to type your selection in this box automatically.
- 3. Add column:** Lists the columns of the data source. Click a column in the list to automatically type your selection in the coding box.
- 4. Add formula:** Lists the formulas supported by Metatron. Click a formula in the list to type your selection in the coding box automatically, with the text cursor relocated to where a parameter needs to be inserted. For details on each formula's purpose, use, and examples, see the help box on the right.

5.3.2 Draw a chart (pivoting)

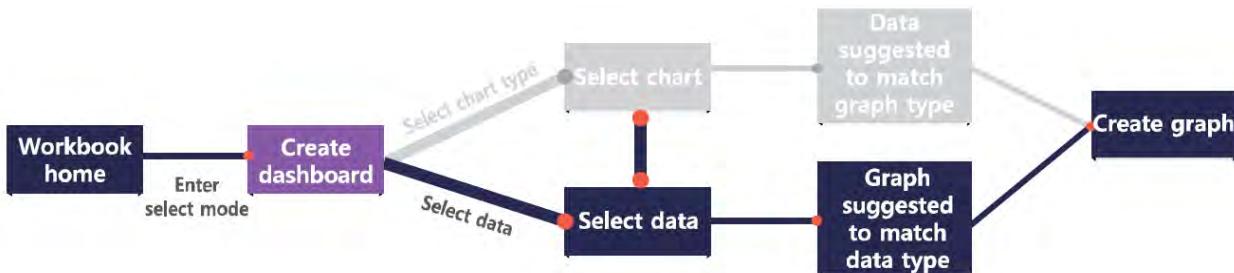
What is pivoting

Pivoting is a process of grouping the given table by specific columns, thereby helping the analyst view particular aspects of the source data in a graphic or tabular chart. This process includes selecting columns that contain meaningful data and placing them on the column/row/cross shelves.



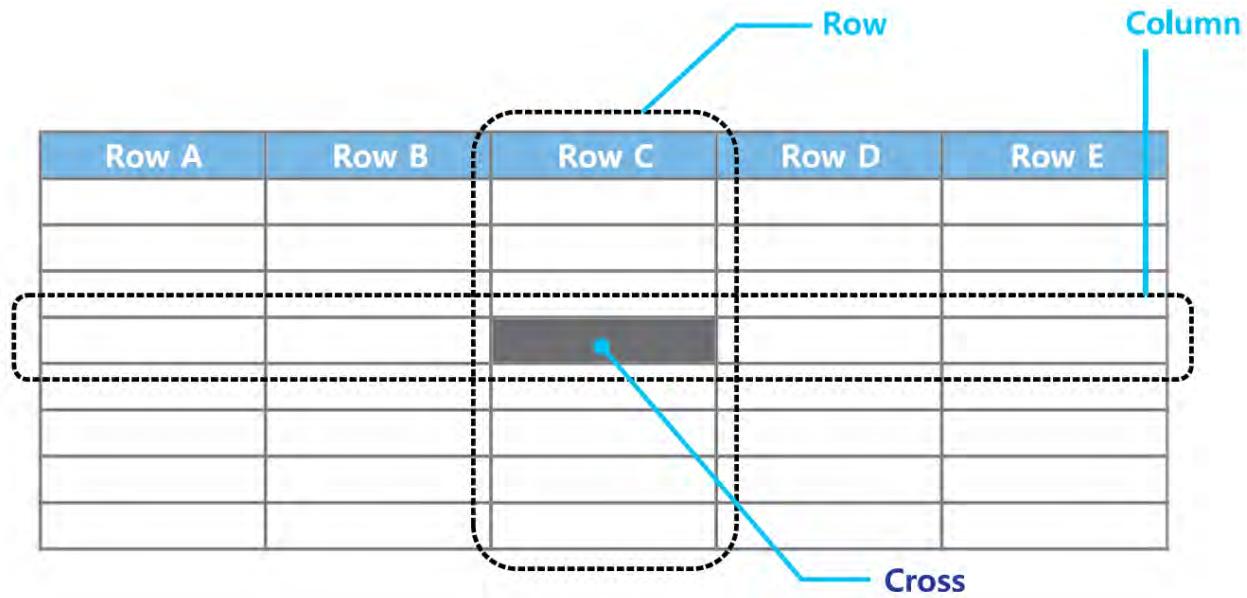
In the example shown above, two dimension columns are placed on the column shelf and one measure column is placed on the cross shelf. The chart displays data resulting from the columns placed on the shelves in this way.

Mandatory/recommended column types for each shelf vary depending on the chart type. Selecting a chart type before placing columns on a shelf shows the necessary column types for each shelf.

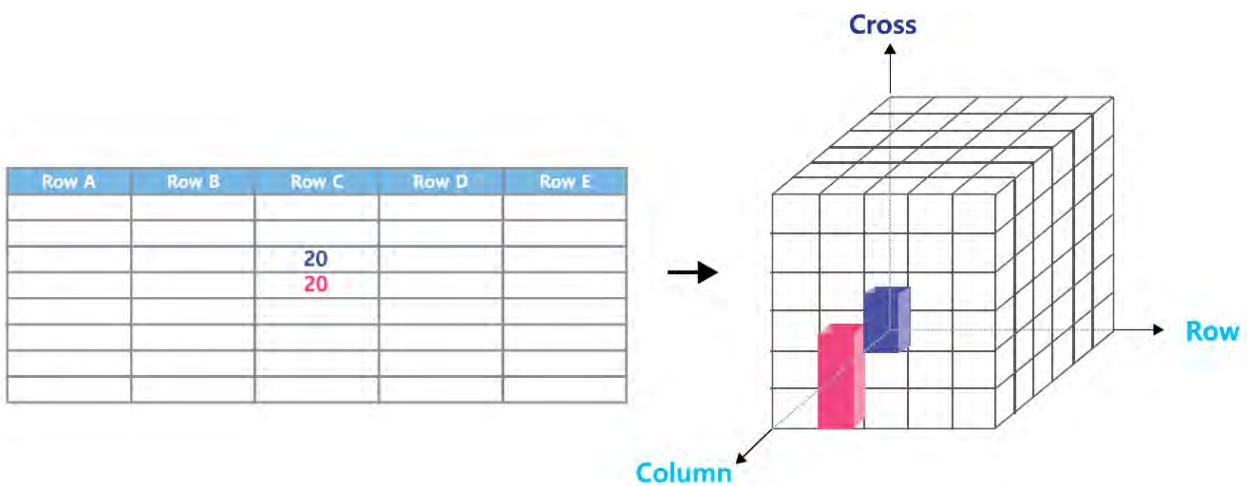


Column/row/cross shelves

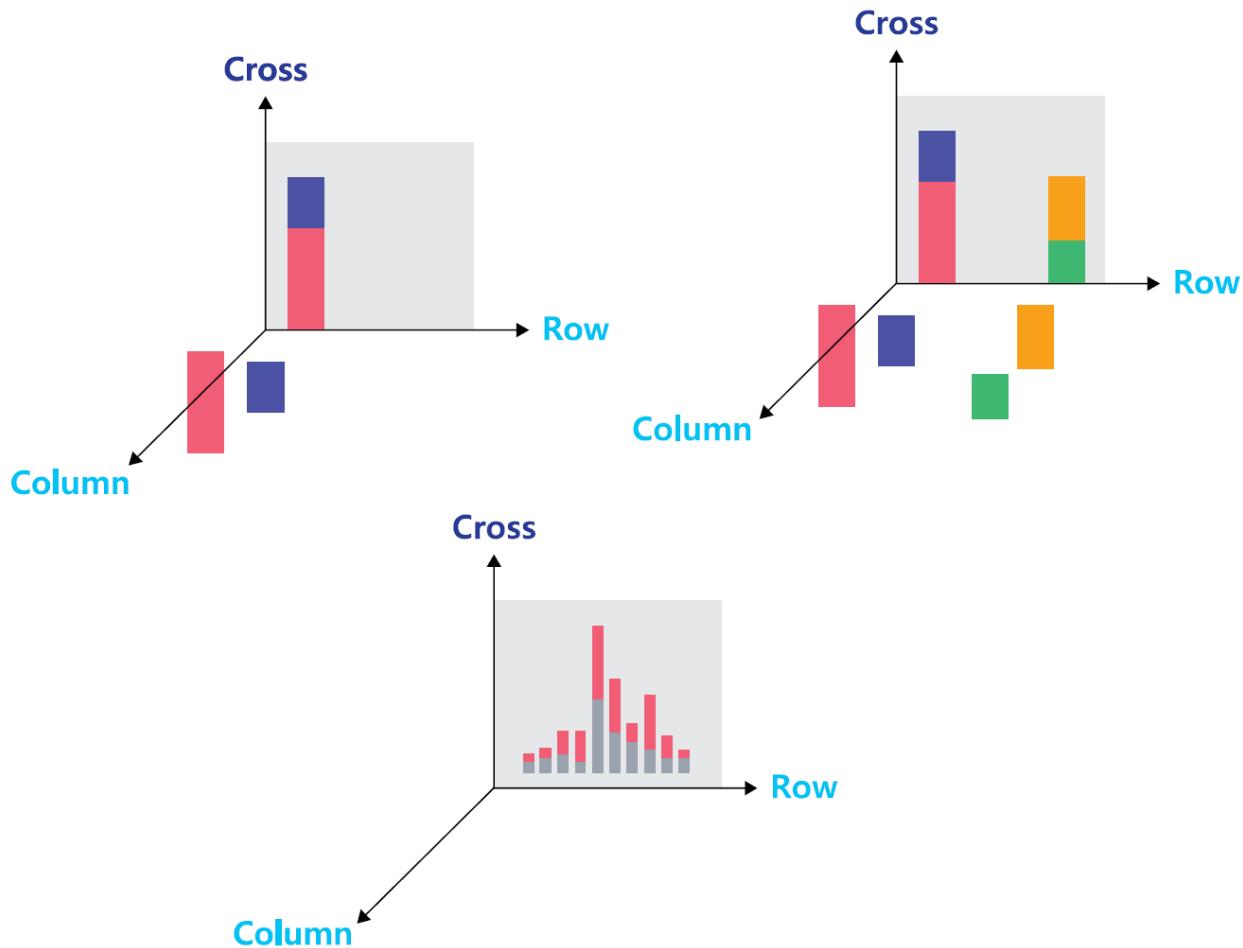
Think of the structure of Excel to understand what column/row/cross shelves work for. As shown below, the crossing of each column and row cross contains a value.



Whereas Excel shows data in a two-dimensional grid composed of columns, rows and crosses, Metatron is an OLAP data discovery tool capable of multidimensional data representation. In the following Metatron chart, the column, row, and crossing axes form a three-dimensional cube.

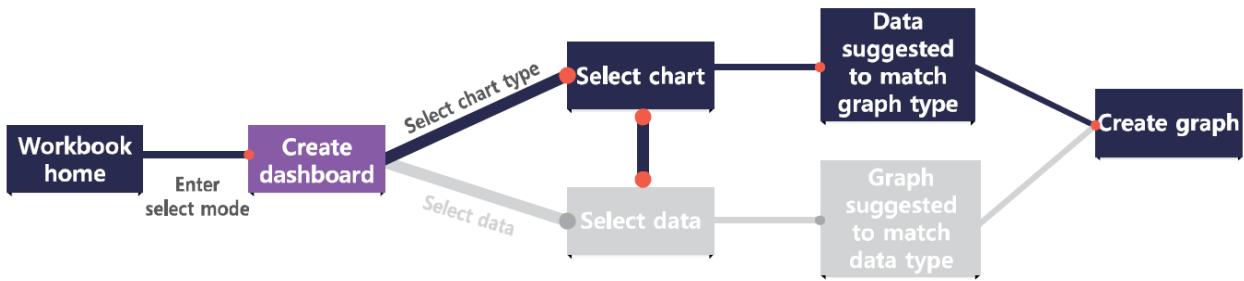


If the values of an Excel grid are displayed in a three-dimensional chart, each crossing value will be represented by a bar. However, Metatron needs to display such a chart two-dimensionally; for this, bars either in the same column or in the same row get stacked at one point while remaining distinctive from one another. The resulting two-dimensional chart is shown in the gray area of the chart below.



5.3.3 Select a chart type

Metatron Discovery provides about 20 types of charts. If you place columns on shelves before selecting a chart, suitable charts are highlighted in purple.



The table below summarizes conditions to create, uses, and examples for each chart.

Chart name/icon	Conditions to create	Characteristics	Uses	Examples
	Column: 1 or more dimensions / Cross: 1 or more measures	Compares the value of each item.	Used to compare groups or view trends over time. Very effective when the trend is significantly fluctuating.	Comparison between products regarding their sales and profits
	Column or row: 1 or more dimensions / Cross: 1 or more measures	Displays the values of crossings between two dimensions as text.	Used to view measure values aggregated by certain criteria. Useful to check exact values rather than a visualization of them.	Sales details by year
	Column: 1 or more dimensions / Cross: 1 or more measures	Displays data changes over time.	Used to view trends over time. If changes are moderate, a line chart is more effective than a bar chart.	Monthly sales trend
	Column: 1 measure / Row: 1 measure / Cross: 1 or more dimensions	Displays relations between items.	Used to define relations between two parameters.	Relations between product sales and profits
	Column or row: 1 or more dimensions / Cross: 1 or more dimensions	Displays the values of crossings between two dimensions represented by colors and sizes.	Used to provide an intuitive view of relations between two dimensions represented by colors and sizes. Similar to a table chart, but more of a visual type.	Sales of each product by region
168	1 or more measures	dimensions in colors and sizes at different points	colors and sizes. Similar to a table chart, but more of a visual type.	Chapter 5. Workbook

5.3.4 Chart style attributes

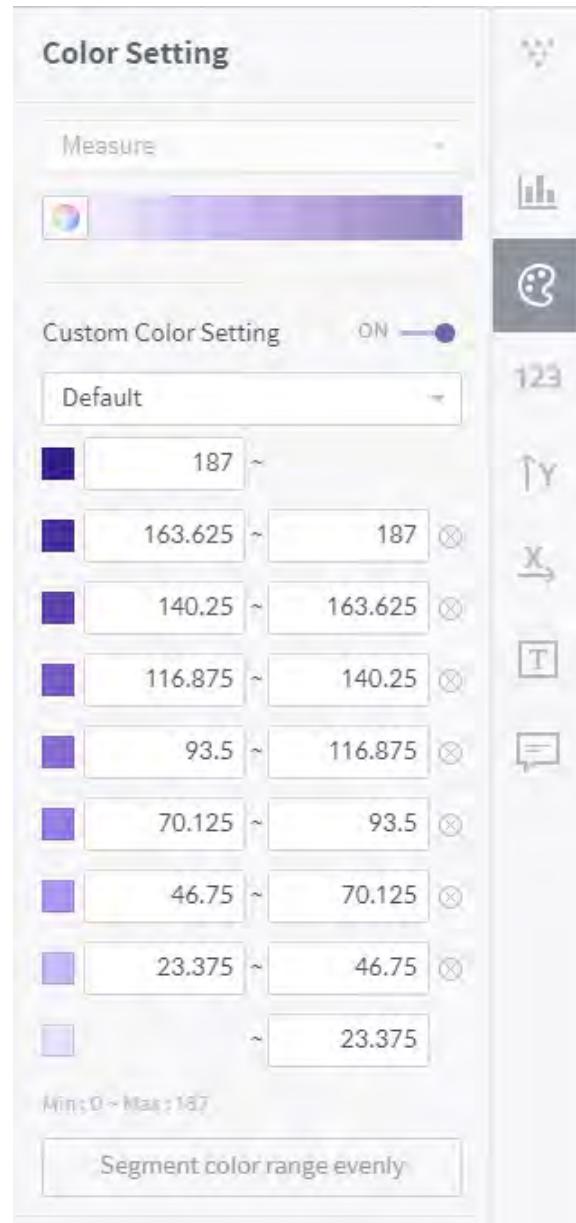
Once data is pivoted, an options menu is shown on the right of the screen to allow you to set the chart style. The composition of the menu varies with chart type. This section describes the settings used universally by all chart types and the “Common Setting” items for each chart type.

Chart style settings menu

This section describes how to configure the settings of the chart style settings menu. Note that not all the settings are shown for every chart type.

Color setting

Defines various colors used in the chart.

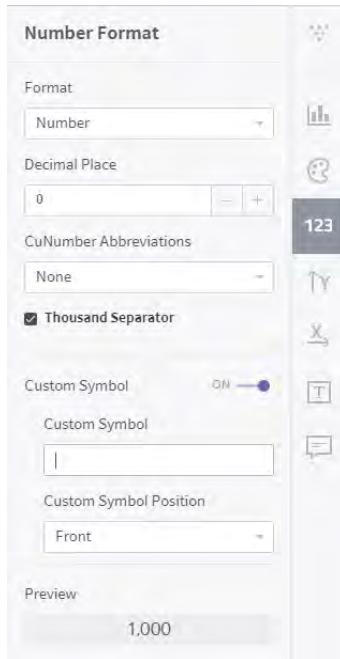


1. **Graph color setting:** Set criteria to classify data on the chart by color, and select a coloring theme.
 - **Series:** Colors data elements differently with measures.
 - **Dimension:** Colors data elements differently with dimensions.
 - **Measure:** Colors elements differently with the size of each aggregate of measure values.
2. **Setting color range:** This setting is displayed when **Measure** is selected as the criterion to classify data by color. Set “ON” to set colors differently with each range of measure

values. The measure data to be colored can be subdivided into as many ranges as you want, starting with the lowest one. To add a new range, adjust the upper limit of the highest range and click **Add new range**.

Number format

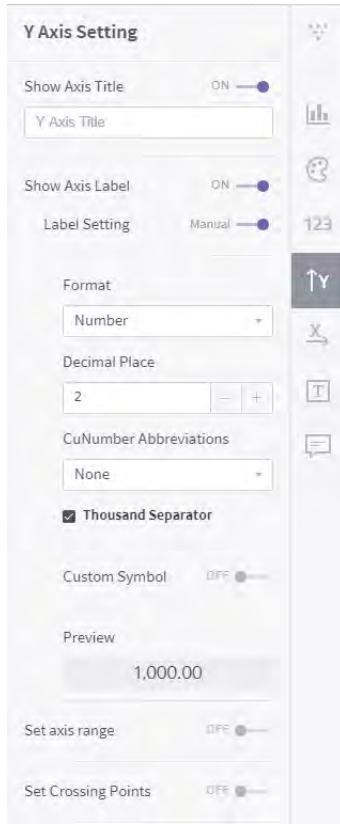
Defines how to display numerical text data on the chart graph. To use this function, turn on Show Axis Label in the Data Label Settings Menu.



1. **Format:** Select a display format for numeric values from among number, currency, percent, and exponent.
2. **Decimal place:** Set how many digits to display after the decimal point.
3. **Number abbreviations:** You can use K (thousands), M (millions), or B (billions) as an abbreviation for a large numeric value. Select **Automation** to automatically set the most proper symbol in accordance with the number of digits.
4. **Thousands separator:** Select whether to add thousands separators when displaying numeric data values.
5. **Customer symbol:** Insert a custom text before/after numeric data values.
6. **Preview:** Displays the result of the defined number format.

Y-axis setting (when chart type is vertical)

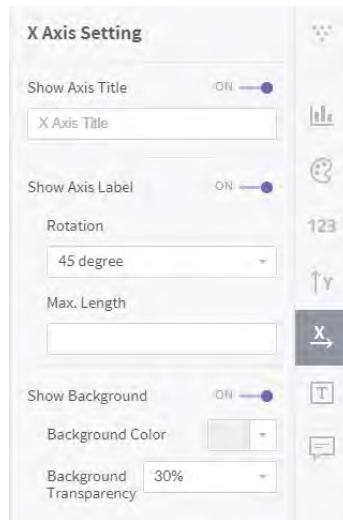
If you set the chart direction **Horizontal** in the Common Setting area, the settings are exchanged between X-axis and Y-axis.



1. **Show axis title:** Used to set a title for the Y-axis of the chart. Disabling this function hides the title of the Y-axis.
2. **Show axis label:** Select whether or not to show the data labels on the Y-axis of the chart. Disabling this function hides the data labels on the Y-axis.
 - **Label setting:** Set the numeric format of the data labels on the Y-axis. Set automatic to import the settings of **Format** or manual to set specific format for the data labels on the Y-axis.

X-axis setting (when chart type is vertical)

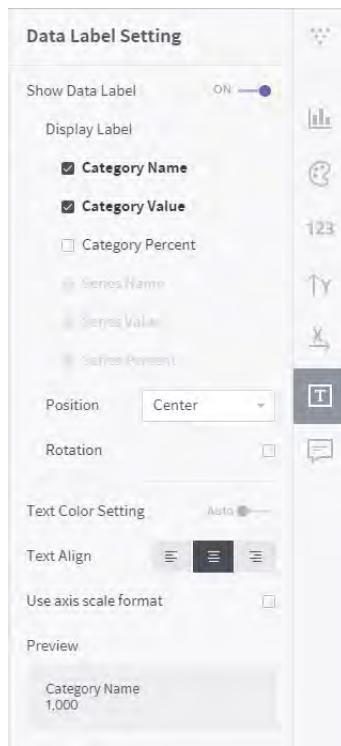
Defines how to display the X-axis of the chart. If you set the chart direction **Horizontal** in the Common Setting area, the settings are exchanged between X-axis and Y-axis.



1. **Show axis title:** Used to set a title for the X-axis of the chart. Disabling this function hides the title of the X-axis.
2. **Show axis label:** Select whether or not to show the data labels on the X-axis of the chart. Disabling this function hides the data labels on the X-axis.
 - **Rotation:** Select an angle for the data labels on the X-axis from among 0, 45, and 90 degrees.

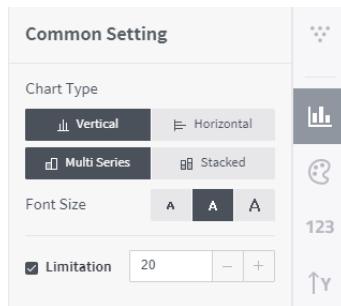
Data label setting

Selects whether to display the data values on the chart graph.



Common settings for each chart type

This section describes how to style the six most popular chart types (bar chart, table, line chart, scatter chart, heatmap, and pie chart).



Bar chart

This type of chart presents data values in each category of a dimension column with rectangular bars.



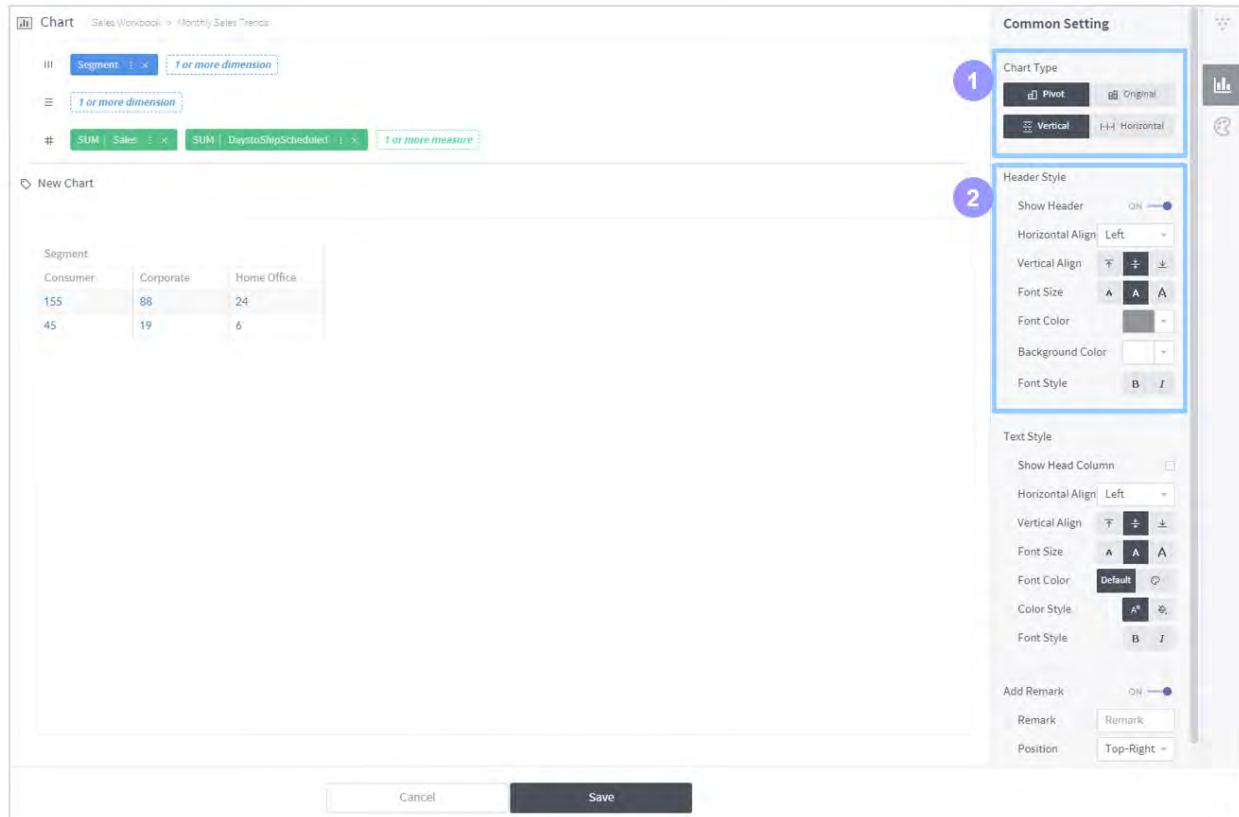
1. Chart type

- **Vertical:** Displays data values as vertical bars with the dimension axis set vertical.
- **Horizontal:** Displays data values as horizontal bars with the dimension axis set horizontal.
- **Parallel:** If more than one measure are selected, different bars representing those measures are displayed in parallel.
- **Stacked:** If more than one measure are selected, different bars representing those measures are stacked at one position.

2. Limitation: Set how many columns to display on the chart.

Table

A table block is formed based on the categories into which the dimension columns on the column/row shelves are grouped; accordingly, the values of the measure columns on the cross shelf are displayed as text in the crossings.



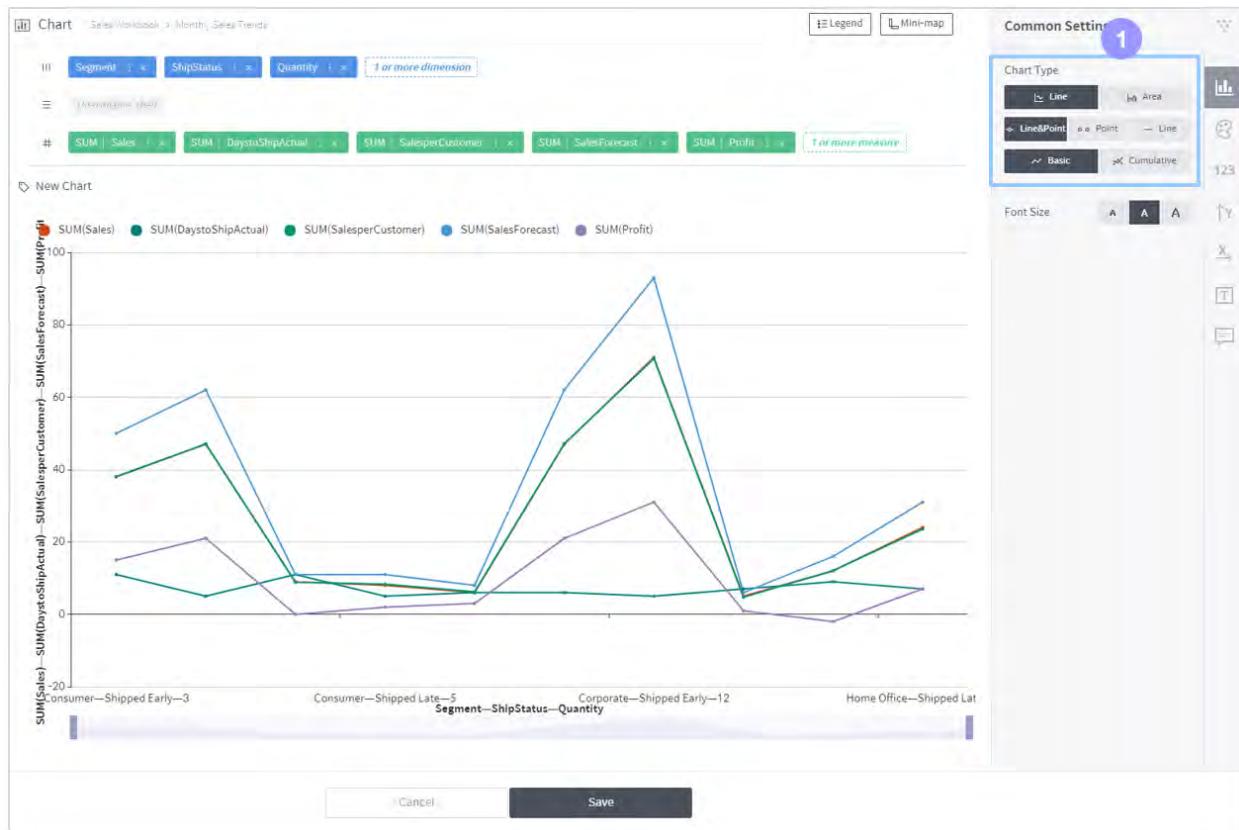
1. Chart type

- **Pivot:** Aggregates (SUM, MIN, MAX, etc) measure values for each pair of column and row dimensions into a different cell.
- **Original:** Displays all original measure values as unaggregated together with the selected dimensions.
- **Vertical:** Displays measure values vertically in the table. This cannot be used when “Original” is selected for displaying the table.
- **Horizontal:** Displays the table horizontally when “Pivot” is selected for displaying the table. Displays measure values horizontally in the table.

2. **Show head column:** Set horizontal and vertical text alignment in the column headers. When “Original” is selected, the column headers are necessarily shown. When “Pivot” is selected, you may optionally hide the column headers.

Line chart

This type of chart presents data values in each category of a dimension column with points. Adjacent data points are connected with each other. This type of chart is used to view trends.

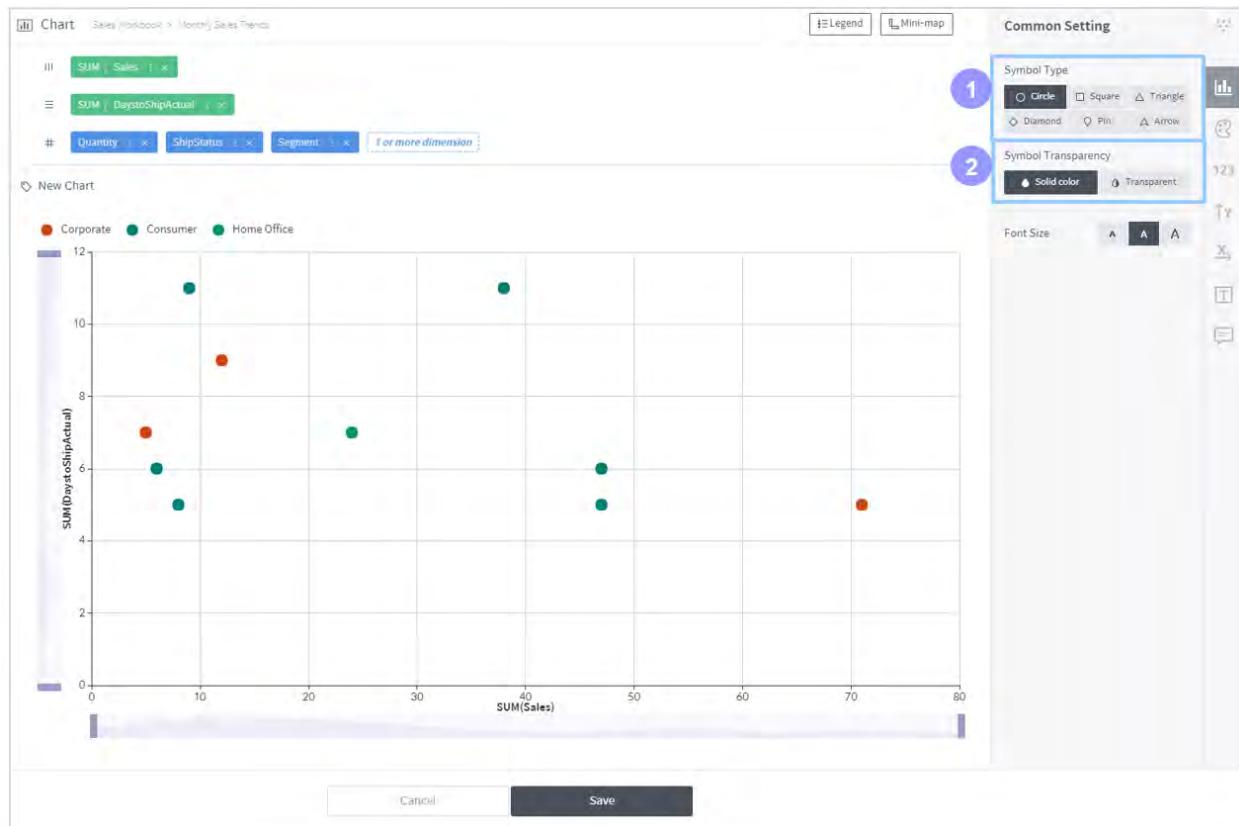


1. Chart type

- **Line type:** Displays the chart graph by drawing lines between points that represent measure value aggregates.
- **Area type:** Colors the area formed by the connecting lines.
- **Line & point:** Shows both the data points and connecting lines.
- **Point:** Shows the data points only.
- **Line:** Shows the connecting lines only.
- **Basic:** Displays each aggregate as it is on the chart.
- **Cumulative:** Displays cumulative aggregates on the chart.

Scatter chart

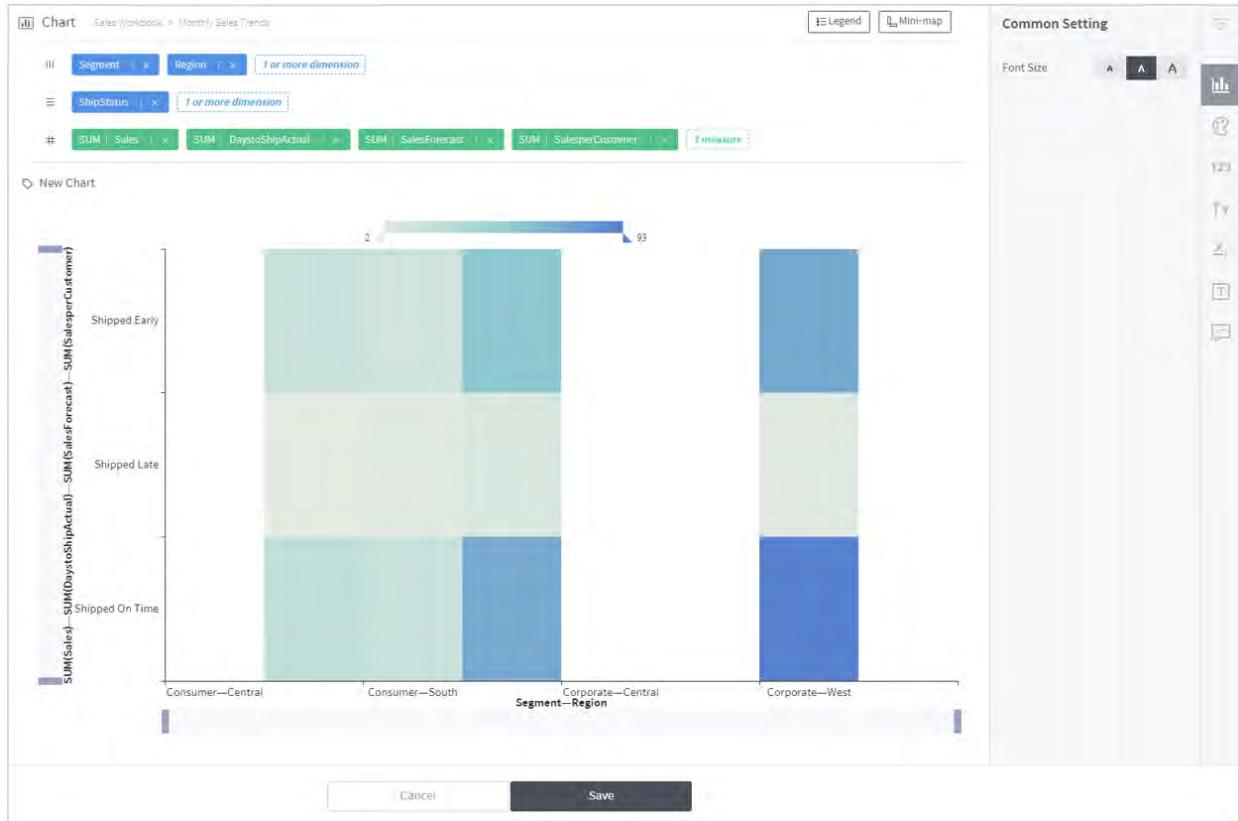
This type of chart presents data values in each category of a dimension column with defined symbols.



- 1. Symbol type:** Set the shape of the symbol to be shown on the chart.
- 2. Symbol transparency:** Set the transparency of the symbol to be shown on the chart. You can set colors either solid or transparent.

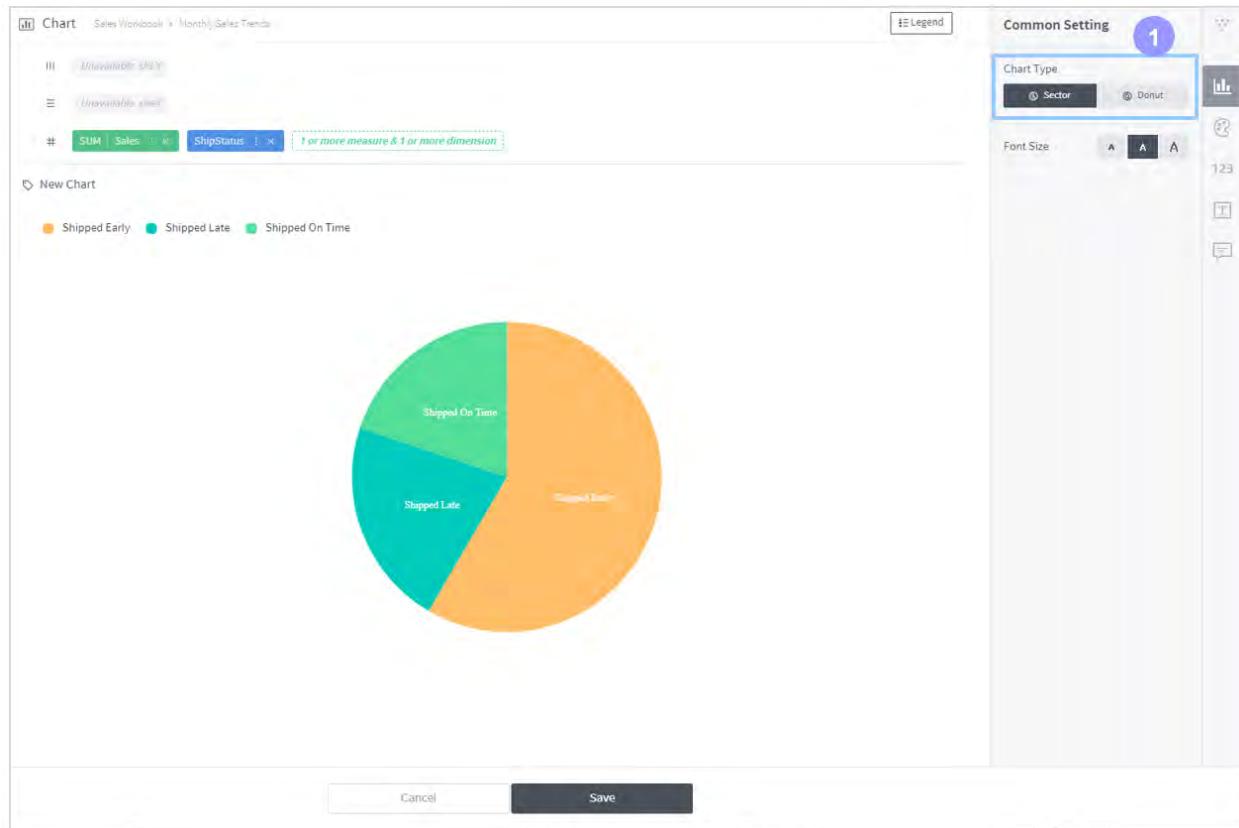
Heatmap

This type of chart displays values aggregated from the measure column placed on the cross shelf by using colors. For a larger aggregated value, a darker color is applied. The heatmap type does not provide any common settings.



Pie chart

This type of chart visualizes the proportion of each category of the dimension column.

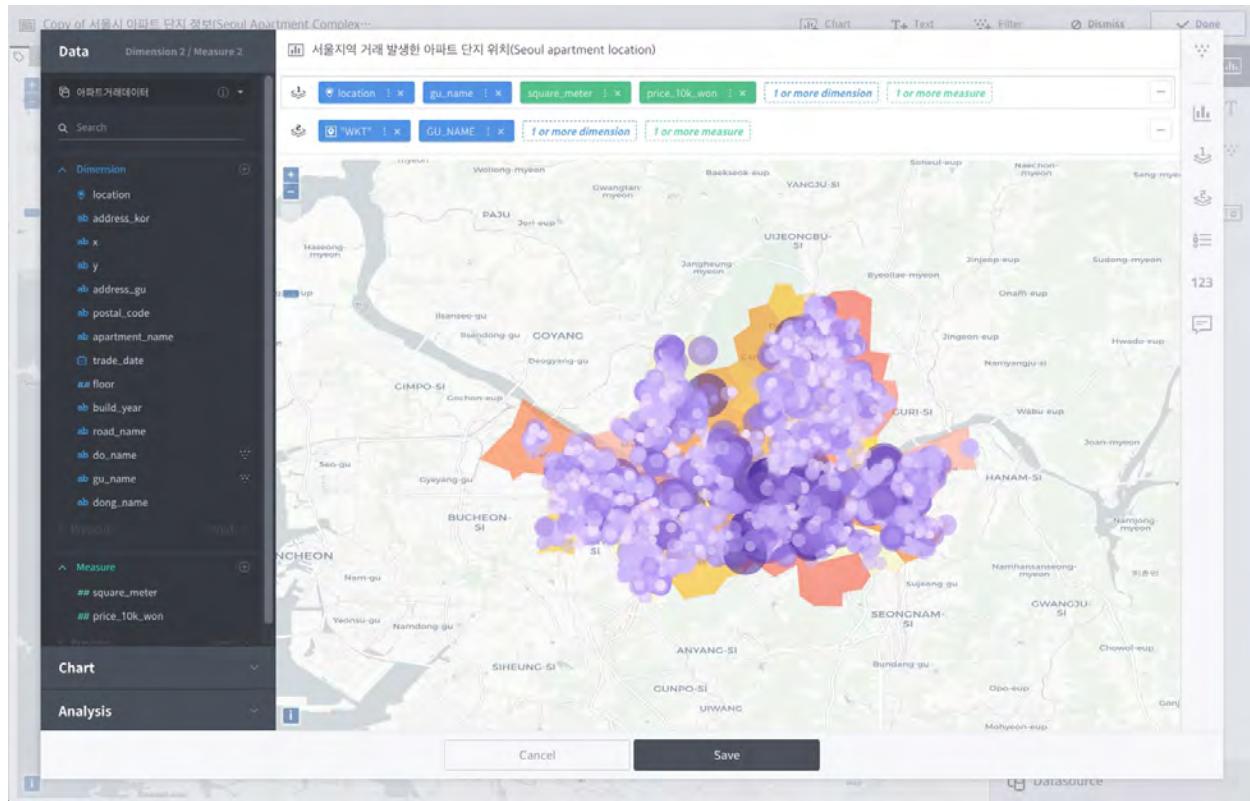


1. Chart type

- **Sector:** Displays a pie-shaped chart.
- **Donut:** Displays a donut-shaped chart.

5.3.5 Map view and spatial operations

Metatron Discovery, from version 3.1.0 and up, offers a **map view** function for visualizations of location data. Creating a chart in map view involves different conditions compared to other chart types.



- At least one **location** dimension is required.
- Data is placed on **map layer shelves** instead of the row/column/intersection shelves.
- **Style properties** are set for each layer.
- **Spatial operations** are provided.

Location dimensions

To use map view, dimension columns of WKT geometry types such as Point, LineString, and Polygon must be placed on the layer shelf. There are largely three types of location data.

- **Point**: This is a 2D coordinate geometry type comprised of x and y values. Similar to GPS data, a point has a latitude and longitude.
- **Line**: This is a geometry type with line coordinates. WKT representations of LineString and Multi-LineString are supported.

- **Polygon:** This is a geometry type with shape coordinates. WKT representations of Polygon and MultiPolygon are supported.

Geometry primitives (2D)

Type	Examples	
Point		POINT (30 10)
LineString		LINESTRING (30 10, 10 30, 40 40)
Polygon		POLYGON ((30 10, 40 40, 20 40, 10 20, 30 10))
		POLYGON ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))

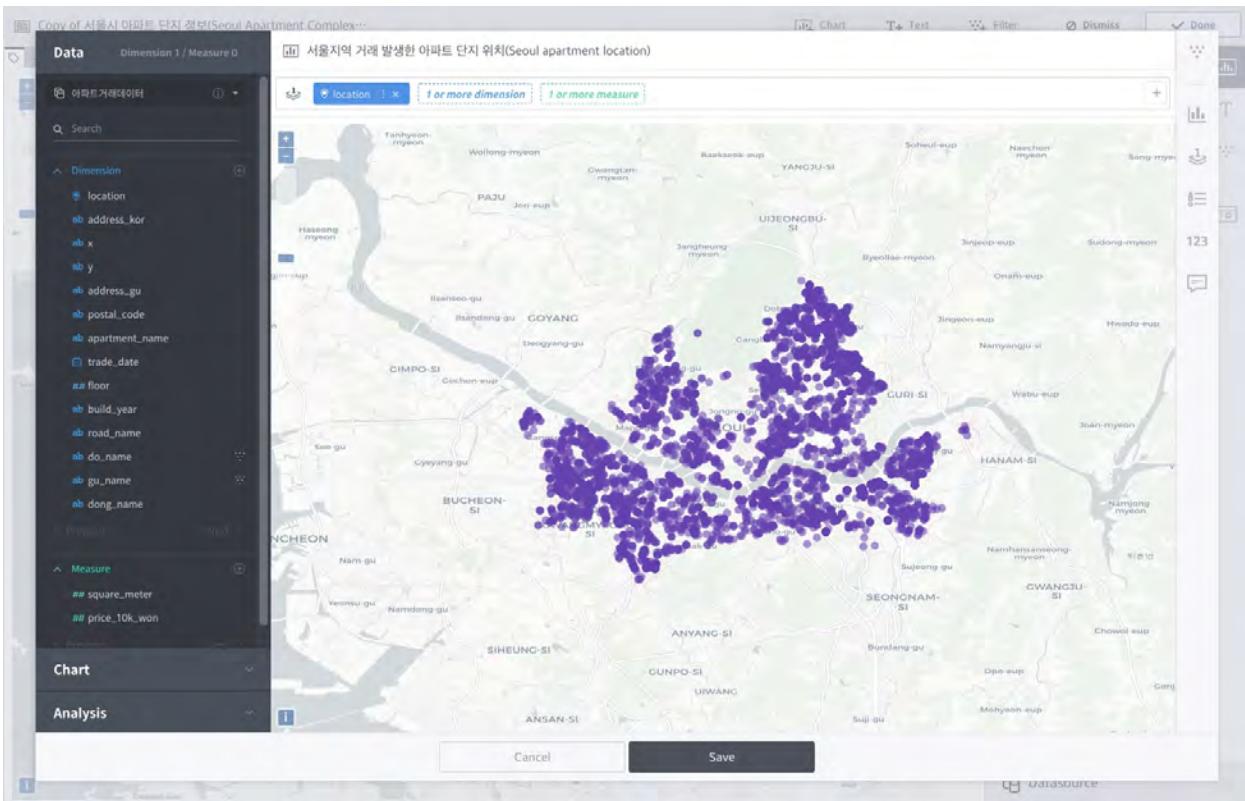
Multipart geometries (2D)

Type	Examples	
MultiPoint		MULTIPOINT ((10 40), (40 30), (20 20), (30 10))
		MULTIPOINT (10 40, 40 30, 20 20, 30 10)
MultiLineString		MULTILINESTRING ((10 10, 20 20, 10 40), (40 40, 30 30, 40 20, 30 10))
		MULTIPOLYGON (((30 20, 45 40, 10 40, 30 20), ((15 5, 40 10, 10 20, 5 10, 15 5))), ((40 40, 20 45, 45 30, 40 40), (20 35, 10 30, 10 10, 30 5, 45 20, 20 35), (30 20, 20 15, 20 25, 30 20)))
GeometryCollection		GEOMETRYCOLLECTION (POINT (40 10), LINESTRING (10 10, 20 20, 10 40), POLYGON ((40 40, 20 45, 45 30, 40 40)))

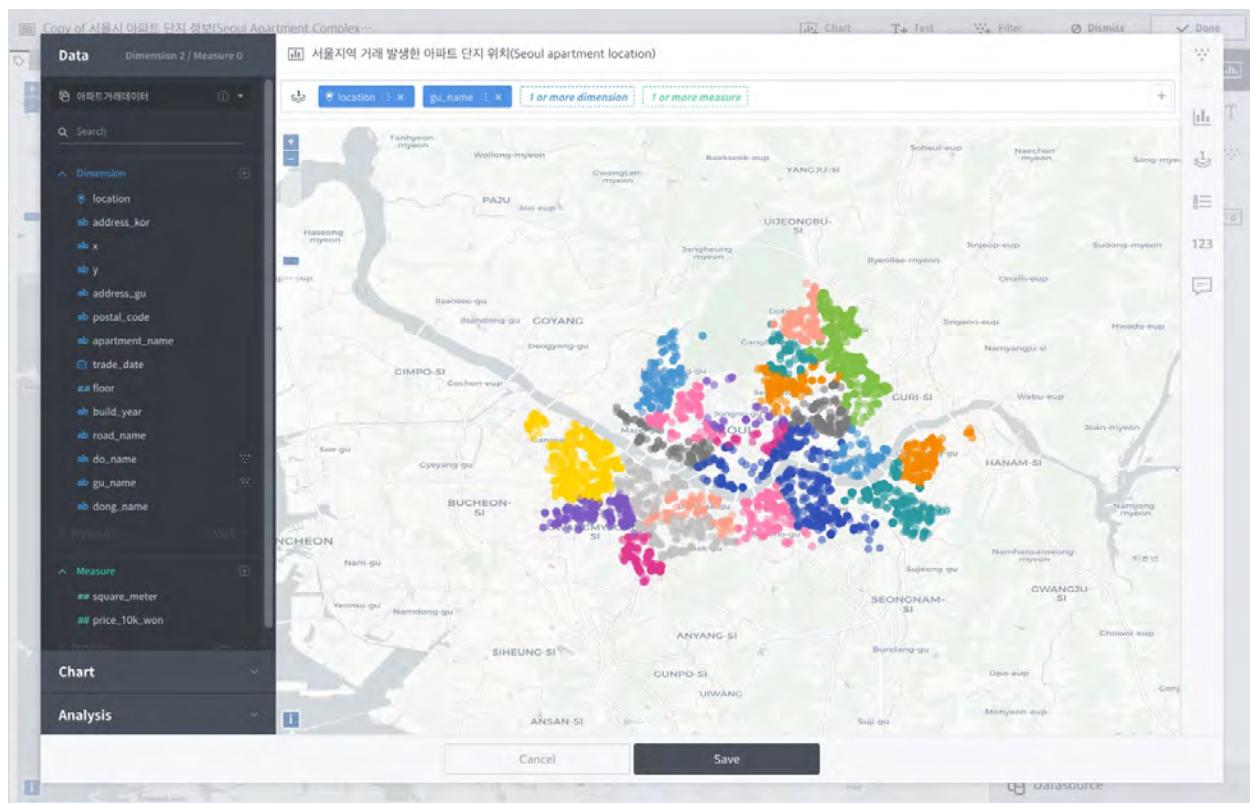
Map layer shelves



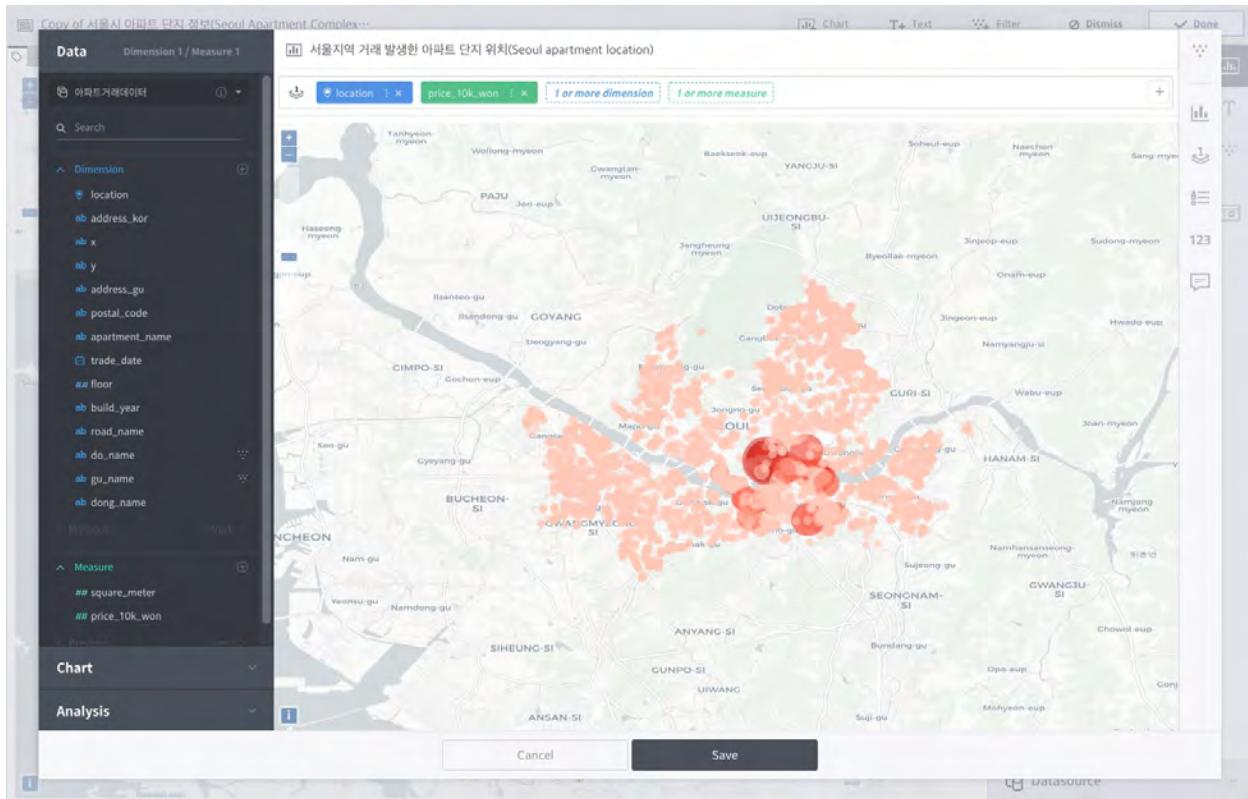
Map view uses map layer shelves instead of the row/column/intersection shelves that are used by other chart types. A map layer shelf requires at least one location dimension.



When a string dimension is placed on a map layer shelf, data points are colored based on its elements; when the mouse is over a data point, the corresponding string is displayed in the data tooltip.

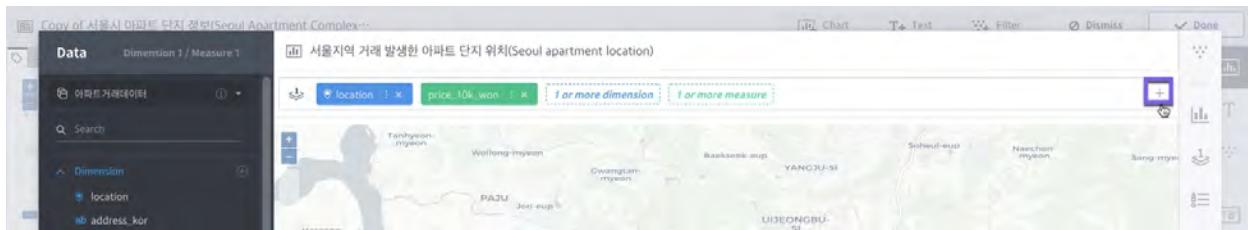


또한 측정값을 레이어 선반에 배치하면 측정값으로 색상을 분류하고 동시에 해당 측정값을 기준으로 포인트 크기를 다르게 표현합니다. 차원값과 마찬가지로 툴팁에 해당 측정값이 표기됩니다.



Add layer shelf

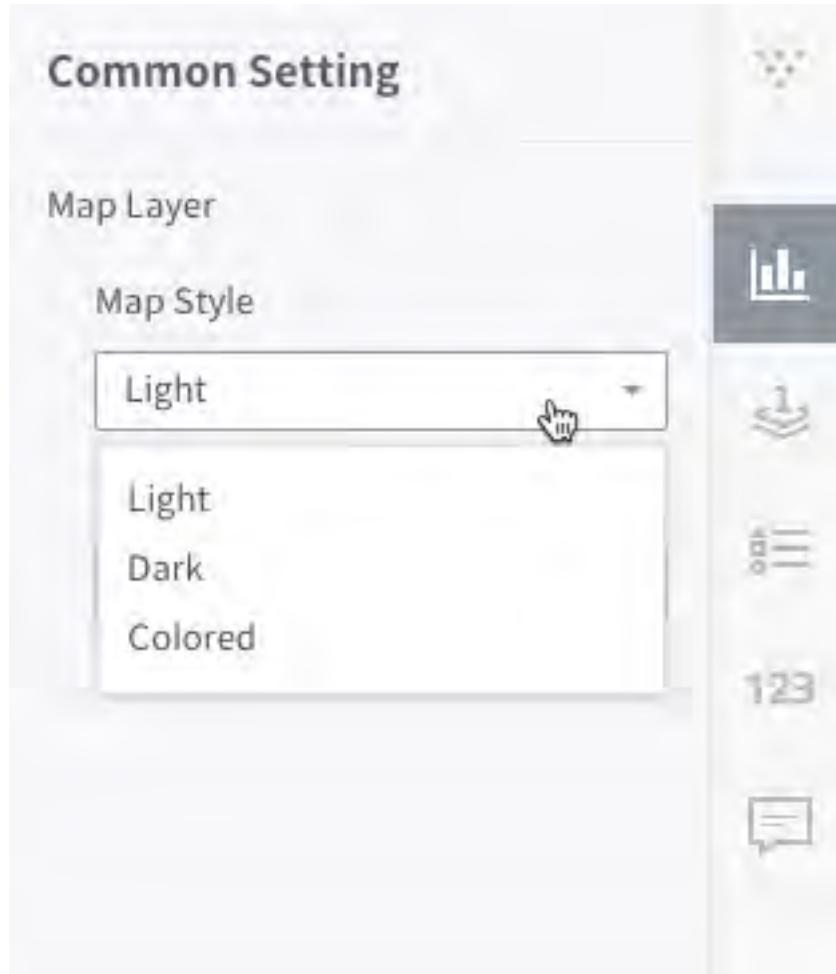
Click the + button on the right of a layer shelf to add another layer on top of the first layer. Each layer must use a different data source, and columns of only one data source are allowed to be placed per layer. Currently, up to two layer shelves are supported.



Style properties of map view layer

Common setting

지도 레이어에서 기본 지도를 표현하는 맵 스타일의 유형을 선택할 수 있습니다. OpenStreetMap을 활용하여 세 가지의 맵 스타일을 기본적으로 제공하고 있습니다.

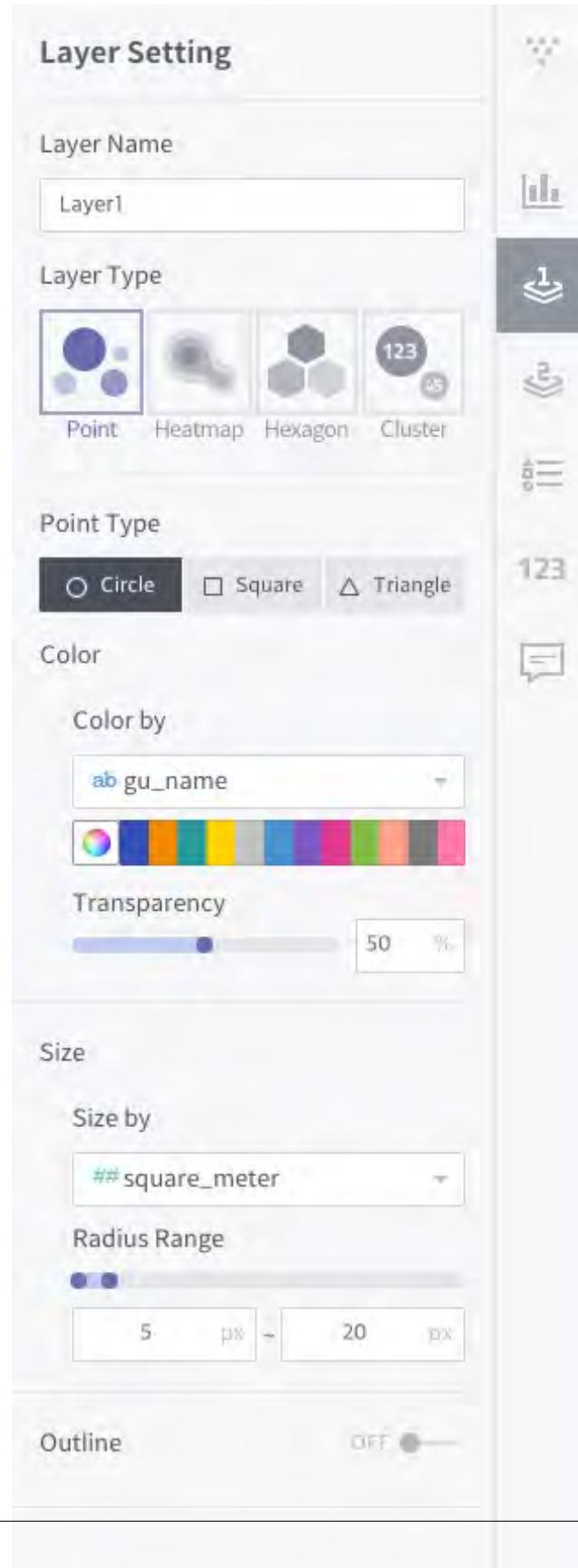


- Open Street Map Light (Default)
- Open Street Map Dark
- Open Street Map Colored

Layer settings

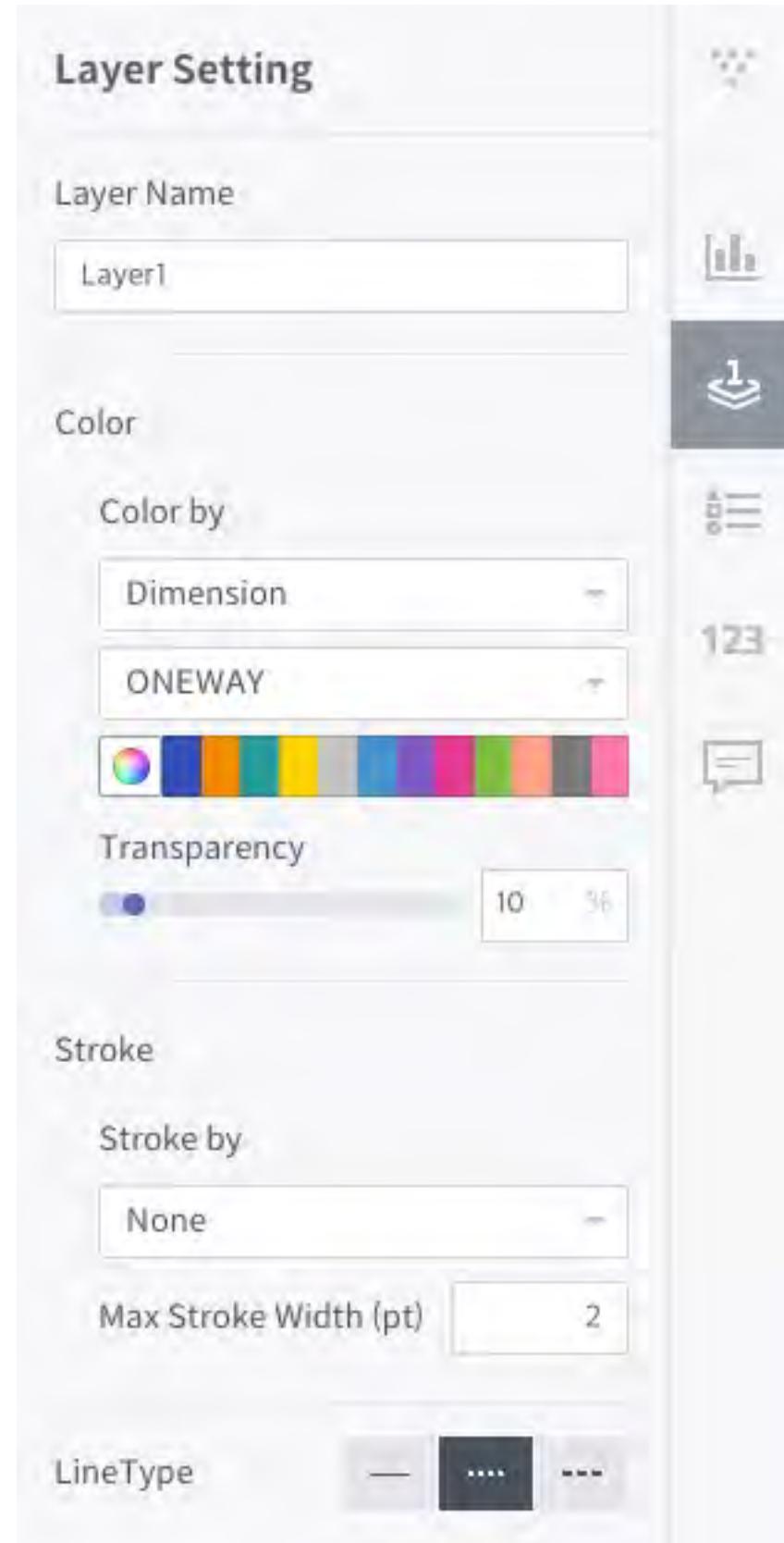
Sets how to express layers. When a layer shelf is added, separate setting menus are created for the first and second layers.

Layer properties of point type



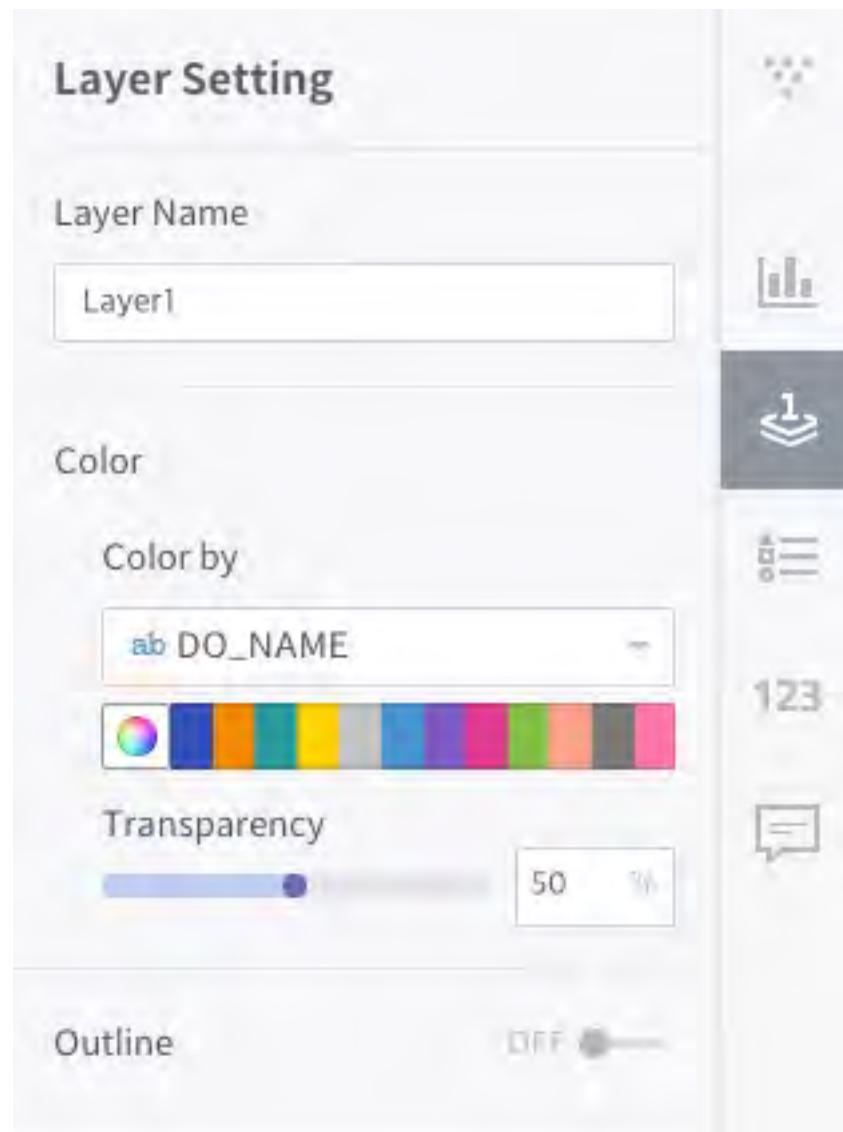
1. **Layer Name:** Set a name of the layer for legend and tooltip settings in the map view.
2. **Layer Type:** Data points can be displayed on the map as Point, Heatmap, Hexagon, or Cluster. The point type is selected by default.
3. **Point Type:** With Point selected as the layer type, you can choose the shape of data points from among Circle, Square, and Triangle. Circle is selected by default. The shapes are displayed on the map when cluster use is set to Off.
4. **Color:** Data points can be distinguished by color based on a string dimension or a measure on the layer shelf. A color can be picked from the palette if color standards are not available. The transparency can be set as a % value.
5. **Size:** If the layer type is Point, data points can be distinguished by size based on a measure on the layer shelf.
6. **Outline:** When set to On, an outline is drawn for each data point. The default is Off, and the color and thickness are customizable.
7. **Cluster Distance:** With Cluster selected as the layer type, you can set the cluster distance as a % value. The use of clusters is recommended to optimize browser performance when working with a large number of data points.
8. **Blur:** With Heatmap selected as the layer type, you can adjust the blur effect on the heat map. The default is 20%.
9. **Radius:** If the layer type is Heatmap or Hexagon, the display radius can be adjusted in the range of 1 to 100.

Layer properties of line type



1. **Layer Name:** Set a name of the layer for legend and tooltip settings in the map view.
2. **Color:** Data points can be distinguished by color based on a string dimension or a measure on the layer shelf. A color can be picked from the palette if color standards are not available. The transparency can be set as a % value.
3. **Thickness:** Set the line thickness.
4. **Line type:** Choose among a solid line, dotted line, and dashed line. The default is a solid line.

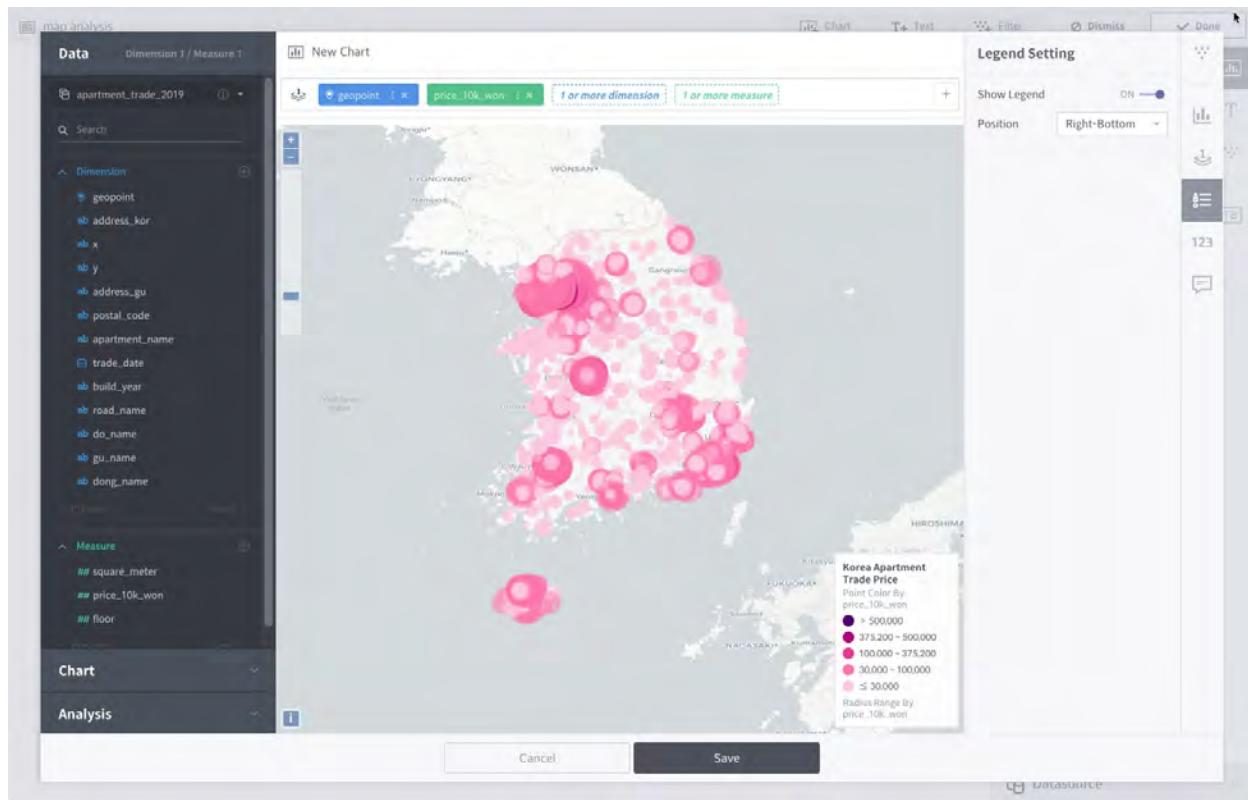
Layer properties of polygon type



1. **Layer Name:** Set a name of the layer for legend and tooltip settings in the map view.
2. **Color:** Data points can be distinguished by color based on a string dimension or a measure on the layer shelf. A color can be picked from the palette if color standards are not available. The transparency can be set as a % value.
3. **Outline:** When set to On, an outline is drawn for each polygon. The default is Off, and the color and thickness are customizable.

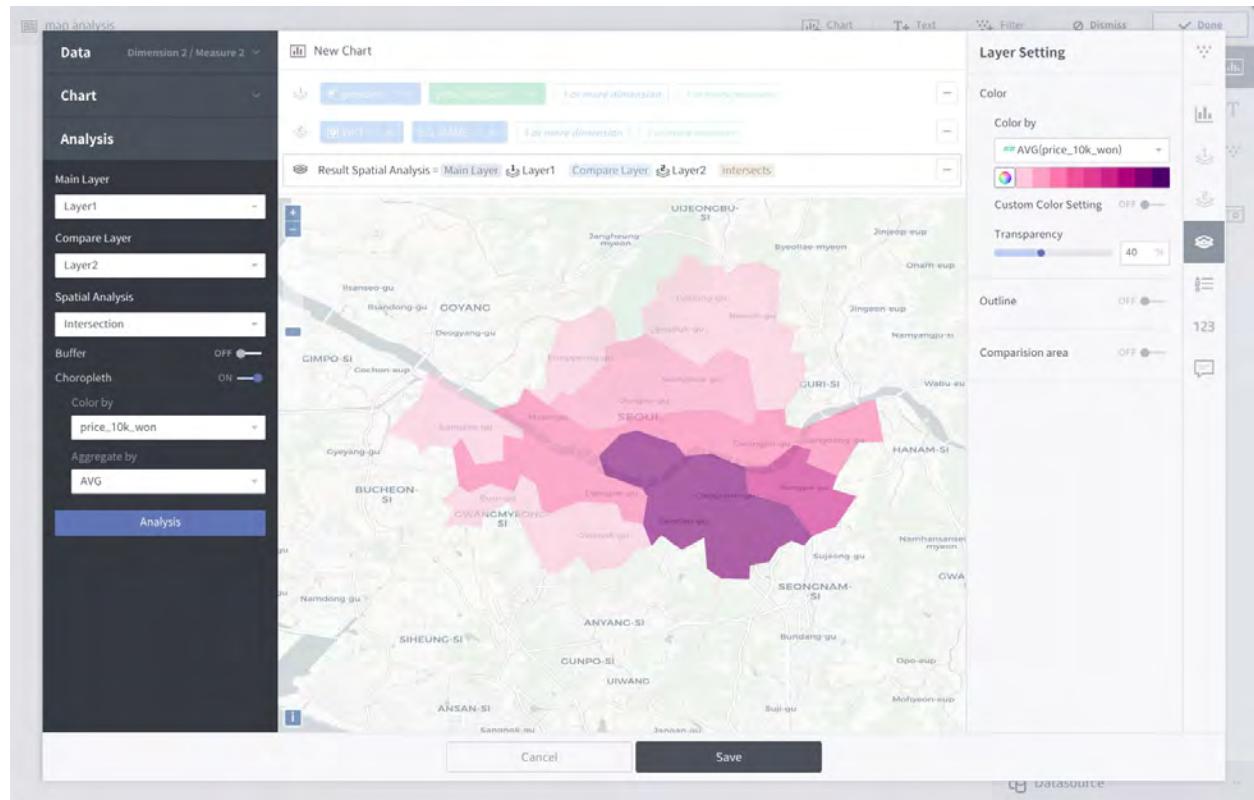
Legend settings

Choose whether or not to display a legend. The default is Off. The position of the legend can be set when turned on.



Spatial analysis

The map view of Metatron Discovery supports simple spatial analysis between two layers. Spatial operations can be set in the analysis tab on the left, and the current version supports two types of spatial operations.



- **Within:** This returns values within a distance designated between elements of the Main and Compare Layers.
- **Intersection:** This method returns overlapping areas between the Main and Compare Layers. Return values may vary with the scale of the geometry selected (Polygon > Line > Point).

Additional settings that can be customized for each operation are as follows:

- **Buffer:** Set a tolerant distance within which the Main and Compare Layers could be compared. The distance can be set either in meters or in kilometers.
- **Choropleth map:** The resulting layer can be displayed in the form of a choropleth map. The color scheme of the choropleth map can be selected; by default, colors are divided according to the data count. If the Main Layer includes a measure, colors can be changed based on its elements.

5.4 Filter

Filters are to display only data matching their preset conditions when forming dashboards and charts. Charts use two types of filters: chart filters and global filters. Chart filters are applied to individual charts, whereas global filters are applied to an entire dashboard.

5.4.1 Chart filters

A chart filter defines what range of data is to be shown on the chart. This chapter describes how to set up and make use of chart filters.

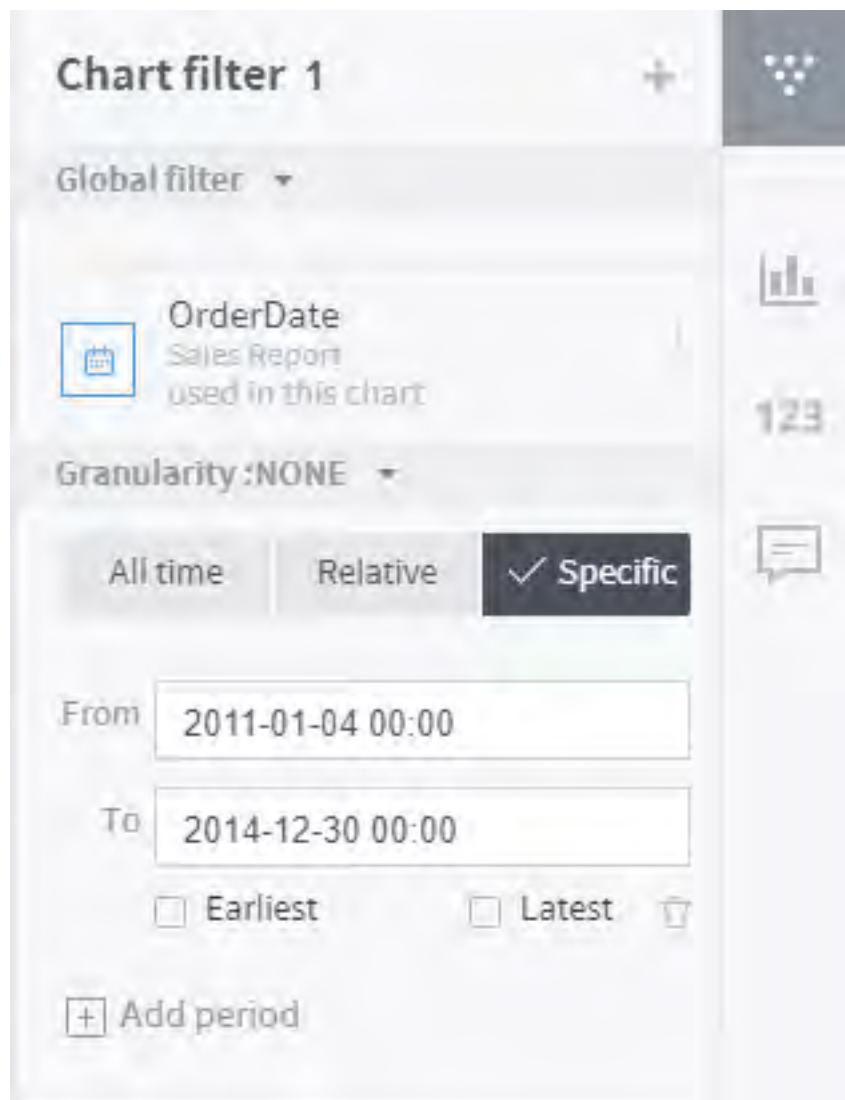
Automatically included filters

The following column filters are included automatically when a chart is created:

- **Timestamp column filter:** As a time-series data store, the Metatron engine necessarily uses a time filter.
- **Recommended filters:** Column filters designated as “recommended filters” during the registration of the data source.
- **Dashboard filters set global:** Filters applied to all charts registered in the dashboard.

Chart filter panel

The chart filter panel is located on the right-hand side of the chart home screen. On this panel, you can easily view and configure registered filters.



1. **Filter number:** Displays how many filters are registered for the chart.
2. **Add/edit filter:** Click on “+” at the top right to either add a new filter or open a popup for configuring an existing filter.
3. **Columns applied with the filter:** The top part of each individual filter displays which columns are applied with the filter.
4. **Filter settings:** Click the hamburger menu at the top right of an individual filter either to reset the filter or configure the details of the filter.

Chart filter dialog box

Click the button at the top of the chart filter panel or click the button in each filter area to open the chart filter dialog box. With this dialog box, you can add a new filter or configure an existing filter.

The chart filter dialog box is divided into the Dimension and Measure tabs as shown below:

Add chart filter

Sales Report

Dimension

Measure

Search by field name

GeoPoint

OrderDate

Category

City

Country

CustomerName

OrderID

PostalCode

ProductName

Region

Segment

ShipDate

ShipMode

State

Sub_Category

ShipStatus

OrderProfitable

SalesaboveTarget

latitude

longitude

Cancel

Dimension filtering

From the connected data source, select a dimension on which to create a filter.

Region Sales Report [New Chart](#)

Single Multiple

Search by item name ▼ ↑ ⌂

(All) Turn all on off

<input type="radio"/> Central 2322	<input type="radio"/>
<input type="radio"/> East 2845	<input type="radio"/>
<input type="radio"/> South 1620	<input type="radio"/>
<input type="radio"/> West 3200	<input type="radio"/>

All

Defined value

- **Value range:** Select whether to filter the chart by a single or multiple data categories.
 - **Single:** Select one data category by which to filter the chart.
 - **Multiple:** Select multiple data categories by which to filter the chart.
- **Search:** If there are too many elements in the column, this function allows you to limit the results only to those you wish to see.
 - **Search by name:** Search the column element list by name.
 - **Element filtering:** Filters elements either by matching element names with regular expressions or wildcards, or by applying a range condition to a measure.

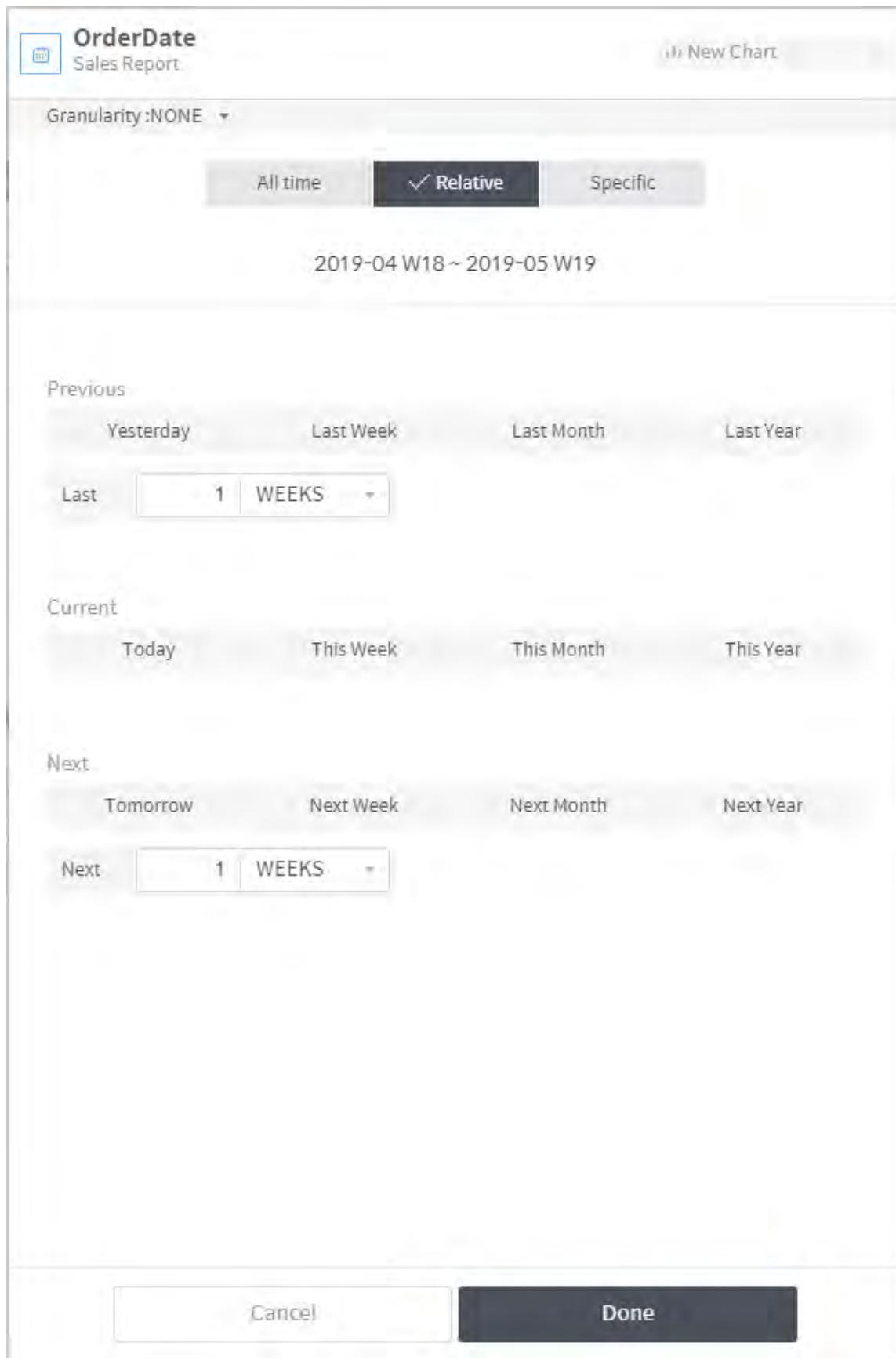
The screenshot shows a configuration interface for a 'Matcher'. It has three main sections: 'Matcher', 'Condition', and 'Limitation'. In the 'Matcher' section, there are two tabs: 'Wildcard' (selected) and 'Regular Expression'. Below the tabs are two input fields: the first is empty, and the second contains the text 'AFTER' with a dropdown arrow. In the 'Condition' section, there are four input fields: 'Select Measure' (dropdown), 'SUM' (dropdown), '=' (dropdown), and '10'. In the 'Limitation' section, there are four input fields: 'TOP' (dropdown), '10' (input field), 'Select Measure' (dropdown), and 'SUM' (dropdown). At the bottom are two buttons: 'Reset' (with a circular arrow icon) and 'Apply' (highlighted in grey).

- **Defined value:** Used to add?as a filter criterion?a data element that is not contained in the column. This allows you to create a filter in advance for a data element that may be added later.

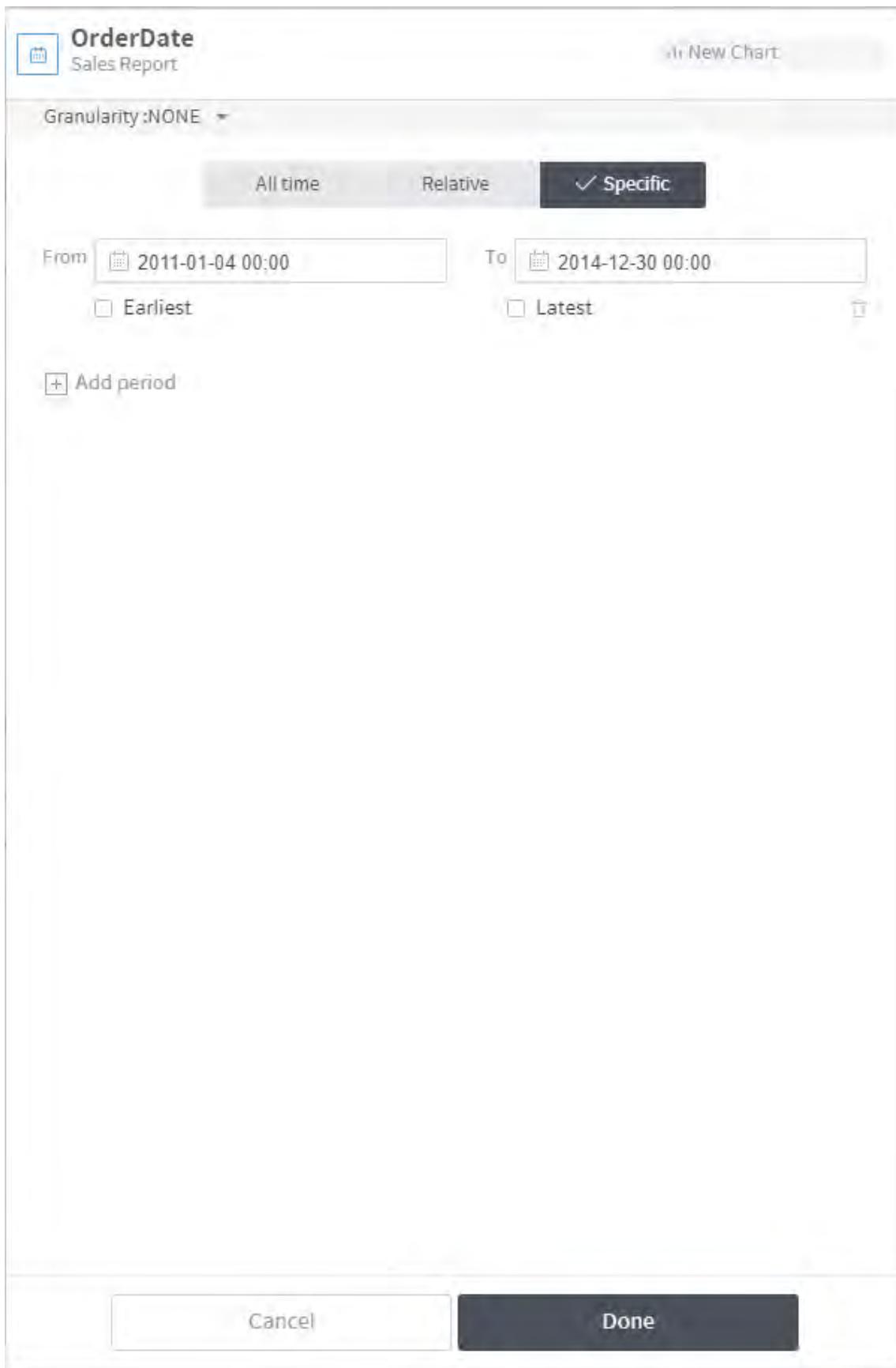
Timestamp column filter settings

Dimensions with a time icon displayed are of a timestamp type for which a timestamp filter can be configured. Although they are set to “All time” by default, you can select Relative or Specific if you wish to display only data from a certain period in the chart.

“Relative” sets a period of time relative to the present and displays only data from the applicable period of time in the chart.



“Specific” directly sets a certain period of time of data and displays only data from the applicable period of time in the chart.



Measure filtering

From the connected data source, select a measure on which to create a filter.

Add chart filter

Sales Report

Dimension Measure

Search by field name:

- ## Discount +
- ## Profit +
- ## Quantity +
- ## Sales +
- ## DaystoShipActual +
- ## SalesForecast +
- ## DaystoShipScheduled +
- ## SalesperCustomer +
- ## ProfitRatio +

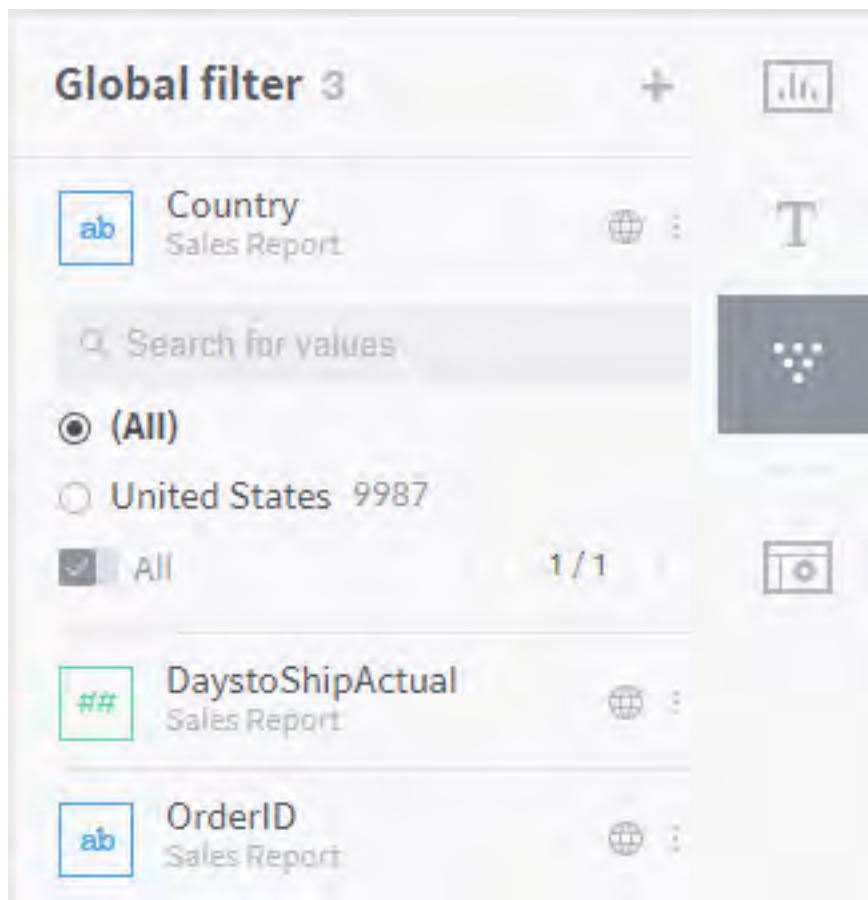
Cancel

Once you have selected a measure, designate the range of values to filter.



5.4.2 Global filters

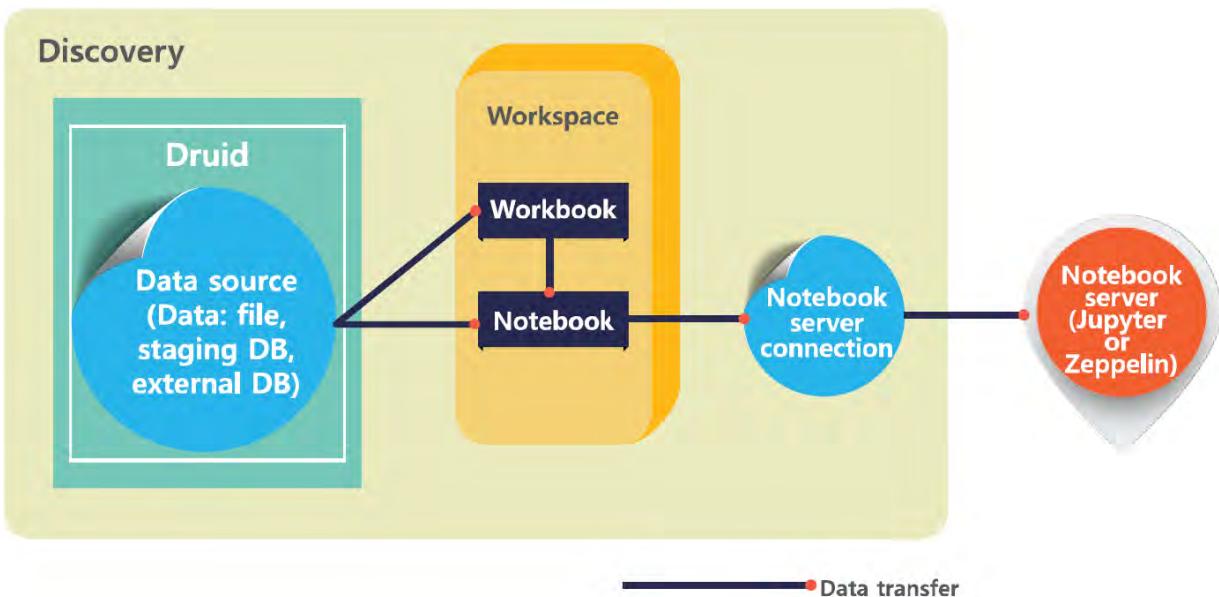
Global filters specify which data is to be displayed in all charts of a dashboard. They can be added, edited, or deleted in the filter panel in the dashboard editing window.



- Number of filter widgets:** Displays how many filter widgets are currently registered in the dashboard next to the global filter heading.
- Add a filter widget:** Click the “+” icon at the top right to create a new filter widget in the dashboard. The filter creation popup interface and process for creating filters are the same as the process for creating chart filters described in the previous section.
- Filter widget list:** Lists filter widgets registered in the dashboard. Hover the mouse over a widget to display the edit and delete icons. Drag a widget to the widget display area to display the widget in the display area.

Global filters applied to the entire dashboard are also listed when creating an individual filter for a new chart. When creating a global filter, if there are any individual chart filters, it intuitively notifies you of which column the filter was created from.

NOTEBOOK



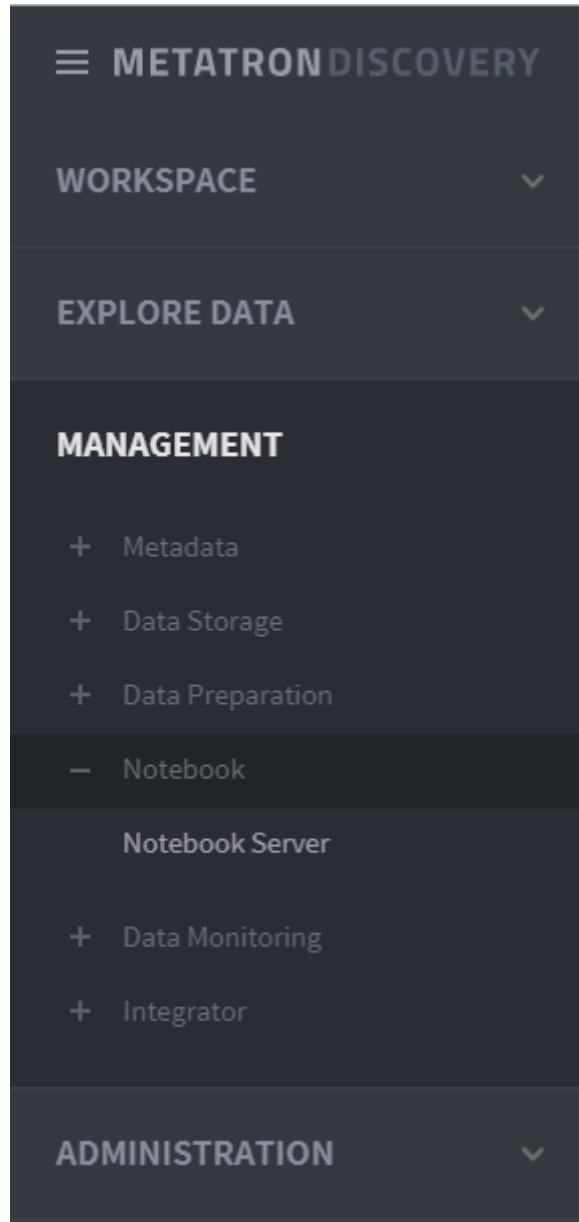
Metatron Discovery supports a notebook function. Notebook is a tool for creating and sharing documents that include live codes, equations, visualizations, and descriptive texts. It is mostly used for data cleaning and manipulation, numerical simulations, statistical modeling, and machine learning.

Metatron Discovery allows users to register and use external Jupyter and Zeppelin servers. Jupyter uses Python and R? programming languages commonly used in data science? while Zeppelin uses Spark (Scala) to help with real-time and interactive analysis and visualization of data. Before running the notebook, its server must be set up.

6.1 Manage notebook servers

To enable the Notebook module, the **administrator** must connect to a “notebook server,” which refers to a server that provides an external analytics tool.

On the left-hand panel of the main screen, go to MANAGEMENT → Notebook → Notebook Server to register a new notebook server or view and edit registered notebook server.



6.1.1 Notebook server list

This page shows a list of notebook servers. The notebook server list can be filtered by server name or type, and clicking on an entry in the list allows you to view and edit the selected server's information. Also, you can delete a notebook server either by clicking its  button that appears when hovering the mouse over the server, or by clicking the Delete selections after selecting the checkboxes next to the servers you want to delete.

Notebook

Notebook Server

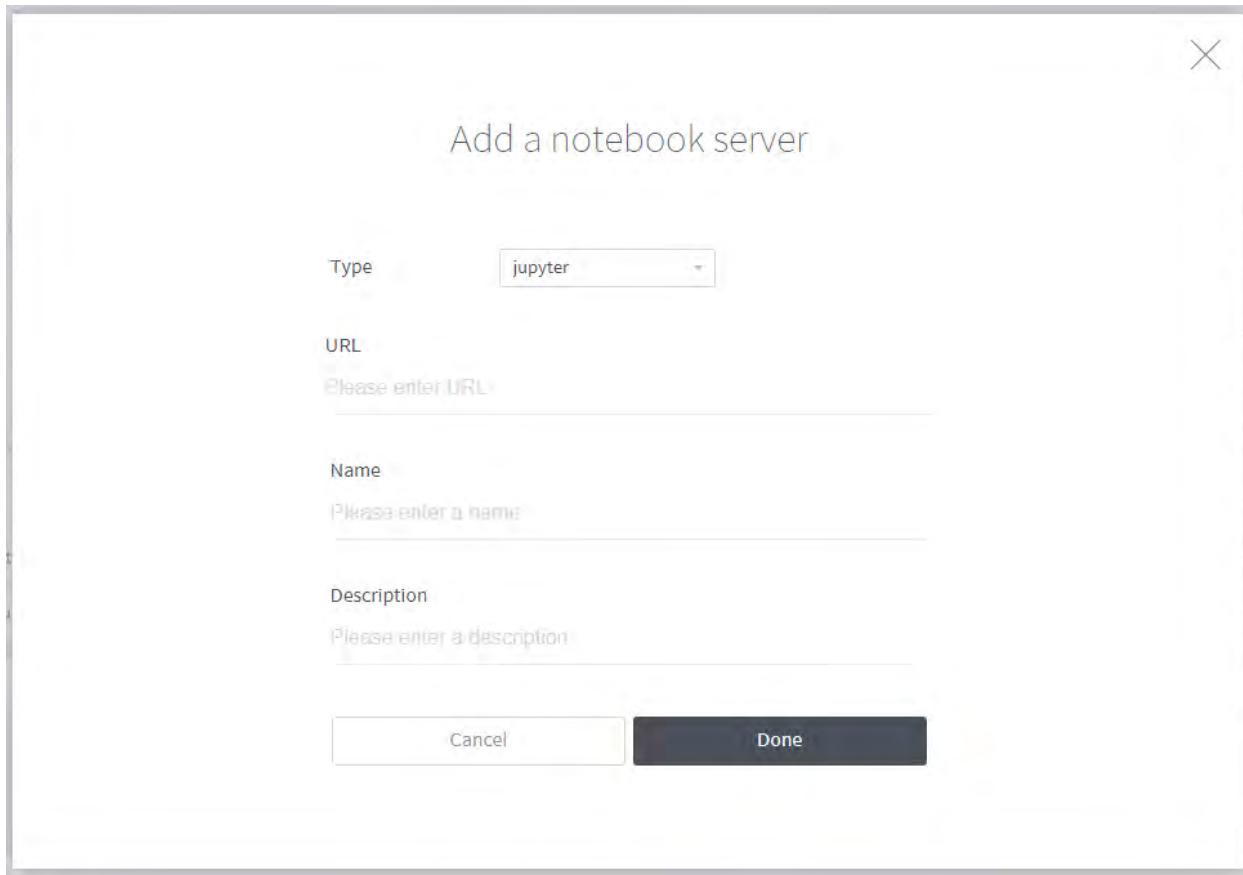
Type	Server	Type	URL	Updated	Created
<input type="checkbox"/>	QA_TEST2-test	 zeppelin	http://jupyter.mcloud.sktelecom.com:80	2019-08-22 13:04 by admin	2019-08-22 13:01 by admin
<input type="checkbox"/>	QA_Test-테스트입니다 수정 확인	 jupyter	http://www.	2019-08-22 12:59 by admin	2019-08-22 12:58 by admin
<input type="checkbox"/>	jupyter-수정	 jupyter	http://metatron-web-04:8888	2019-08-20 14:41 by admin	2019-07-02 19:03 by admin
<input type="checkbox"/>	asd-asd	 zeppelin	https://zeppelin1.svc.stg.apm.cloud.metatr...	2019-07-22 15:06 by admin	2019-06-14 13:22 by admin
<input type="checkbox"/>	te1-테스트	 zeppelin	http://52.231.201.148:8080	2019-05-20 09:50 by admin	2018-11-23 16:05 by admin
<input type="checkbox"/>	Zeppelin Dev-Metatron 개발서버에 구축된 Zeppelin	 zeppelin	http://metatron-web-04:8080	2019-04-04 14:57 by admin	2019-03-21 14:35 by admin
<input type="checkbox"/>	te2-	 zeppelin	http://150.28.69.116:80	2018-11-23 16:06 by admin	2018-11-23 16:06 by admin
<input type="checkbox"/>	jupyter-default	 jupyter	http://jupyter.mcloud.sktelecom.com:80	2018-08-24 15:49 by Polaris	2018-08-24 15:49 by Polaris
<input type="checkbox"/>	zeppelin-default	 zeppelin	http://zeppelin.mcloud.sktelecom.com:80	2018-08-24 15:49 by Polaris	2018-08-24 15:49 by Polaris

There are 9 lists.  Add a server  Delete all

Show up to 

6.1.2 Add a notebook server

Click the Add a server button in the notebook management home to pop up a window to register a notebook server as follows:

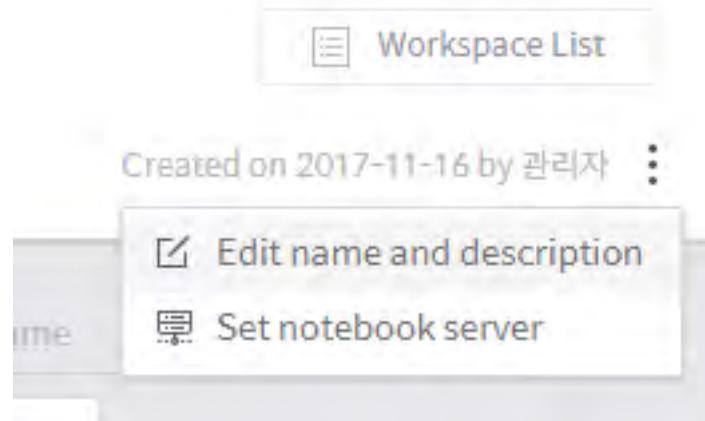


- **Type:** Select the external analytics tool installed in the notebook server to be registered. You can select either **Jupyter** or **zeppelin**.
- **URL:** Enter the URL of the notebook server to be registered. `http://` and `https://` are supported.
- **Name:** Enter a name for the notebook server to be registered.
- **Description:** Enter a description for the notebook server to be registered.

6.2 Register a notebook server

To analyze data in a workspace using a notebook, initial settings are required for the notebook server. The procedure for initial settings for a notebook server is as follows:

1. 워크스페이스의 우측 상단에 있는 버튼을 클릭한 후 노트북 서버 설정을 선택합니다.



- 관리자가 사전에 등록해 둔 Jupyter, Zeppelin 서버 목록 중에서 본인 워크스페이스에서 연결해서 사용하고자 하는 노트북 서버를 선택 후 마침버튼을 클릭합니다.
 - 아무 서버도 선택하지 않고자 한다면, (없음) 항목을 선택하십시오.

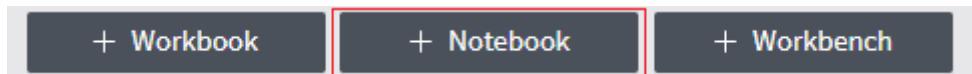
A screenshot of a 'Set notebook server' dialog box. It has tabs for 'Jupyter' and 'Zeppelin'. Below them, it says 'Connected server : jupyter'. There's a search bar labeled 'Search by sever name'. A table lists four servers:

Server	URL
(None)	
jupyter-수정	http://metatron-web-04:8888
jupyter-default	http://jupyter.mcloud.sktelecom.com:80
QA_Test -test입니다 수점확인	http://www.

6.3 Create a notebook

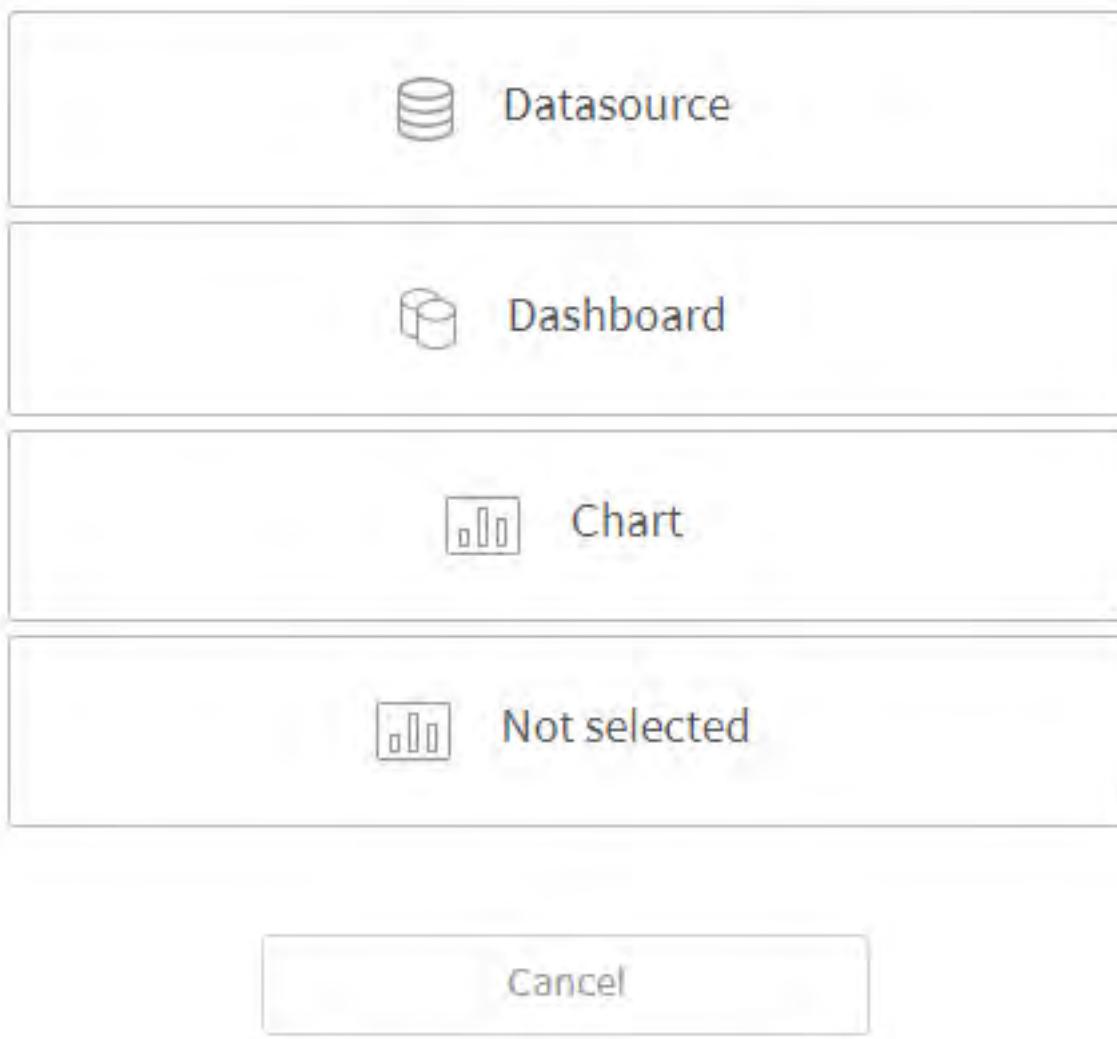
Once the notebook server has been set up, you can create a notebook. A notebook is created as follows:

1. Click the **+ Notebook** button at the bottom of the workspace. You'll be prompted to create a notebook.



2. Select the type of data set that you wish to analyze in the notebook. You can choose between **Data source**, the unit of data used in Metatron Discovery, **Dashboard**, **Chart**, and **Not selected**. If you want to use Zeppelin, select **Not selected**.

Select a data type



3. After selecting either **Data Source**, **Dashboard**, or **Chart**, you can see a list of data currently registered in Metatron Discovery. Select the data to analyze and click Next.

Please select a datasource

No.	Datasource	Type	Used in	Updated
32	aaaaaaaaaa [Open data]	Ingested...	All Notebooks	2018-05-15
31	AirBnB_in_NYC [Open data]	Ingested...	All Notebooks	2018-05-15
30	Economy - Tableau example data [Open data]	Ingested...	All Notebooks	2018-05-15
29	edas_sample_100_01 셈플용 [Open data]	Ingested...	All Notebooks	2018-05-16 ✓
28	employee_001 [Open data]	Ingested...	All Notebooks	2018-05-15
27	Excel-test - Excel-test [Open data]	Ingested...	All Notebooks	2018-05-15
26	ignoreInvalidRows_sample [Open data]	Ingested...	All Notebooks	2018-05-15
25	null_test [Open data]	Ingested...	All Notebooks	2018-05-15

More ▾

edas_sample_100_01 X
 Created: 2018-05-15
 Type: Ingested type
 Visibility: Public
 Size: 31.40 MB
 Rows: 3,700,900

Dimension: edas_sample_100.event_time
 Dimension: edas_sample_100.dim_01
 Dimension: edas_sample_100.dim_02
 Dimension: edas_sample_100.dim_03
 Dimension: edas_sample_100.dim_04
 Dimension: edas_sample_100.dim_05
 Dimension: edas_sample_100.dim_06
 Dimension: edas_sample_100.dim_07
 Measure: edas_sample_100.meas_01

Cancel Next

4. Enter the information about the notebook that you want to use as an analytics tool for data. The **server type** can only be selected for a notebook server connected at the initial notebook server setup. If **Jupyter** is selected, “R” or “Python” can be selected for analysis, whereas “Spark” (Scala) is used when **Zeppelin** is selected.

 Create a Notebook

○ — ●

Please complete notebook creation

Chart sale performance > sales performance dashboard > q-over-q

Server type

Develop language

Name
Please enter a name

Description
Please enter a description

- Once a notebook has been created, you can find it in the workspace.

The screenshot shows the 'Admin workspace' in Metatron Discovery. At the top, there are tabs for 'Workbook 55', 'Notebook 14' (which is highlighted in blue), and 'Workbench 42'. Below the tabs, there's a search bar and a message indicating the workspace was created on 2017-11-16 by 관리자. The main area displays a grid of workspace items. A red box highlights the 'notebook' item titled 'dfads', which was last updated a day ago. Other visible items include 'metatron', 'SJ', 'sohnw', '시연용(변경금지)', '엑스투', and '서브폴더'. Below the grid, there are sections for 'workbook' and 'workbench' with their respective details. At the bottom, there are buttons for 'Select all', 'Move Selections', 'Delete Selections', and three buttons labeled '+ Workbook', '+ Notebook', and '+ Workbench'.

6.4 Use a notebook

In a newly created notebook, you can write a script and serve it through a REST API. A notebook can be used as follows:

6.4.1 Detailed notebook page

On the workspace screen, select the notebook you want to use as an analytics tool. Then, the following screen with detailed information appears. You'll see basic information on the notebook: data type, data source name, development language, and analytic code, etc.

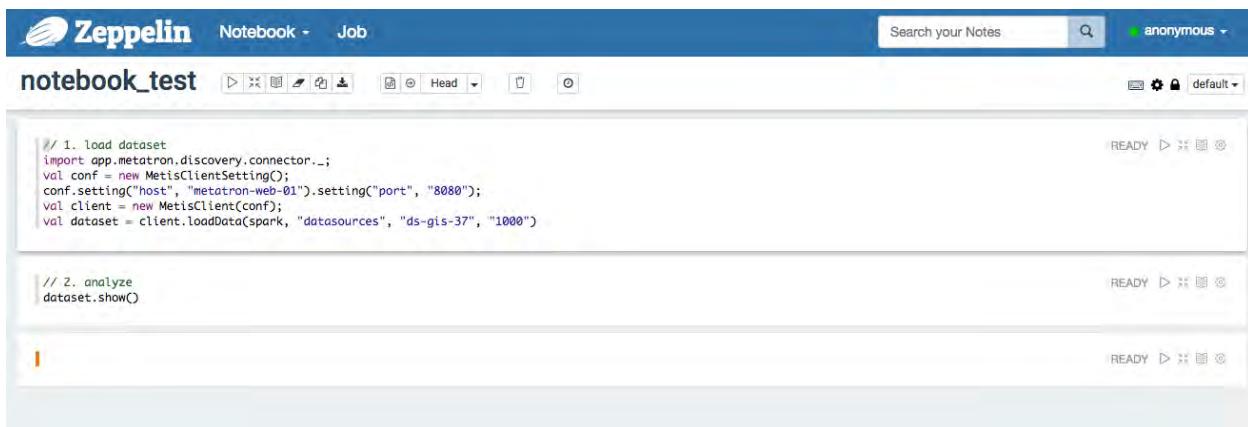
The screenshot shows a dataset named 'dfads' in a notebook. The dataset has the following properties:

Property	Value
Data type	DATASOURCE
Datasource	Economy
Develop language	SPARK
Code	Detail

Below the properties, there is an API section with the message "No API information." and a "Create API" button.

6.4.2 Notebook coding

Click **Detail** on the notebook page to pop up a new window for coding in the notebook. At the top of this window, a code to load a dataset is inserted; executing this cell loads a JSON dataset as the dataset object.



The screen above appears when Zeppelin is selected and includes a cell for loading the data selected when the notebook was created. After coding the program starting from the third cell, click **Save** when you are finished.

6.4.3 Register a notebook API

Once you write a notebook code, you can return the results by calling a REST API. Select a **Return type** by referring to the descriptions below, and enter a **Name** and **Description**.

X

API information

Return type HTML JSON NONE

Name
Please enter a name

Description
Please enter a description

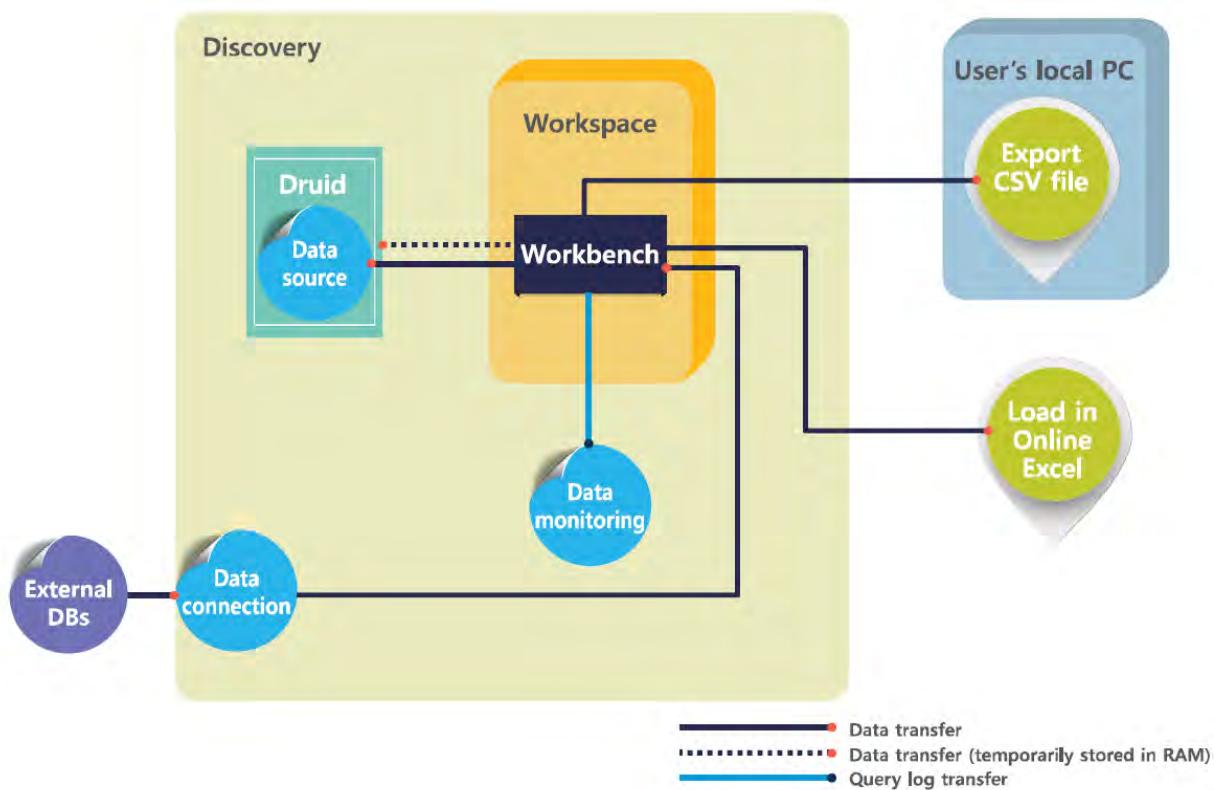
- **HTML:** The results of running the notebook script are returned in HTML.
- **JSON:** The results of running the notebook script are returned in a custom JSON format. In this case, the `response.write(...)` function provided by Metatron Discovery will be used. The following is an example code for using the `response.write` function:
 - R-based notebook: `response.write(list(coefficient = 2, intercept = 0))`
 - Python-based notebook: `response.write({'coefficient' : 2.5, 'intercept' : 0})`
- **None:** Runs the notebook script but does not provide returns.

Once you enter API information and click **Done**, the API is created to provide a REST API URL as shown below. Click **Result** to view the URL execution results in a popup window.

Name	RESTful API
Description	
URL	http://metatron.mcloud.sktelecom.com/api/notebooks/rest/354599d4-444a-43ab-b966-fdadffd12e7e
Return type	HTML
API result	Result

[Edit API](#) [Delete API](#)

WORKBENCH



Metatron Workbench provides an environment for data preparation and analytics based on SQL. Its main functions are as follows:

- Various external databases can be loaded in one space.
- The user can conveniently navigate/select linked tables and columns and view their details.
- Query edit tools are embedded and query results can be viewed interactively and available for various uses:

- Query results can be downloaded into a local file or exported to an online Excel.
- Query results can be interactively visualized to help the analyst see an outline of the resulting data table.
- Query results can be stored as a data source available for analytics in a workbook or notebook.

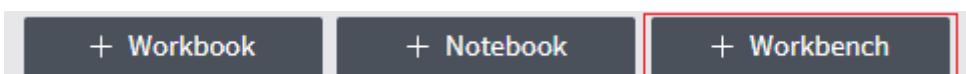
Each document that stores SQL-based analytic queries is called a “workbench.” This chapter introduce how to **create** and **use** workbenches.

7.1 Create a workbench

To use a workbench in the workspace, a workbench-type data connection must be established. See [Data Connection](#) for how to handle it.

To create a workbench:

1. Click the **+ Workbench** button at the bottom of the workspace. You’ll be prompted to select a data connection for data analytics.



2. Select the workbench-type data connection that connects to the data table you want, and click **Next**.

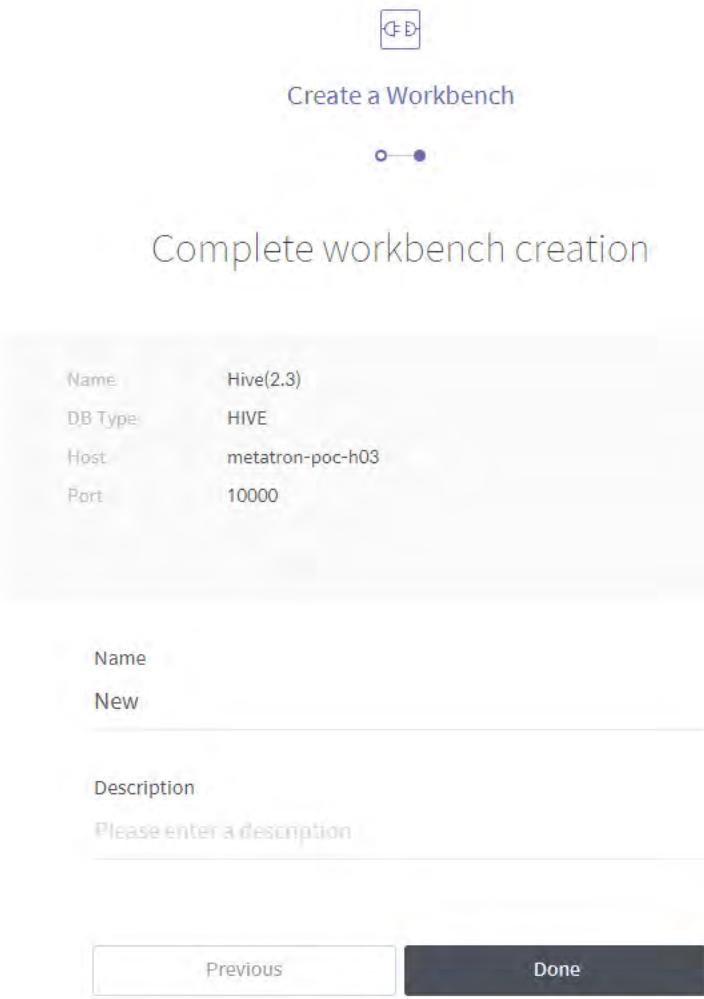
Please select data connection

No.	Data connection	Type	Host	Port	Account type	Updated
5	Tibero_Exntu	TIBERO	exntu.kr	8629	Enter by manager	2018-05-08
4	local_mysql	MYSQL	metatron-po...	3306	Enter by manager	2018-04-10
3	azure-mysql-test	MYSQL	metatron-po...	3306	Enter by manager	2018-03-22
2	Hive(2.3)	HIVE	metatron-po...	10000	Enter by manager	2018-01-10
1	Hive(1.2)	HIVE	metatron-po...	10000	Enter by manager	2017-11-23

More ▾

Cancel Next

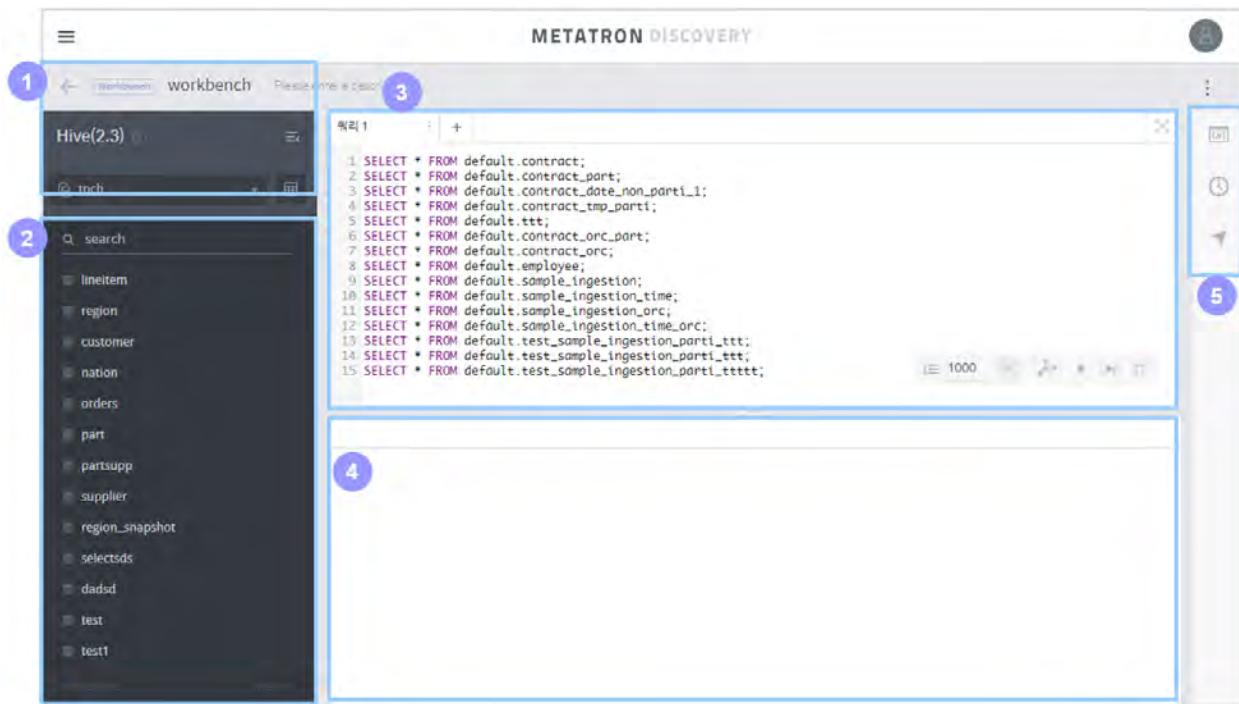
- Search by name of data connection:** Searches the list of data connections available to the workspace by the name you type in.
 - DB type:** Filters data connections by database type (Oracle/MySQL/Hive/Presto/Tibero). Select All to display data connections regardless of database type.
 - Account type:** Filters data connections by account type (All/Always connect/Connect by user's account/Connect with ID and password). Select All to display data connections regardless of account type.
 - Data connection:** Lists data connections filtered by specified criteria.
3. Confirm the information of the selected data connection and enter a name and a description to create a workbench.



4. The created workbench is immediately available.

7.2 Use a workbench

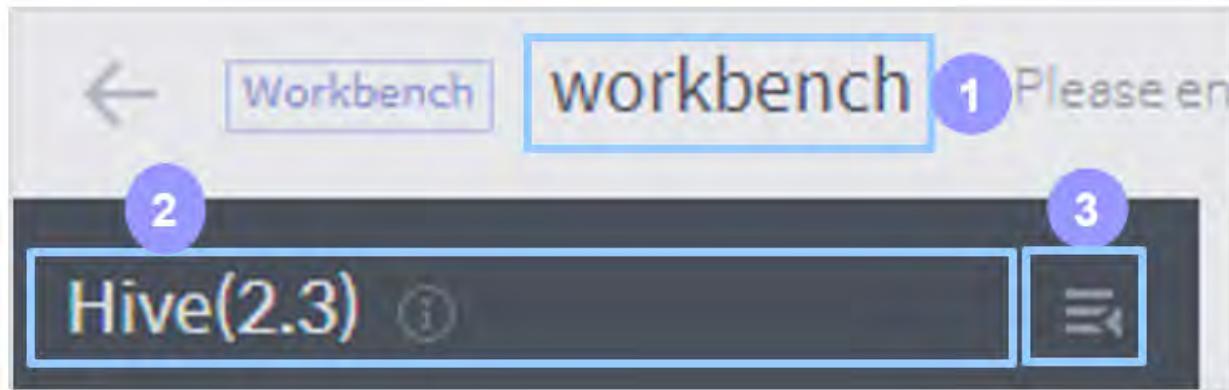
In the workbench, you can edit and manage an SQL database easily, as well as visualize and store the results of a query on it in various forms. The workbench page consists of five sections shown below, and an additional schema browser is provided.



1. Basic information section (See [Basic information section](#))
2. Schema and table section (See [Schema and table section](#))
3. Query editor section (See [Query editor section](#))
4. Query results section (See [Query results section](#))
5. Extra tools section ([Extra tools section](#))
6. Schema browser ([Schema browser](#))

7.2.1 Basic information section

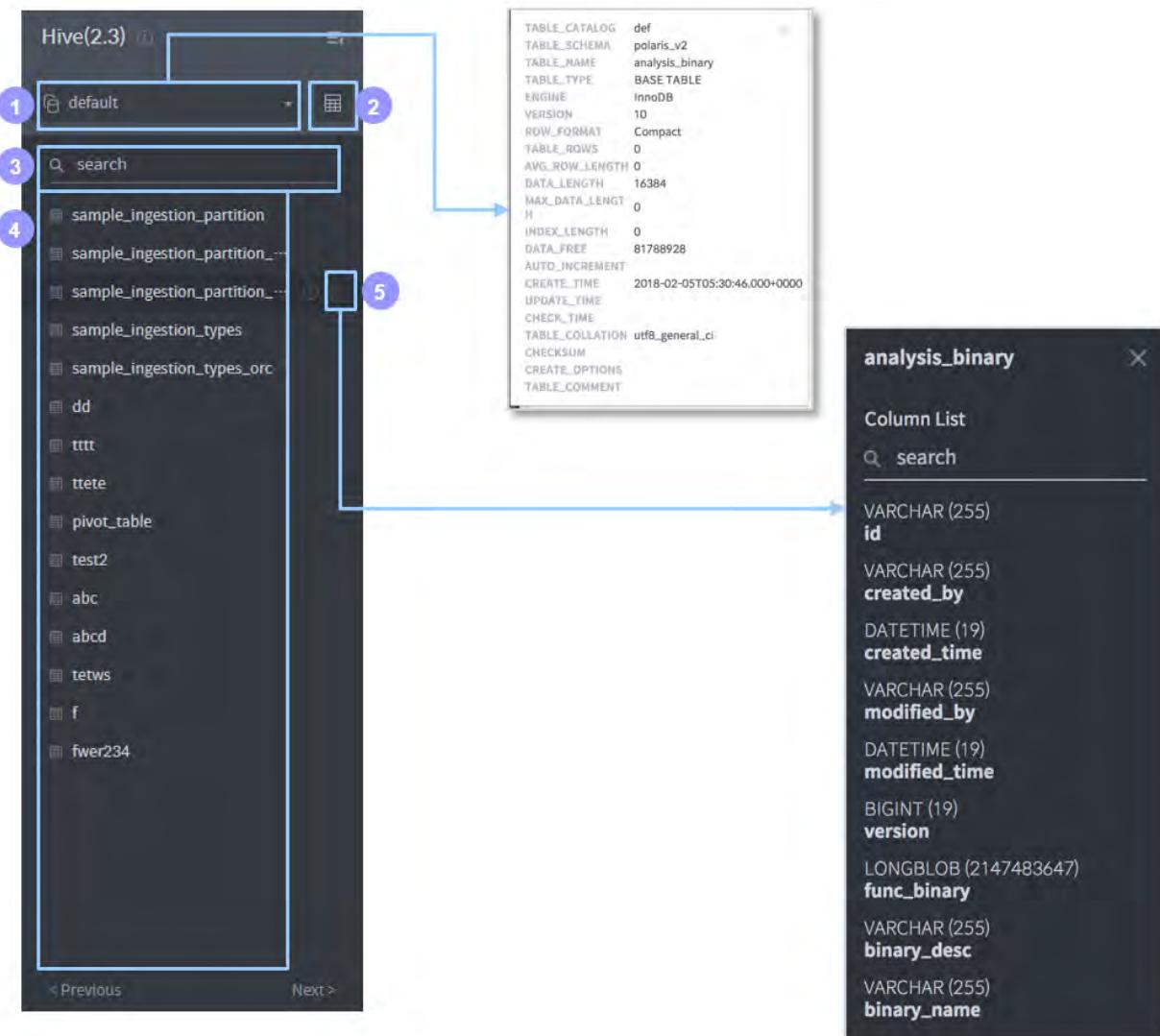
This section displays basic information on the active workbench.



1. **Name:** Name of the workbench. Click on it to change the workbench's name.
2. **Data connection:** Name of the data connection used by the workbench. Click the icon to view its details.
3. : UI button to collapse or expand the panel.

7.2.2 Schema and table section

This section provides a UI to conveniently insert the name of a database, table, or column in the query editor.

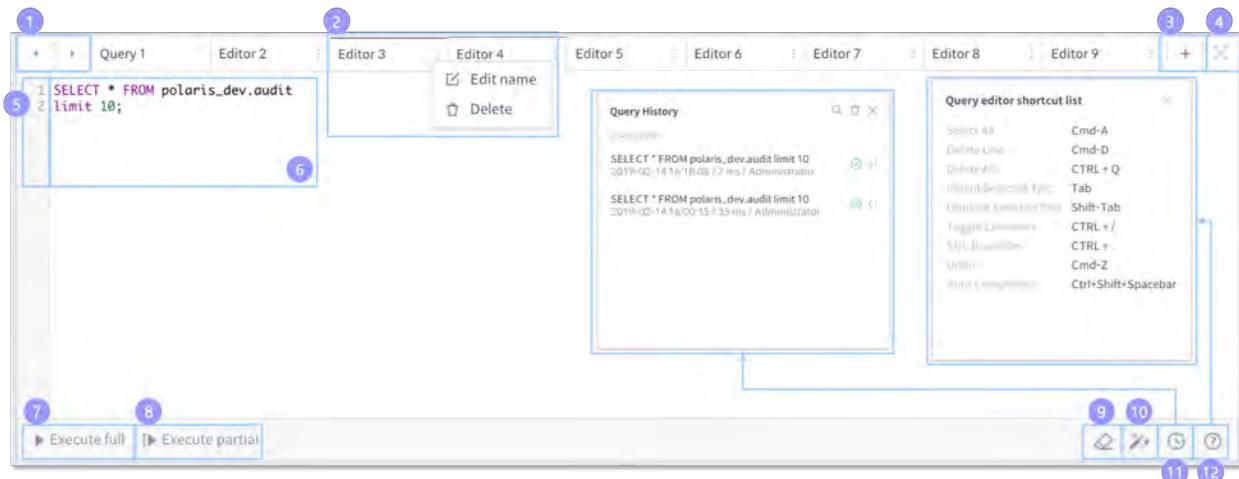


1. **Database name:** Displays the name of the selected database. By default, the first database of the data connection used by the workbench is selected. Click on it to list all databases included in the data connection. Select a database in the list to replace the currently selected database.
2. **Schema browser:** A popup browser displaying the table list of the selected database, and information of all the columns and records in each table.
3. **Search table:** Searches the list of the tables registered in the selected database by the name you type in.
4. **Table name:** Select a table to automatically insert it in the query editor along with a `SELECT * FROM {table name}` query.
5. **Column list:** Displays all columns belonging to the table and their respective data types.

Click a column name to automatically insert it in the query editor.

7.2.3 Query editor section

This section allows you to edit and run queries.



1. : Navigates to tabs of previous or subsequent queries when there are too many tabs. If tabs are not many, this button will not appear.
2. **Tab**: You can run or store queries in separate tabs for more efficient management of them. Click the button to edit the tab title or delete the tab.
3. : Click this button to add a new tab.
4. : Click this button to minimize the query editor or maximize it to full screen.
5. **Query lines**: Displays the numbering of the query code lines.
6. **Editor area**: Write query statements in this area. You can run either single or multiple queries. Insert ; at the end of each query statement to run them separately. Autocomplete is supported.
7. **Execute full**: Execute all queries in the editor. (Shortcut: Ctrl + Enter)
8. **Execute partial**: Executes only the query statement where the cursor is located, or execute queries selected by dragging the mouse. (Shortcut: Command + Enter)
9. **CLEAR SQL**: Clears all query statements.
10. **SQL BEAUTIFIER**: Re-words query statements using standard query syntax.

11. **Query History:** Lists past queries executed in the query editor. If you select a query in the list, it will be inserted in the query editor.
12. **Query Editor Shortcuts:** Shows a list of shortcuts available in the query editor.

7.2.4 Query results section

Once a query is executed, its results are displayed in a query results tab. Query results tabs are cumulatively added, and you can selectively delete specific results tabs. Query results are displayed in a text grid, and they can be previewed in charts, stored into data sources, and exported into CSV files.

During query execution



1. **Query result tabs:** When multiple queries are executed, a different tab is created for each query to show its result. While a query's execution is in progress, "Loading" is displayed in its tab title.
2. **Query log:** Shows an execution log for the query. In the case of a Hive connection, a Hive job log is additionally displayed.
3. **Cancel:** Cancels the execution of the query. The time taken for cancellation may vary with the DB type.
4. **Query execution phase:** Shows the current phase of query execution. There are a total of five query execution phases.
 - Getting connection
 - Creating statement

- Executing query
 - Getting result set
 - Done!
5. **No. of the current query:** Shows the number of the currently executed query when multiple queries are executed.
 6. **Output/Result tabs:** By clicking either tab, you can switch to the query log/result view.
 7. **Query status:** Shows the query's status from among:
 - Running query
 - Query execution failed..
 - Query execution canceled..
 8. **Query start time:** Displays when the query execution started.
 9. **Query running time:** Displays how long it took to execute the query.

After query execution

The screenshot shows the Metatron Workbench interface after a query has been executed. A data table is displayed in a tab labeled 'Result'. The table has columns: No., lineage.eventtime, lineage.cluster, lineage.currentdatabase, lineage.targettabletype, and lineage.expr. The data consists of 10 rows, each with the value '1001' in the 'No.' column and '1521503450894' in the 'lineage.eventtime' column. The 'lineage.cluster' column contains 'collector', 'adw', 'DFS_DIR', and 'DFS_LN' repeated multiple times. The 'lineage.currentdatabase' column contains 'adw' and 'DFS_DIR'. The 'lineage.targettabletype' column contains 'DFS_DIR' and 'DFS_LN'. The 'lineage.expr' column contains 'DFS_DIR' and 'DFS_LN'. Below the table, a status bar shows 'Start: 2019-02-14 17:39:56', 'Finish: 2019-02-14 17:39:58', 'Running, 1.7 Secs.', and '2,000 / 2,500 Rows'. The interface includes tabs for 'Output' and 'Result', and various navigation and search controls at the bottom.

Callouts numbered 1 through 12 point to specific elements of the interface:

1. Top-left corner of the result table.
2. Second tab from the left in the tab bar.
3. Tab bar showing multiple tabs: Editor 2 - Result1, Editor 2 - Result2, Editor 2 - Result3, Editor 2 - Result4, and Editor 2 - Result5.
4. Column header for 'lineage.eventtime'.
5. Column header for 'lineage.cluster'.
6. Column header for 'lineage.currentdatabase'.
7. Column header for 'lineage.targettabletype'.
8. Column header for 'lineage.expr'.
9. 'Output' tab in the bottom navigation bar.
10. 'Result' tab in the bottom navigation bar.
11. Status bar showing execution time.
12. Status bar showing row count.

1. **Query result tabs:** When multiple queries are executed, a different tab is created for each query to show its result. While a query's execution is in progress, “Loading” is displayed in its tab title.
2. **Data details:** Shows a data table resulting from executing the query. You can copy this data output to the clipboard.

3. **Output/Result tabs:** By clicking either tab, you can switch to the query log/result view.
4. **Search for column data:** Searches for a column or value in the resulting table.
5. **Chart preview:** Draws a virtual chart of the query results. This chart is only for visualization; it is not stored in the workspace. (See [Chart](#) for how to handle it)
6. **Save as Data source:** Stores the query results into a data source in the workspace. A dialog box will pop up to create a data source, and the resulting table is used instead of selecting a data connection and a table. Therefore, you will be immediately prompted to set the schema definition and ingestion cycle. (See [Create a data source](#) for how to handle it)
7. **Export CSV file:** Downloads the resulting table into a local file (CSV).
8. **Data page navigation:** If the resulting data includes more than 1,000 rows, you can navigate the data pages using the Prev and Next buttons.
9. **Query start time:** Displays when the query execution started.
10. **Query finish time:** Displays when the query execution finished.
11. **Query running time:** Displays how long it took to execute the query.
12. **Query data rows:** Shows the number of rows of the resulting data and the current page number.

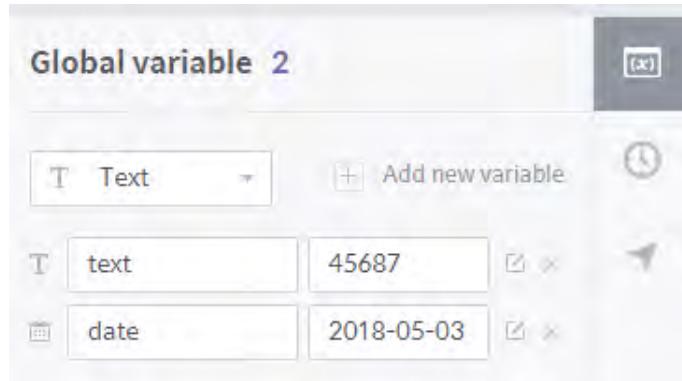
7.2.5 Extra tools section

The extra tools section provides useful tools for the workbench.

- Setting up global variables for repeatedly used statements (See [Setting up global variables](#))
- Navigation to move to another workbench (See [Workbench Navigation](#))

Setting up global variables

If a certain type of statement is repeatedly used with a different value for each query run, set the variable element as a “global variable” for convenient use.



- **Variable type:** You can select either a calendar or text type.
- **Add new variable:** Select the variable type you want and click “Add new variable.” A new global variable will be added in the query editor.
- **Name:** Enter a name for the variable.
- **Variable value:** For a calendar variable, select a date; for a text variable, select a text value.

Workbench Navigation

Used to move to another workbench. Click the target workbench to move to.

Workbench Navigation (15)

 search

No.	Workbench name	Updated
42	workbench	2018-05-16
41	new	2018-05-16
40	metatron_metadata	2018-05-15
39	stage_03	2018-05-15
38	Tibero	2018-05-08
37	로컬 확인용	2018-04-30
36	aaa	2018-04-30
35	gg	2018-04-26
34	test-Magenta	2018-04-20
33	teddypark	2018-04-19
32	test_workspace	2018-04-16
31	경제지표	2018-04-09
30	111	2018-04-05
29	ddd	2018-04-05
28	test	2018-04-04

- **Search for workbench:** Search for a workbench stored in the workspace.
- **Workbench list:** Displays all workbenches stored in the workspace. Click a workbench in the list to move to that workbench.

7.2.6 Schema browser

Displays the table list of the selected database, and information of the columns and records in each table.

Scheme Information			
		Columns	Information
No.	Column Name	Type	Description
1	base_time	STRING(2147483647)	
2	local_time	STRING(2147483647)	
3	recv_time	STRING(2147483647)	
4	os_name	STRING(2147483647)	
5	os_version	STRING(2147483647)	
6	resolution	STRING(2147483647)	
7	screen_width	BIGINT(19)	
8	screen_height	BIGINT(19)	
9	language_code	STRING(2147483647)	
10	rake_lib	STRING(2147483647)	
11	rake_lib_version	STRING(2147483647)	
12	ip	STRING(2147483647)	
13	recv_host	STRING(2147483647)	
14	token	STRING(2147483647)	
15	log_version	STRING(2147483647)	
16	device_id	STRING(2147483647)	
17	device_model	STRING(2147483647)	
18	manufacturer	STRING(2147483647)	
19	carrier_name	STRING(2147483647)	
20	network_type	STRING(2147483647)	
21	app_version	STRING(2147483647)	
22	browser_name	STRING(2147483647)	
23	browser_version	STRING(2147483647)	
24	referrer	STRING(2147483647)	
25	url	STRING(2147483647)	
26	document_title	STRING(2147483647)	

- **Column:** Shows the names and data types of all columns of the selected table.
- **Information:** Displays attributes of the selected table.
- **Data:** Displays data of the selected table. A maximum of 50 rows can be viewed.

CHAPTER EIGHT

DATA PREPARATION

Data Preparation is a tool that creates transformation rules to transform files and tables for more convenient analysis of datasets, and saves the results into HDFS or Hive.

Advantages of data preparation in Metatron Discovery

The screenshot shows the Metatron Discovery Data Preparation interface. At the top, there's a header bar with a file icon, the title "sales.csv", and buttons for "Snapshot" and "Done". Below the header, a summary bar displays "28 Columns", "100 Rows", and "47 types". The main area is a table with columns: "OrderDate", "Category", "City", "Country", "CustomerName", "Discount", and "OrderID". Each column has a histogram and a count of categories. The "CustomerName" column has 42 categories, while others have fewer. The "Discount" column shows values like 0.2, 0.8, and 0. The "OrderID" column has 45 categories. To the right of the table is a "RULE (9) - SNAPSHOT (1)" panel containing a list of nine transformation steps:

- create with sales.csv
- convert row 1 to header
- set type _OrderDate_ to Timestamp
- set type ShipDate to Timestamp
- set type 9 columns to Long
- set type 3 columns to Double
- drop SalesAboveTarget_1
- drop orderprofitable_1
- drop location

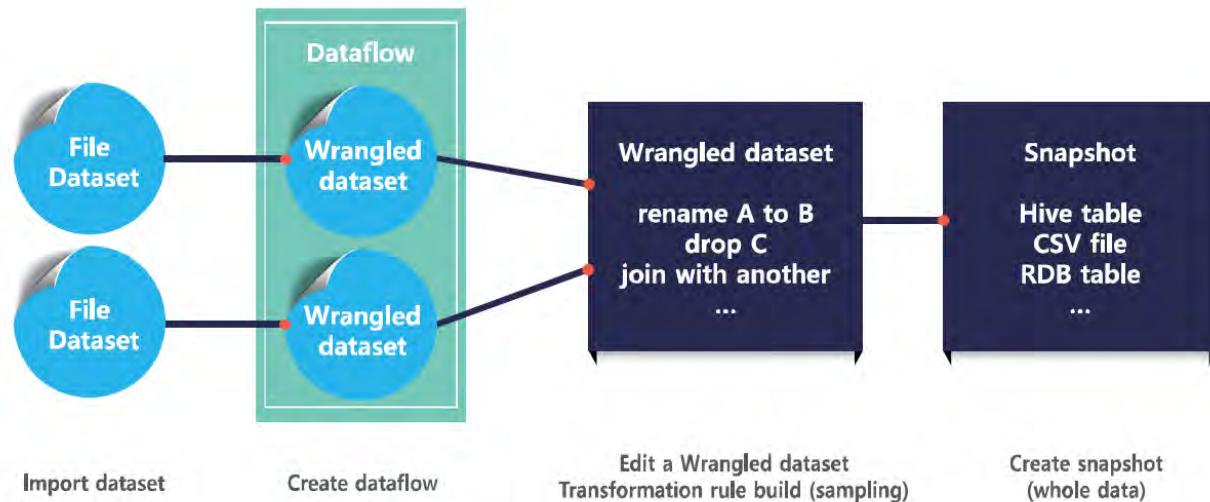
At the bottom left, there's an "Add rule" button and a "Choose Rule Function" dropdown. On the right, there are "Cancel" and "Next" buttons.

Users can create transformation rules by following the step-by-step process as shown in the above GUI. Since the transformation results from each step are stored in memory together with the data distribution, users can easily check the results through the simple click of a button and perform **undo** and **redo** just like using a text editor.

Based on these characteristics, the data preparation tool offers the following advantages:

- Users unfamiliar with programming or data processing can obtain the desired results.
- Adding a transformation rule usually involves programming or writing an SQL query. However, Metatron Discovery's Data Preparation provides a GUI for **exploratory transformation** that enables the creation of transformation rules simply by clicking a button or typing.
- Basic data transformation is conducted automatically. For instance, a type cast is automatically applied to columns comprised of numerals. This is made possible by the **undo** and rule deletion functions.
- Data of different forms can be combined as desired (e.g. reference file + fact table).
- The results of data refinement can be shared with others, thus reducing the burden of exchanging physical data.
- Storage space is saved and **information life cycle (ILM)** shortened by deleting the actual data and retaining only the transformation rules involved. The actual data can be easily created whenever needed.

Structure of data preparation in Metatron Discovery



As shown in the above figure, data preparation is comprised of a **dataset** built from the target data, a **dataflow** that defines transformation rules for the designated dataset, and a **data snapshot** that shows the transformation results.

8.1 Install Guide Detailed

This document is a guide for installing metatron and using data preparation feature from the scratch Linux OS environment (CentOS 7).

8.1.1 1. Install requirements

Run following commands by root.

```
yum clean all && yum repolist && yum -y update
yum -y install tar unzip vi vim telnet apr apr-util apr-devel apr-util-devel net-tools curl openssl
elinks locate python-setuptools
yum -y install java-1.8.0-openjdk-devel.x86_64
export JAVA_HOME=/usr/lib/jvm/java
export PATH=$PATH:$JAVA_HOME/bin
```

8.1.2 2. Install Hadoop

Run below commands by root. You'd better to download the Hadoop binary from the closest mirror.

```
yum -y install openssh-server openssh-clients rsync netstat wget
yum -y update libselinux

ssh-keygen -q -N "" -t dsa -f /etc/ssh/ssh_host_dsa_key
ssh-keygen -q -N "" -t rsa -f /etc/ssh/ssh_host_rsa_key
ssh-keygen -q -N "" -t rsa -f /root/.ssh/id_rsa
cp /root/.ssh/id_rsa.pub /root/.ssh/authorized_keys

wget http://archive.apache.org/dist/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz
tar -zvxf hadoop-2.7.3.tar.gz -C /opt
rm -f hadoop-2.7.3.tar.gz
ln -s /opt/hadoop-2.7.3 /opt/hadoop
```

(continues on next page)

(continued from previous page)

```

export HADOOP_PREFIX=/opt/hadoop
export HADOOP_COMMON_HOME=$HADOOP_PREFIX
export HADOOP_HDFS_HOME=$HADOOP_PREFIX
export HADOOP_MAPRED_HOME=$HADOOP_PREFIX
export HADOOP_YARN_HOME=$HADOOP_PREFIX
export HADOOP_CONF_DIR=$HADOOP_PREFIX/etc/hadoop
export YARN_CONF_DIR=$HADOOP_PREFIX
export PATH=$PATH:$HADOOP_PREFIX/bin:$HADOOP_PREFIX/sbin

sed -i "/^export JAVA_HOME/ s/.*/export JAVA_HOME=$JAVA_HOME/" $HADOOP_CONF_DIR/hadoop-env.sh
sed -i "/^export HADOOP_CONF_DIR/ s/.*/export HADOOP_CONF_DIR=$HADOOP_CONF_DIR:" $HADOOP_CONF_DIR/
~hadoop-env.sh

```

Put files below into \$HADOOP_CONF_DIR.

```
core-site.xml hdfs-site.xml mapred-site.xml yarn-site.xml
```

Run followings by root.

```
$HADOOP_PREFIX/bin/dfs namenode -format
```

Append following contents into /root/.ssh/config

```

Host *
  UserKnownHostsFile /dev/null
  StrictHostKeyChecking no
  LogLevel quiet
  Port 2122

```

Run followings by root.

```

chmod 600 /root/.ssh/config
chown root:root /root/.ssh/config

chmod +x $HADOOP_CONF_DIR/*-env.sh

sed -i "/^#[^#]*UsePAM/ s/.*/#&/" /etc/ssh/sshd_config
echo "UsePAM no" >> /etc/ssh/sshd_config
echo "Port 2122" >> /etc/ssh/sshd_config

```

Restart SSH server.

```
service sshd restart
```

Run HDFS and Yarn daemons.

```
start-dfs.sh
start-yarn.sh
```

Test if Hadoop works fine.

```
hdfs dfs -mkdir -p /user/hadoop/input
hdfs dfs -put $HADOOP_PREFIX/LICENSE.txt /user/hadoop/input
hadoop jar $HADOOP_PREFIX/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.3.jar wordcount /user/
↪hadoop/input /user/hadoop/output
```

8.1.3 3. Install MySQL

```
wget http://dev.mysql.com/get/mysql57-community-release-el7-7.noarch.rpm \
&& yum -y localinstall mysql57-community-release-el7-7.noarch.rpm \
&& yum repolist enabled | grep "mysql.*-community.*" \
&& yum -y install mysql-community-server mysql \
&& rm -f mysql57-community-release-el7-7.noarch.rpm
service mysqld start
```

Get the temporary password with the following command.

```
grep 'temporary password' /var/log/mysqld.log | awk {'print $11'}
Z&0+estx9vTt
```

Run mysql_secure_installation with the temporary password.

```
mysql_secure_installation
Enter password for user root: -> Z&0+estx9vTt
New password: -> Metatron123$
Re-enter new password: -> Metatron123$
Change the password for root ? ((Press y|Y for Yes, any other key for No) : y
New password: -> Metatron123$
Re-enter new password: -> Metatron123$
Do you wish to continue with the password provided? -> y
Remove anonymous users? -> enter
```

(continues on next page)

(continued from previous page)

```
Disallow root login remotely? -> enter
Remove test database and access to it? -> enter
Reload privilege tables now? -> enter
```

Connect to MySQL.

```
mysql -uroot -pMetatron123$
```

8.1.4 4. Install Hive

```
wget http://mirror.navercorp.com/apache/hive/hive-2.3.6/apache-hive-2.3.6-bin.tar.gz \
&& tar -zvxf apache-hive-2.3.6-bin.tar.gz -C /opt \
&& rm -f apache-hive-2.3.6-bin.tar.gz \
&& ln -s /opt/apache-hive-2.3.6-bin /opt/hive
export HIVE_HOME=/opt/hive
export PATH=$PATH:$HIVE_HOME/bin:$HIVE_HOME/hcatalog/sbin
wget https://repo1.maven.org/maven2/mysql/mysql-connector-java/5.1.38/mysql-connector-java-5.1.38.jar
mv mysql-connector-java-5.1.38.jar $HIVE_HOME/lib/
```

Put files below into \$HIVE_HOME/conf.

```
hive-site.xml
```

Initialize the Hive metastore.

```
mysql -uroot -pMetatron123$
create database hive_metastore;
create user 'hive'@'%' identified by 'Metatron123$';
grant all privileges on *.* to 'hive'@'%';
grant all privileges on hive_metastore.* to 'hive'@'%';
create user 'hive'@'localhost' identified by 'Metatron123$';
grant all privileges on *.* to 'hive'@'localhost';
grant all privileges on hive_metastore.* to 'hive'@'localhost';
flush privileges;
quit
schematool -initSchema -dbType mysql
```

Start Hive.

```
hdfs dfs -mkdir -p /user/hive/warehouse
mkdir -p $HIVE_HOME/hcatalog/var/log
hcat_server.sh start
hiveserver2 &
```

Connect to Hive.

```
beeline -u jdbc:hive2://localhost:10000 "" ""
```

8.1.5 5. Install Druid

```
wget https://sktmetatronkrssouthshared.blob.core.windows.net/metatron-public/discovery-dist/latest/druid-
˓→0.9.1-latest-hadoop-2.7.3-bin.tar.gz
mkdir /servers
tar zxf druid-0.9.1-latest-hadoop-2.7.3-bin.tar.gz -C /servers
ln -s /servers/druid-* /servers/druid
export DRUID_HOME=/servers/druid
```

Put files below into each target locations.

Download URL	Target Location
jvm.config	\$DRUID_HOME/conf/druid/single/jvm.config
runtime.properties	\$DRUID_HOME/conf/druid/single/broker/runtime.properties
runtime.properties	\$DRUID_HOME/conf/druid/single/historical/runtime.properties
runtime.properties	\$DRUID_HOME/conf/druid/single/middleManager/runtime.properties

```
cd $DRUID_HOME
./start-single.sh
```

Check if you connect to <http://localhost:8090/>

8.1.6 6. Install Metatron

```
wget https://sktmetatronkrssouthshared.blob.core.windows.net/metatron-public/discovery-dist/latest/
˓→metatron-discovery-latest-bin.tar.gz
mkdir /servers
```

(continues on next page)

(continued from previous page)

```
tar zxf metatron-discovery-latest-bin.tar.gz -C /servers  
ln -s /servers/metatron-discovery-* /servers/metatron-discovery  
export METATRON_HOME=/servers/metatron-discovery
```

Put files below into \$METATRON_HOME/conf.

```
application-config.yaml metatron-env.sh logback-console.xml
```

Initialize Metatron.

```
mysql -uroot -pMetatron123$  
create database polaris;  
create user 'polaris'@'%' identified by 'Metatron123$';  
grant all privileges on *.* to 'polaris'@'%';  
grant all privileges on hive_metastore.* to 'polaris'@'%';  
create user 'polaris'@'localhost' identified by 'Metatron123$';  
grant all privileges on *.* to 'polaris'@'localhost';  
grant all privileges on hive_metastore.* to 'polaris'@'localhost';  
flush privileges;  
quit  
cd $METATRON_HOME  
bin/metatron.sh --init start
```

To watch the progress, tail the log file.

```
tail -f logs/metatron-* .out
```

Connect to <http://localhost:8180/>

8.1.7 7. Install Pretool

```
yum -y install https://centos7.iuscommunity.org/ius-release.rpm \  
&& yum install -y python36u python36u-libs python36u-devel python36u-pip git \  
&& ln -s /bin/python3.6 /bin/python3 \  
&& ln -s /bin/pip3.6 /bin/pip3 \  
&& pip3 install requests  
yum -y install git  
git clone https://github.com/metatron-app/discovery-prep-tool.git  
cd discovery-prep-tool
```

Download a test file.

```
sales-data-sample.csv
```

```
python3 pretool -f sales-data-sample.csv
```

If you get “File dataset created”, then it works.

8.2 Docker Migration Guide

This document is a guide on migrating a Metatron Discovery service across docker instances.

I suppose that you use <https://github.com/teamsprint/docker-metatron.git/> for convenience on docker commands. Refer to <https://metatron.app/2020/01/21/deploying-metatron-with-the-fully-engineered-docker-image/>

I assume that you use MySQL as the metadata store.

8.2.1 1. Stop Metatron Service

Run the following command to get into the docker instance.

```
git clone https://github.com/teamsprint/docker-metatron.git/
cd docker-metatron
./attach.sh
```

Stop Metatron service with the following command.

```
cd $METATRON_HOME
bin/metatron.sh stop
```

8.2.2 2. Backup Metadata Store

We must backup all metadata used in Metatron like datasets, dataflows, etc. Run the following commands from host machine. (Container name is “metatron”, the database name of metadata store is “polaris”.

```
sudo docker exec metatron /usr/bin/mysqldump -uroot -pMetatron123$ polaris > metadata_store_backup.sql
```

8.2.3 3. Backup configuration files and run scripts.

```
sudo docker cp metatron:/servers/metatron-discovery/conf/application-config.yaml .
sudo docker cp metatron:/servers/metatron-discovery/conf/metatron-env.sh .
sudo docker cp metatron:/servers/metatron-discovery/conf/logback-console.sh .
sudo docker cp metatron:/servers/metatron-discovery/bin/metatron.sh .
sudo docker cp metatron:/servers/metatron-discovery/bin/common.sh .
```

8.2.4 4. Backup uploaded file datasets, data snapshots.

```
sudo docker cp metatron:/servers/metatron-discovery/dataprep/uploads .
sudo docker cp metatron:/servers/metatron-discovery/dataprep/snapshots .
```

Generally, you don't need to backup data snapshots. If the snapshot is small enough, you can easily remake the snapshots. Or if it's too big, backup size might be also too big.

By the way, you cannot backup snapshots stored in internal databases. If you didn't modify configurations about staging DB (if the configuration is the default of initial image), then you cannot backup staging DB type snapshots.

8.2.5 5. Remove old docker instance

Run the following commands to remove the old docker instance.

```
./destroy.sh
```

8.2.6 6. Run new docker instance

Run the following commands from the host machine.

```
./run.sh
```

In case you patch the binary, you might need to edit run.sh to modify IMAGE_NAME.

Run the following commands inside the docker instance.

```
./prepare-all-metatron.sh
```

Normally, Metatron service will be ready in about 2~3 minutes. Check the service running, then shutdown right after. Let's start to restore.

```
./stop-metatron.sh
```

8.2.7 7. Restore Metadata Store

Run the following commands from the host machine.

```
cat metadata_store_backup.sql | sudo docker exec -i metatron /usr/bin/mysql -uroot -pMetatron123$  
↳ polaris
```

8.2.8 8. Restore configurations, run scripts

Run the following commands from the host machine. In case you patch the binary, you should apply the changes to the corresponding files.

```
sudo docker cp application-config.yaml metatron:/servers/metatron-discovery/conf/  
sudo docker cp metatron-env.sh metatron:/servers/metatron-discovery/conf/  
sudo docker cp logback-console.sh metatron:/servers/metatron-discovery/conf/  
sudo docker cp metatron.sh metatron:/servers/metatron-discovery/bin/  
sudo docker cp common.sh metatron:/servers/metatron-discovery/bin/
```

8.2.9 9. Restore file datasets and snapshots

Run the following commands from the host machine.

```
sudo docker exec metatron mkdir -p /servers/metatron-discovery/dataprep  
sudo docker cp uploads metatron:/servers/metatron-discovery/dataprep/  
sudo docker cp snapshots metatron:/servers/metatron-discovery/dataprep/
```

8.2.10 10. Start New Metatron Service

Run the following commands from the host machine.

```
./attach.sh
```

Run the following commands inside the docker instance.

```
./start-metatron.sh
```

Generally, Metatron service will be ready in about 1~2 minutes.

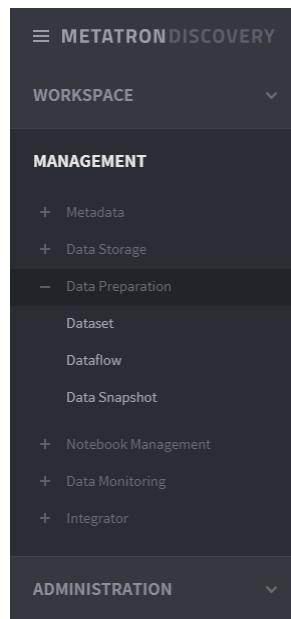
8.3 Create a dataset

A **dataset**, which is the basic unit of data preparation, refers to an entity subject to data operations. Datasets are either **imported datasets** and **wrangle datasets**.

- **Imported Dataset:** A source data entity before the implementation of transformation rules
- **Wrangled dataset:** A data entity subject to analysis following the implementation of transformation rules

A wrangled dataset is created during the **dataflow** setting process, which defines transformation rules, while an imported dataset is created during this dataset creation procedure.

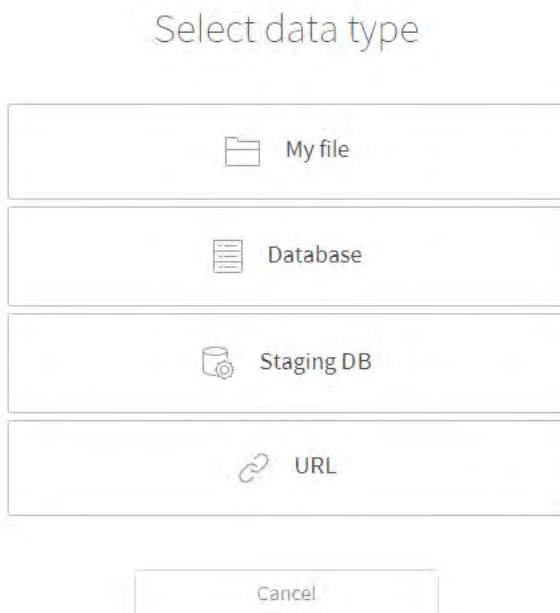
The Dataflow menu can be accessed under **MANAGEMENT** > **Data Preparation** > **Dataset** on the left-hand panel of the main screen.



Next, on the upper right of the **dataset** page, click the **+ Generate new dataset** button to create a new dataset.

The screenshot shows the 'Data Preparation' section of the Metatron Discovery interface. At the top, there are three tabs: 'Dataset' (selected), 'Dataflow', and 'Data Snapshot'. Below the tabs is a search bar with placeholder text 'Search by name or dataset...'. To the right of the search bar are two filter buttons: 'Type' with a checked checkbox labeled 'IMPORTED' and an unchecked checkbox labeled 'WRANGLED'. Further to the right, it says 'There are 48 lists...' and a '+ Generate new dataset' button. The main area is currently empty, indicating no datasets have been created yet.

In the dataset creation page, select the dataset type.



- **My file:** Create a dataset by opening the user's local file or via a URL (upcoming feature) (See [Create a dataset from a file](#) for a detailed procedure).
- **Database:** Create a dataset using external database access information and queries (See [Create a dataset from a database](#) for a detailed procedure).

- **Staging DB:** Create a dataset from the staging DB built in Metatron (See [Create a dataset from staging DB](#) for a detailed procedure).

Note: The Staging DB is an in-cluster database that stores data temporarily in order to facilitate data loading. Hive is generally used for it.

8.3.1 Create a dataset from a file

Create a dataset by opening the user's local file or via a URI (upcoming feature).

1. On the data type selection page, select **My File**.
2. Select a file to be used as a data source from your local PC. You can click the **Import** button to select a file, or drag and drop the file into the box. Once a file is selected, click Next.

Create file type dataset

Please select data

Import or drop file here

xls, .xlsx, .txt, .csv and .json formats are allowed

Upload Location LOCAL

Cancel Next

This screenshot shows the 'Create file type dataset' interface. At the top, the title 'Create file type dataset' is displayed in blue, followed by the instruction 'Please select data'. Below this is a progress bar consisting of three circles, with the first circle filled. The main area features a large, light-gray rectangular placeholder with the text 'Import or drop file here' and a note that 'xls, .xlsx, .txt, .csv and .json formats are allowed'. Below this placeholder is a 'Upload Location' field containing the value 'LOCAL', which is part of a dropdown menu. At the bottom of the interface are two buttons: 'Cancel' on the left and 'Next' on the right, with the 'Next' button being highlighted.

3. Check the grid of the uploaded file, and designate a column delimiter. Proceed if the data is successfully displayed.

Create file type dataset
Please select data

sales-data-sample.csv

	OrderDate	Category	City	Country	CustomerName	#
2011-01-04T00:00:00.000Z	Office·Supplies	Houston	United·States	Darren·Powers		
2011-01-05T00:00:00.000Z	Office·Supplies	Naperville	United·States	Phillina·Ober		
2011-01-05T00:00:00.000Z	Office·Supplies	Naperville	United·States	Phillina·Ober		
2011-01-05T00:00:00.000Z	Office·Supplies	Naperville	United·States	Phillina·Ober		
2011-01-06T00:00:00.000Z	Office·Supplies	Philadelp...	United·States	Mick·Brown		
2011-01-07T00:00:00.000Z	Furniture	Henderson	United·States	Maria·Etezadi		
2011-01-07T00:00:00.000Z	Office·Supplies	Athens	United·States	Jack·OBriant		
2011-01-07T00:00:00.000Z	Office·Supplies	Henderson	United·States	Maria·Etezadi		
2011-01-07T00:00:00.000Z	Office·Supplies	Henderson	United·States	Maria·Etezadi		
2011-01-07T00:00:00.000Z	Office·Supplies	Henderson	United·States	Maria·Etezadi		
2011-01-07T00:00:00.000Z	Office·Supplies	Los·Ange...	United·States	Lycoris·Saunders		
2011-01-07T00:00:00.000Z	Technology	Henderson	United·States	Maria·Etezadi		
2011-01-07T00:00:00.000Z	Technology	Henderson	United·States	Maria·Etezadi		
2011-01-08T00:00:00.000Z	Furniture	Huntsville	United·States	Vivek·Sundaresam		
2011-01-08T00:00:00.000Z	Office·Supplies	Huntsville	United·States	Vivek·Sundaresam		
2011-01-10T00:00:00.000Z	Office·Supplies	Laredo	United·States	Melanie·Seite		
2011-01-10T00:00:00.000Z	Technology	Laredo	United·States	Melanie·Seite		
2011-01-11T00:00:00.000Z	Furniture	Springfield	United·States	Anthony·Jacobs		

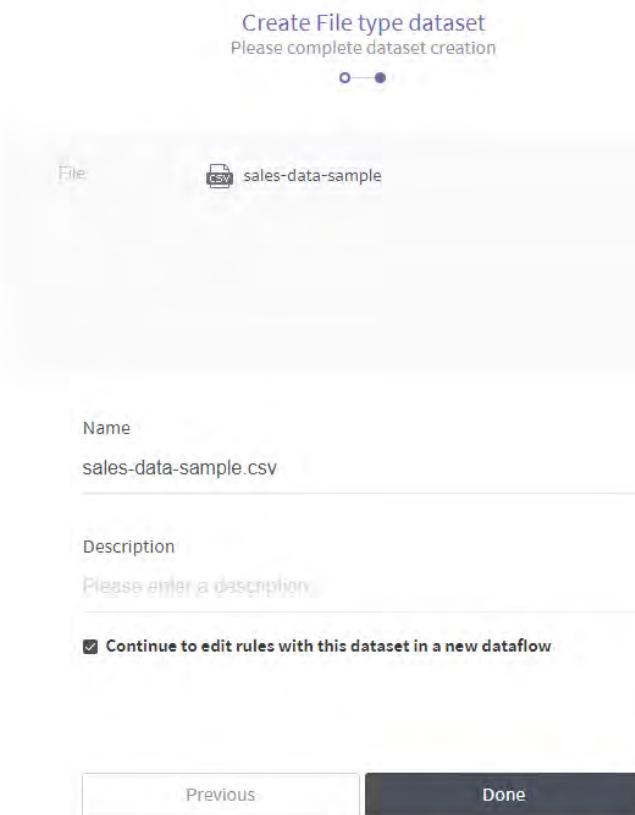
Advanced settings ▲

Column delimiter

Column count

Previous Next

4. Enter the **Name** and **Description** of the dataset, and click the **Done** button.



- Once the dataset is created, the dataset list is displayed. You can check that the list contains the newly created dataset.

≡ METATRON DISCOVERY

Data Preparation

Dataset Dataflow Data Snapshot

Name	Used in	Source	Created
IMPORTED sales-data-sample.csv	1	FILE(CSV)	2019-05-06 18:41 by Administrator

8.3.2 Create a dataset from a database

Create a dataset using external database access information and queries.

To create a dataset from a database, you should first create a data connection. See [Create a data connection](#) for a detailed procedure.

Data Storage

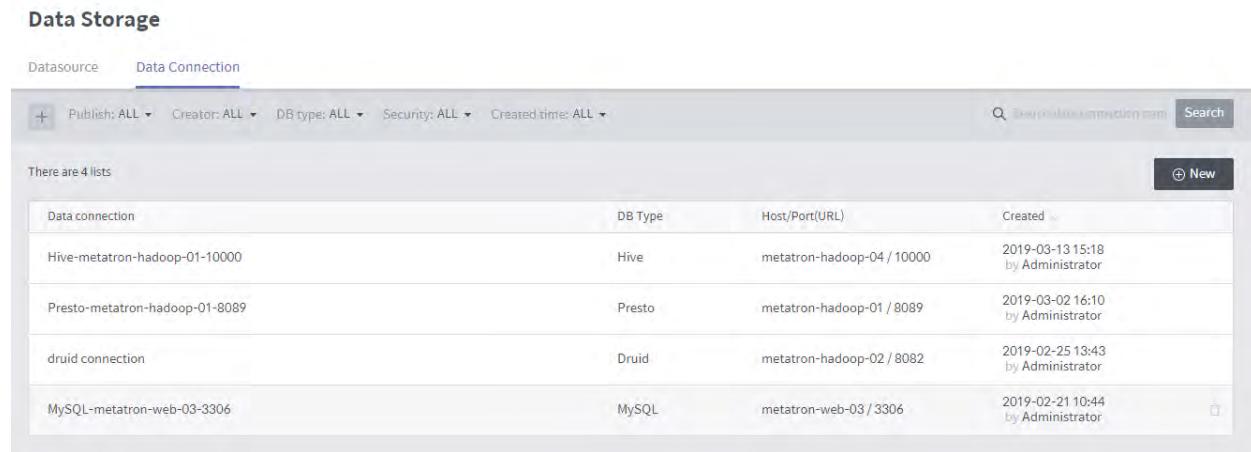
Datasource Data Connection

+ Publish: ALL Creator: ALL DB type: ALL Security: ALL Created time: ALL

There are 4 lists

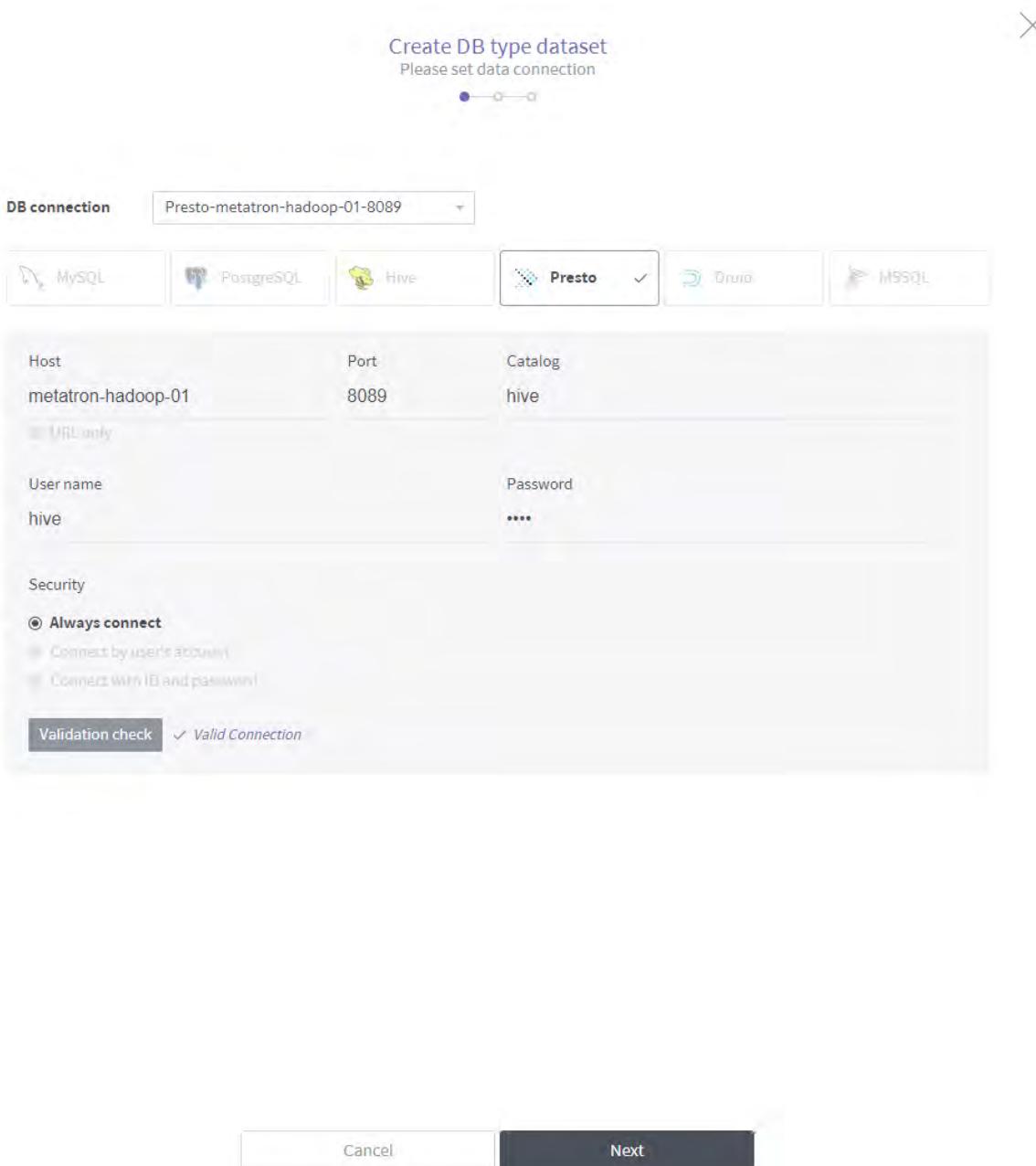
Data connection	DB Type	Host/Port(URL)	Created
Hive-metatron-hadoop-01-10000	Hive	metatron-hadoop-04 / 10000	2019-03-13 15:18 by Administrator
Presto-metatron-hadoop-01-8089	Presto	metatron-hadoop-01 / 8089	2019-03-02 16:10 by Administrator
druid connection	Druid	metatron-hadoop-02 / 8082	2019-02-25 13:43 by Administrator
MySQL-metatron-web-03-3306	MySQL	metatron-web-03 / 3306	2019-02-21 10:44 by Administrator

New

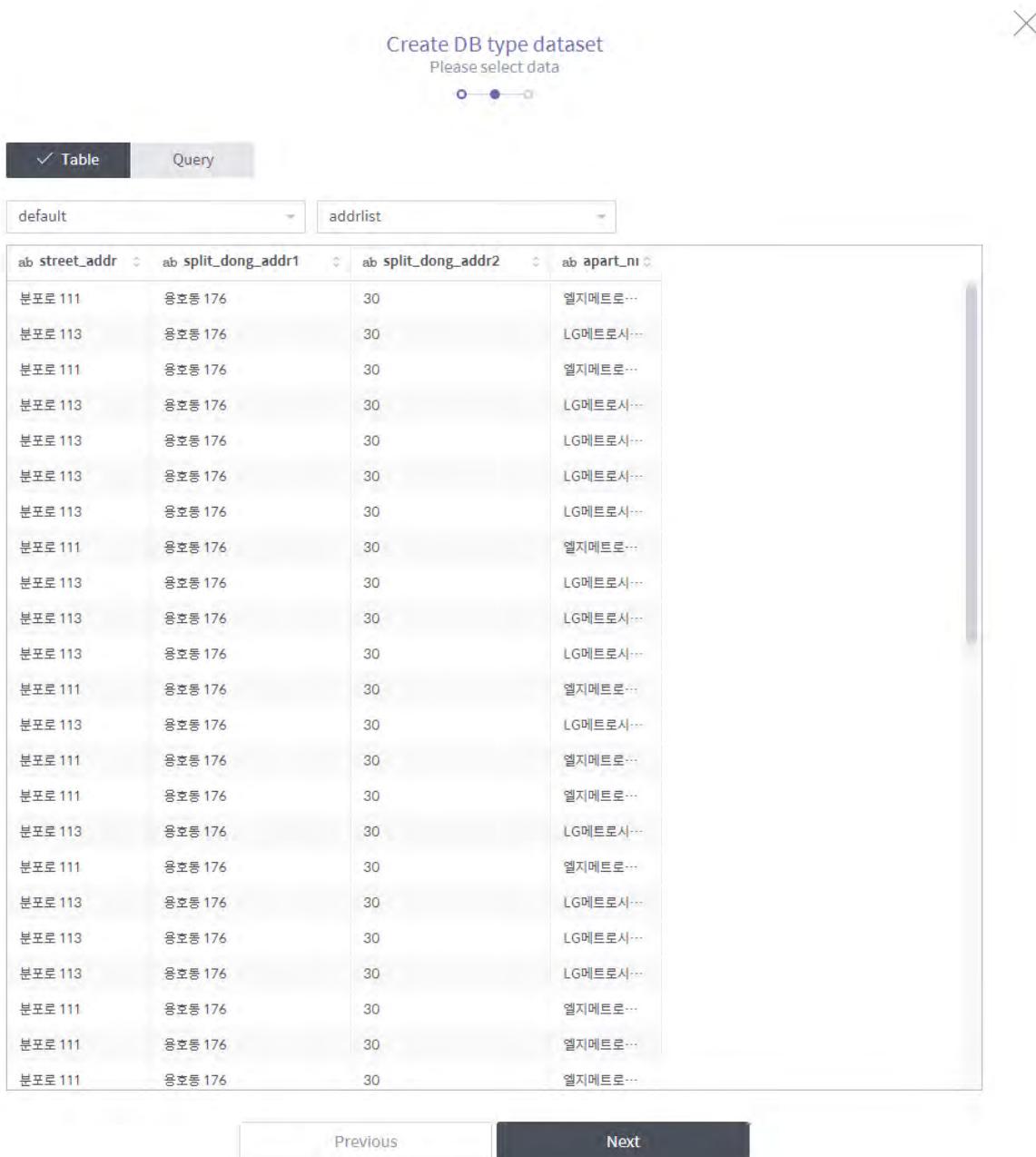


After establishing the data connection, go to **MANAGEMENT** > **Data Preparation** > **Dataset** > **+ Generate new dataset**.

1. On the data type selection page, select **Database**.
2. Select the data connection, and press the **Test** button to check that the connection is valid.



3. Select the data. You can either select a table from the connected database, or write a query yourself.



- **Table:** Select a database and a table to display the table's data. Once the data being ingested has been displayed, confirm the data and click **Next**.
- **Query:** Write a query to import the data you want, and click **Run** to display the data in the lower section. Confirm the data and click **Next**.

4. Enter the **Name** and **Description** of the dataset, and click the **Done** button.

X

Create Database type dataset
Please complete dataset creation

Type: DB(PRESTO)

Database: default

Table: addrlist

Host: metatron-hadoop-01

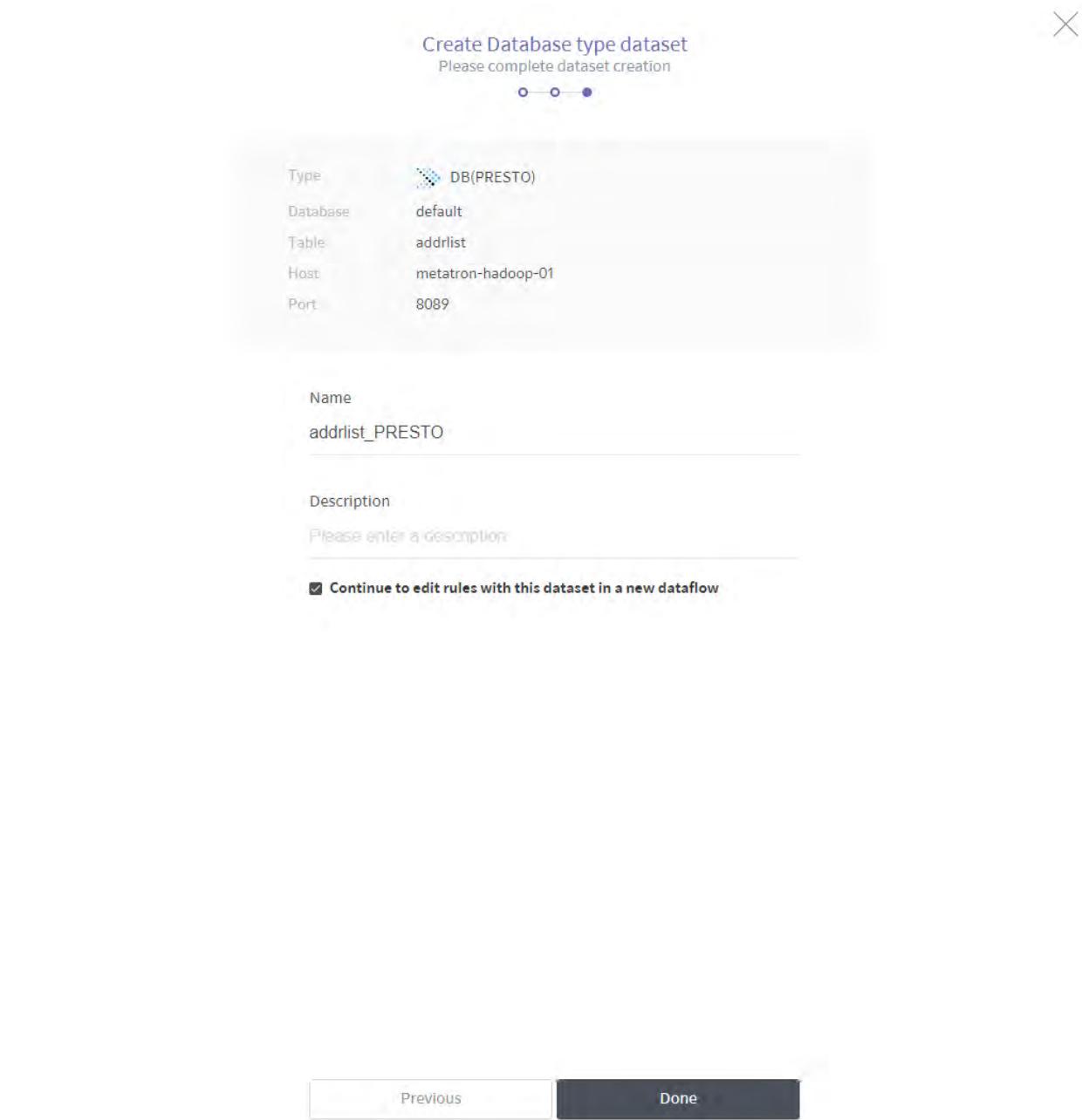
Port: 8089

Name: addrlist_PRESTO

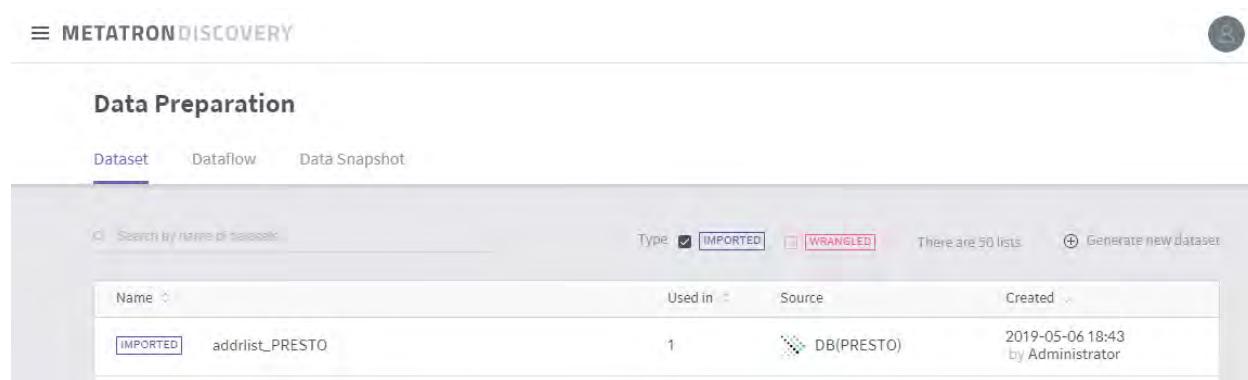
Description: Please enter a description.

Continue to edit rules with this dataset in a new dataflow

Previous Done



- Once the dataset is created, the dataset list is displayed. You can check that the list contains the newly created dataset.



The screenshot shows the Metatron Discovery interface with the title 'METATRON DISCOVERY'. Under 'Data Preparation', the 'Dataset' tab is selected. A search bar at the top left contains 'Search by name or dataset...'. To its right are filters for 'Type' (checkboxes for 'IMPORTED' and 'WRANGLER'), a note 'There are 50 lists.', and a button 'Generate new dataset'. Below the search bar is a table with columns: 'Name', 'Used in', 'Source', and 'Created'. The table has one row: 'addrlist_PRESTO' (IMPORTED), '1', 'DB(PRESTO)', and '2019-05-06 18:43 by Administrator'.

8.3.3 Create a dataset from staging DB

Create a dataset from the staging DB built in Metatron.

The creation of a staging DB dataset is the same as dataset creation from a database, but does not involve the selection of a data connection.

1. On the data type selection page, select **Staging DB**.
2. Select the data. You can either select a table from the connected database, or write a query yourself.

Create Staging DB type dataset

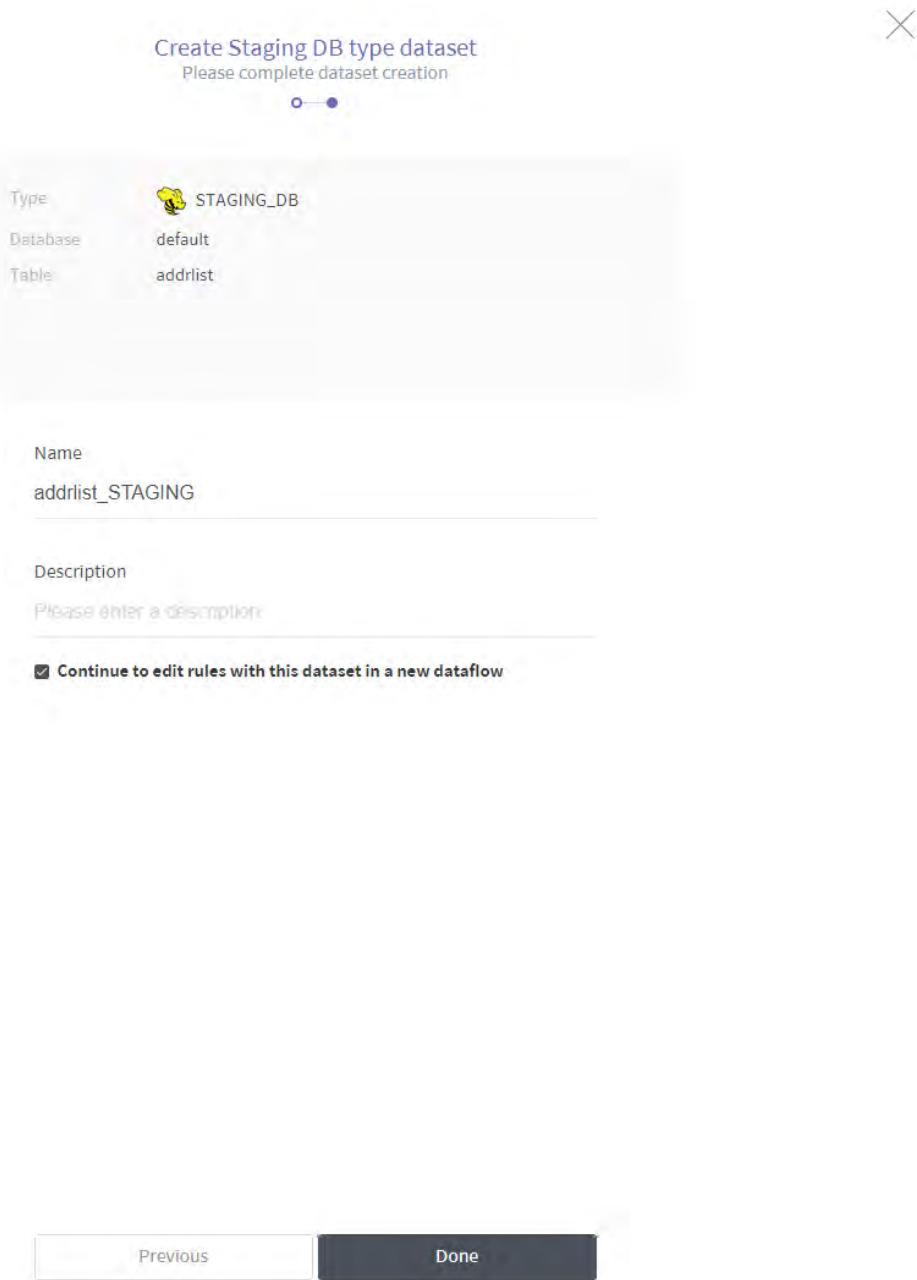
Please select data

Table Query

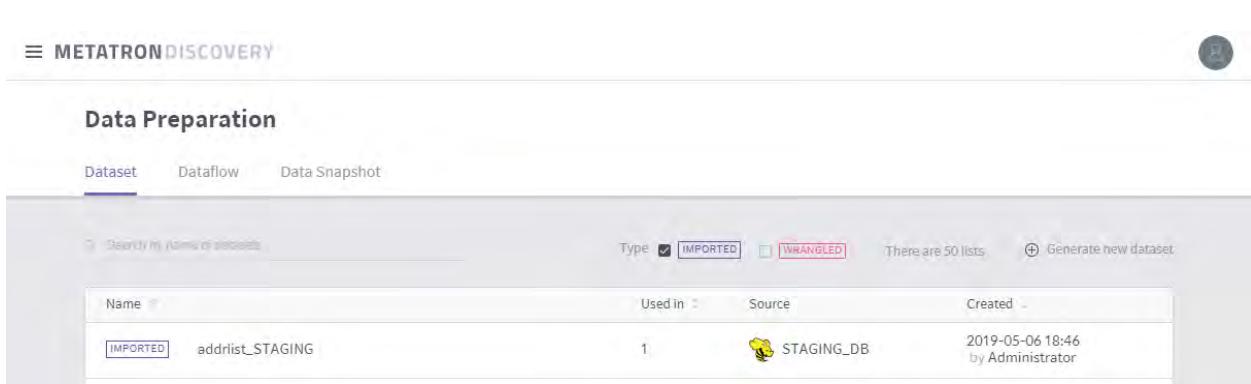
ab_addrlist.street_addr	ab_addrlist.split_dong_addr1	ab_addrlist.split_dong_addr2	ab_addrlist.apart_
분포로 111	옹호동 176	30	엘지메트로시티(3)
분포로 113	옹호동 176	30	LG메트로시티4
분포로 111	옹호동 176	30	엘지메트로시티2
분포로 113	옹호동 176	30	LG메트로시티5
분포로 113	옹호동 176	30	LG메트로시티4
분포로 113	옹호동 176	30	LG메트로시티4
분포로 113	옹호동 176	30	LG메트로시티4-2
분포로 111	옹호동 176	30	엘지메트로시티(3)
분포로 113	옹호동 176	30	LG메트로시티4
분포로 113	옹호동 176	30	LG메트로시티5
분포로 113	옹호동 176	30	LG메트로시티4-2
분포로 111	옹호동 176	30	엘지메트로시티(3)
분포로 113	옹호동 176	30	LG메트로시티4
분포로 111	옹호동 176	30	엘지메트로시티1
분포로 111	옹호동 176	30	엘지메트로시티1
분포로 113	옹호동 176	30	LG메트로시티4
분포로 111	옹호동 176	30	엘지메트로시티1
분포로 113	옹호동 176	30	LG메트로시티4
분포로 113	옹호동 176	30	LG메트로시티4
분포로 111	옹호동 176	30	엘지메트로시티1
분포로 111	옹호동 176	30	엘지메트로시티(3)

Cancel Next

- **Table:** Select a database and a table to display the table's data. Once the data being ingested has been displayed, confirm the data and click **Next**.
 - **Query:** Write a query to import the data you want, and click **Run** to display the data in the lower section. Confirm the data and click **Next**.
3. Enter the **Name** and **Description** of the dataset, and click the **Done** button.



4. Once the dataset is created, the dataset list is displayed. You can check that the list contains the newly created dataset.



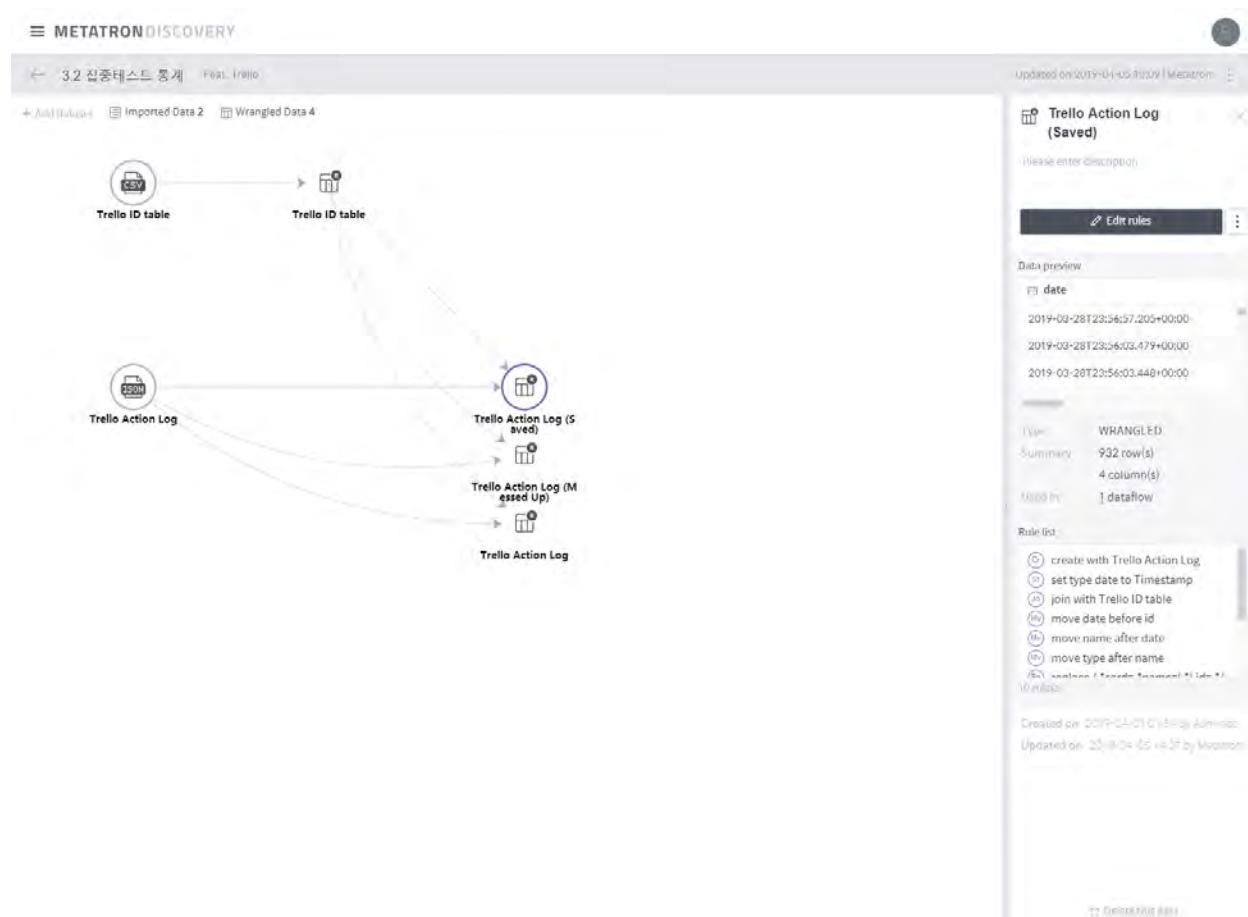
The screenshot shows the METATRON DISCOVERY interface under the 'Data Preparation' section. The 'Dataset' tab is selected. At the top, there are filters for 'Search by name or datasets' and 'Type' (with 'IMPORTED' checked and 'WRANGLLED' unselected). It also indicates 'There are 50 lists' and a 'Generate new dataset' button. A table below lists datasets, with one entry shown:

Name	Used in	Source	Created
addrlist_STAGING	1	STAGING_DB	2019-05-06 18:46 by Administrator

8.4 Manage a dataflow

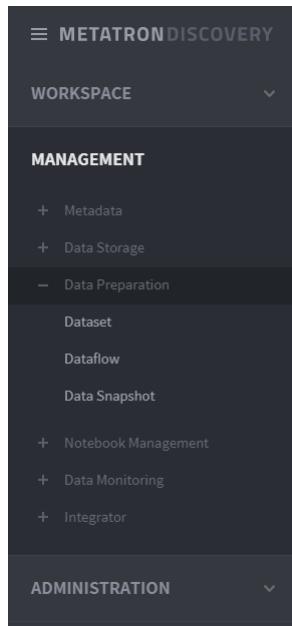
A **dataflow** is the unit of processing a **dataset**. A single dataflow can be associated with multiple datasets to perform transformations. That is, a dataset must belong to a dataflow for transformation rules to be applied. It forms a relationship such as a “join” or “union” with other datasets.

As shown below, the dataflow details page shows the dependency among all datasets in a dataflow, and the transformation rules applied to each dataset.



The following subsections cover the processes involved in defining a dataflow, such as **adding a dataset**, **editing transformation rules**, and **creating a data snapshot with transformation results**.

The Dataflow menu can be accessed under **MANAGEMENT** > **Data Preparation** > **Dataflow** on the left-hand panel of the main screen.



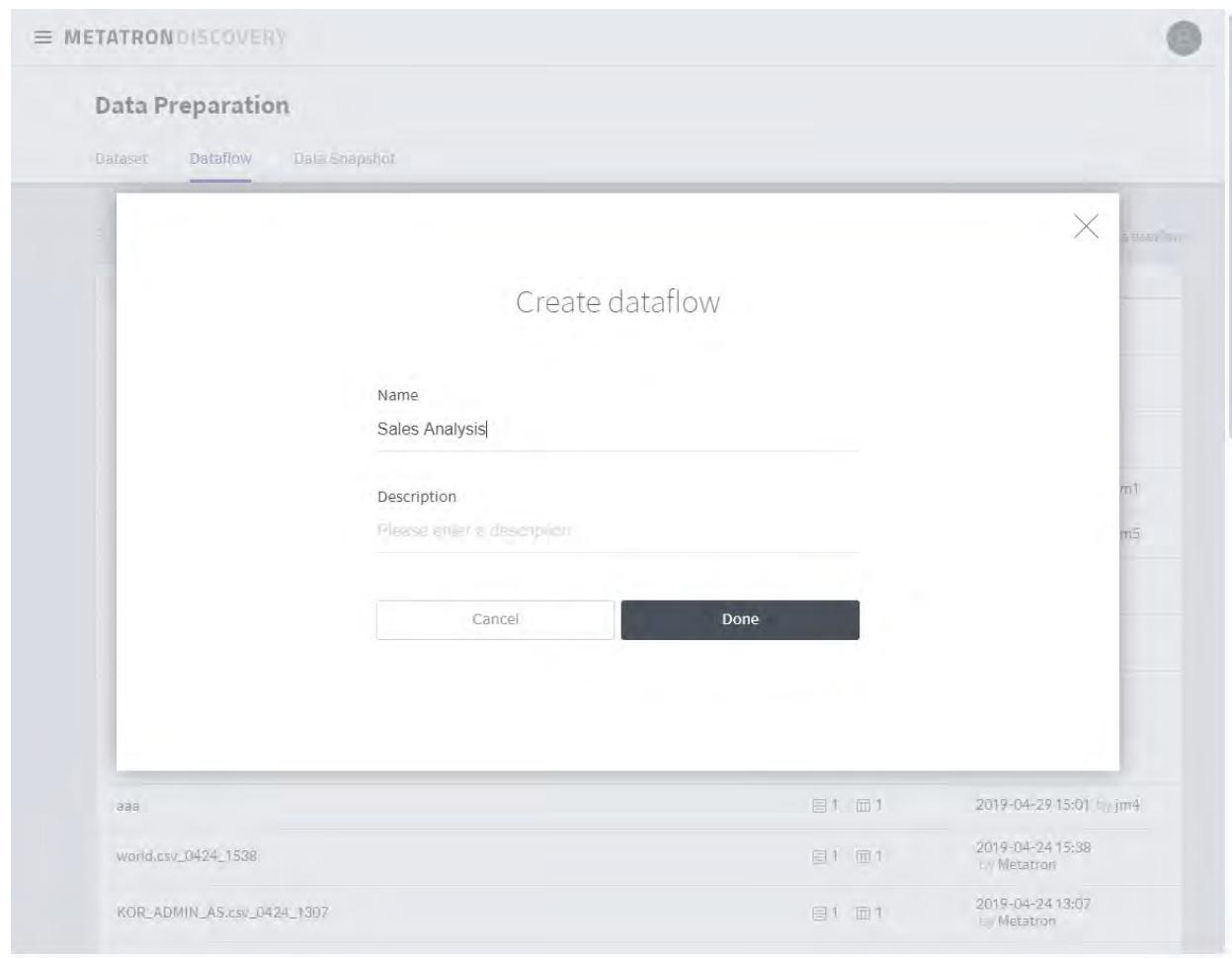
8.4.1 Add a dataset

The first step in defining a dataflow is to add a dataset. This can be conducted using the two methods described below:

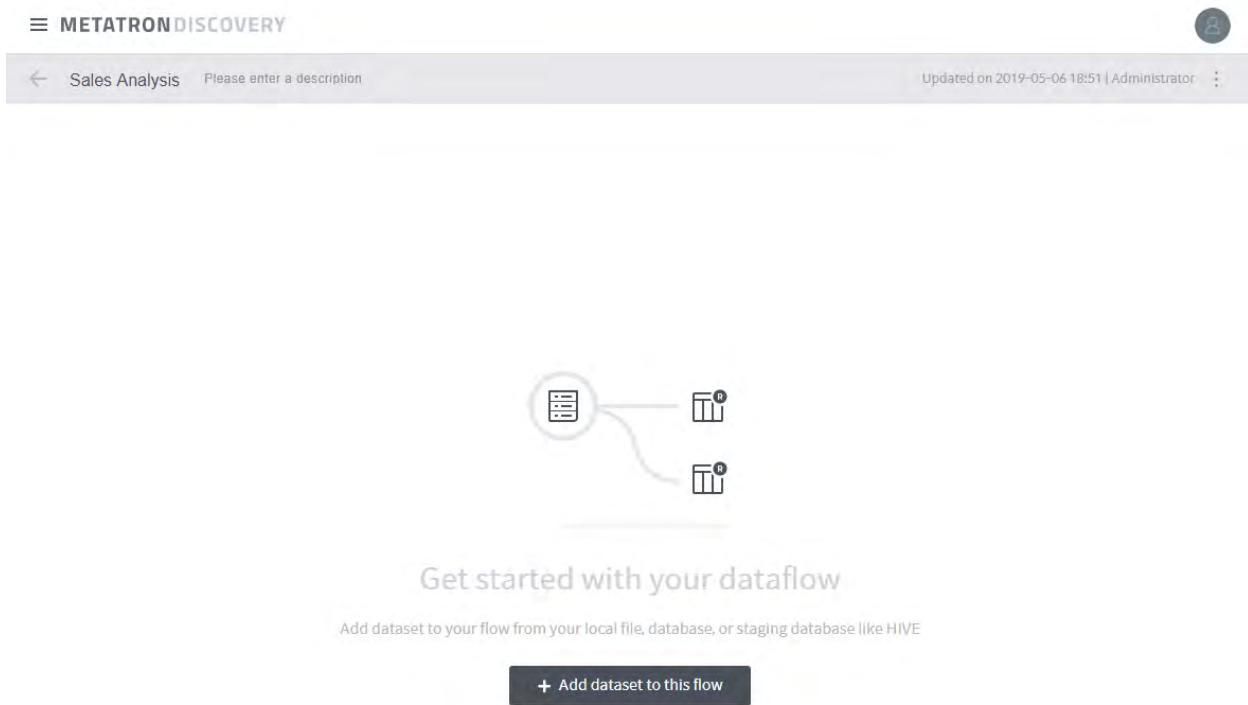
- Adding a dataset after creating an empty dataflow
- Creating a dataflow in the dataset details page

Adding a dataset after creating an empty dataflow

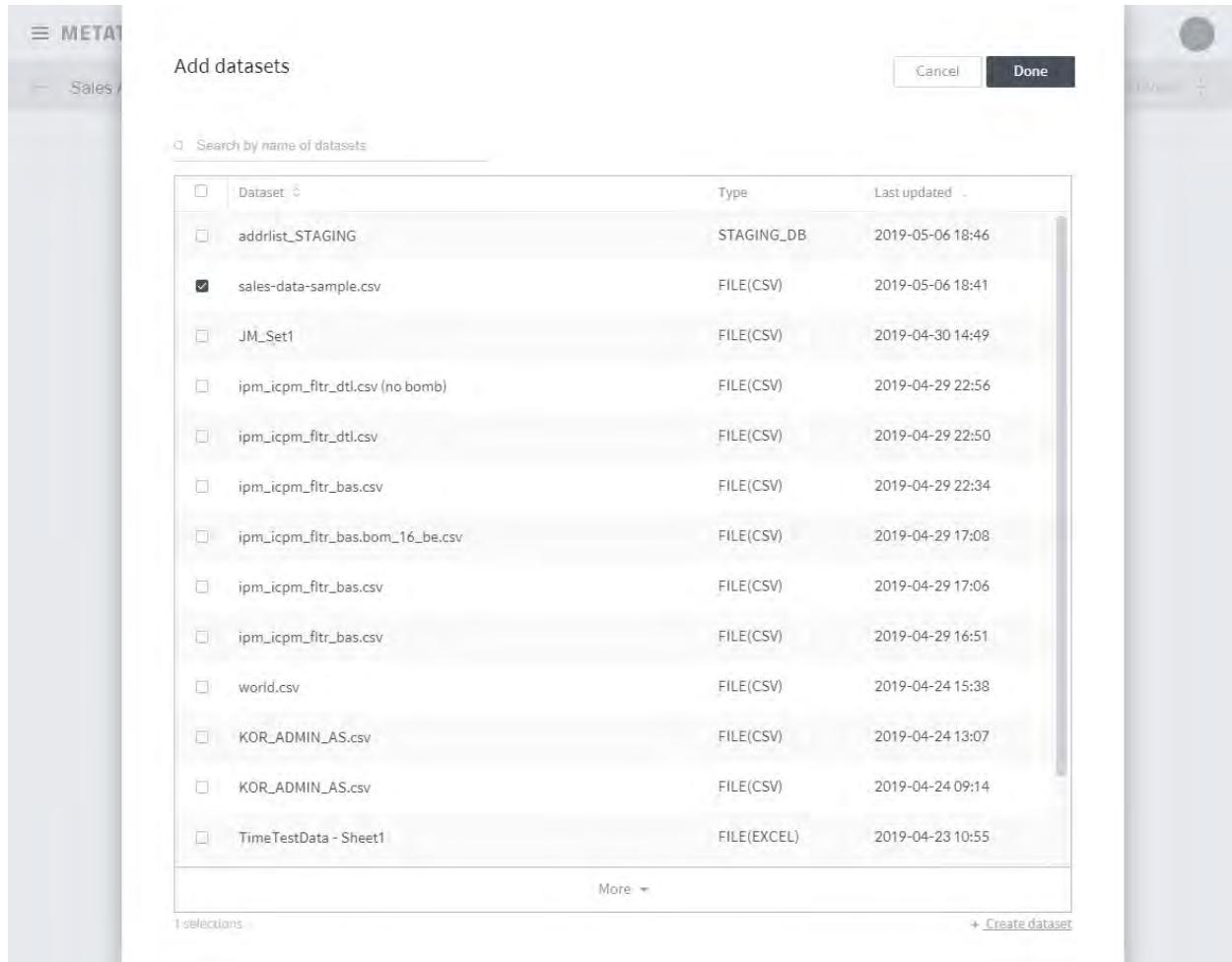
1. Click **Add a dataflow** on the upper right of the **Dataflow** page.
2. Enter the **Name** and **Description** for the dataflow, and click **Done** to create an empty dataflow.



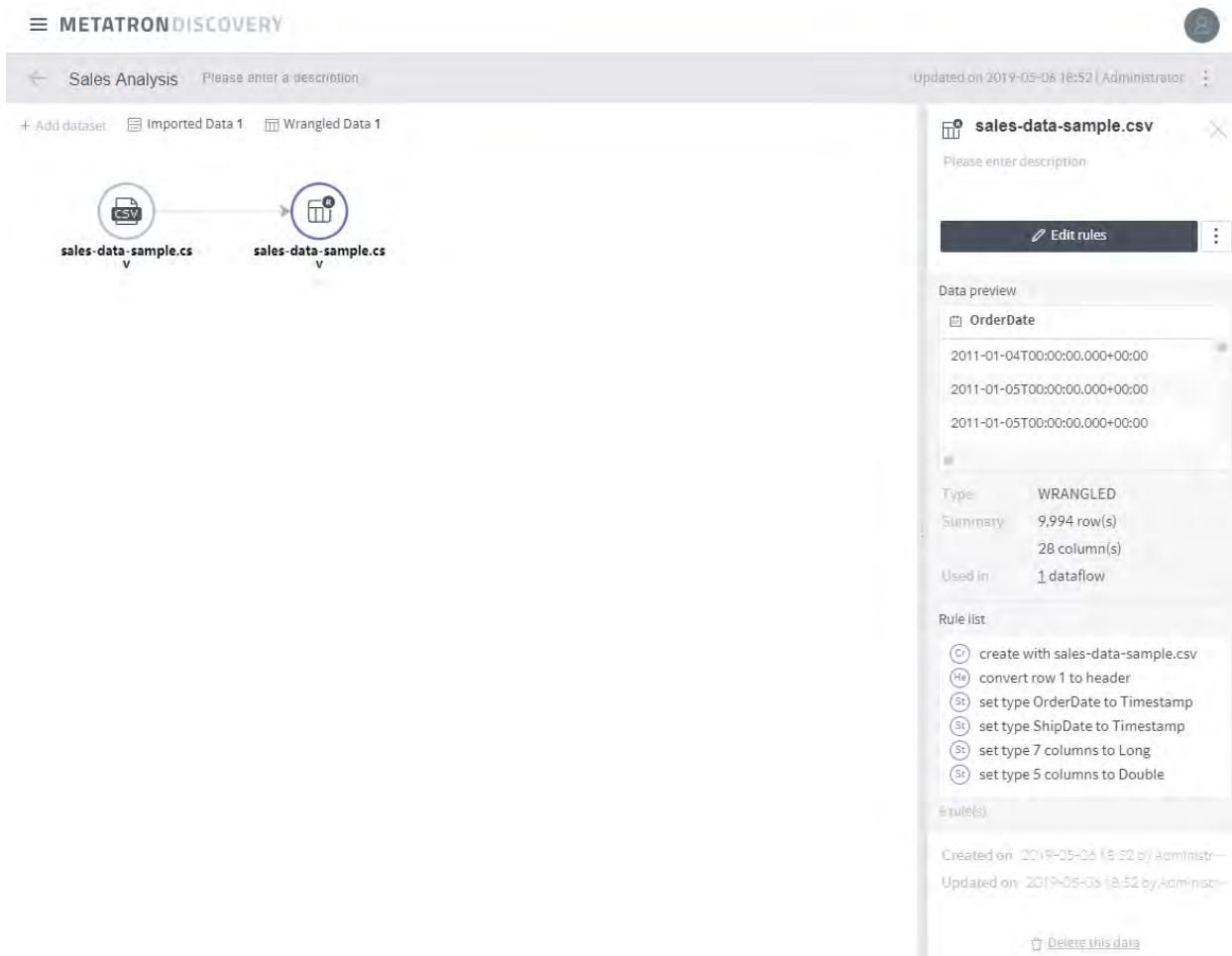
3. Click the **Add dataset to this dataflow** button on the center of the page.



4. Select the datasets to be added.



5. When an imported dataset and its corresponding wrangled dataset are created, click the **Edit rules** button to edit rules (see [Edit rules](#) for a detailed procedure).



Creating a dataflow in the dataset details page

In the dataset details page, click the **Create dataflow with this dataset** button to create a dataflow, and proceed until the step before **Edit rules**.

The screenshot shows the Metatron Discovery interface. At the top, there's a navigation bar with a menu icon, the text 'METATRON DISCOVERY', and a user profile icon. Below the header, the dataset name 'sales-data-sample.csv' is displayed along with a placeholder 'Please enter a description'. A timestamp 'Updated on 2019-05-06 18:41 | UNKNOWN_USER' and a three-dot menu icon are also present.

The main area is divided into two sections: 'Information' on the left and 'Data' on the right.

Information Section:

Type	FILE(CSV)
File	sales-data-sample.csv
URI	file:///data/metatron-discovery/dataprep/uploads/73375...
Size	3.2 MB
Summary	9,995 row(s) 28 column(s)

Data Section:

OrderDate	Category	City
2011-01-04T00:00:00.000+0:00	Office·Supplies	Houston
2011-01-05T00:00:00.000+0:00	Office·Supplies	Naperville
2011-01-05T00:00:00.000+0:00	Office·Supplies	Naperville
2011-01-05T00:00:00.000+0:00	Office·Supplies	Naperville
2011-01-06T00:00:00.000+0:00	Office·Supplies	Philadelphia
2011-01-07T00:00:00.000+0:00	Furniture	Henderson
2011-01-07T00:00:00.000+0:00	Office·Supplies	Athens
2011-01-07T00:00:00.000+0:00	Office·Supplies	Henderson
2011-01-07T00:00:00.000+0:00	Office·Supplies	Henderson

Used in Section:

Created in: Sales Analysis Add to existing dataflow Create dataflow with this dataset

1+ Updated on 2019-05-06 18:52 | admin

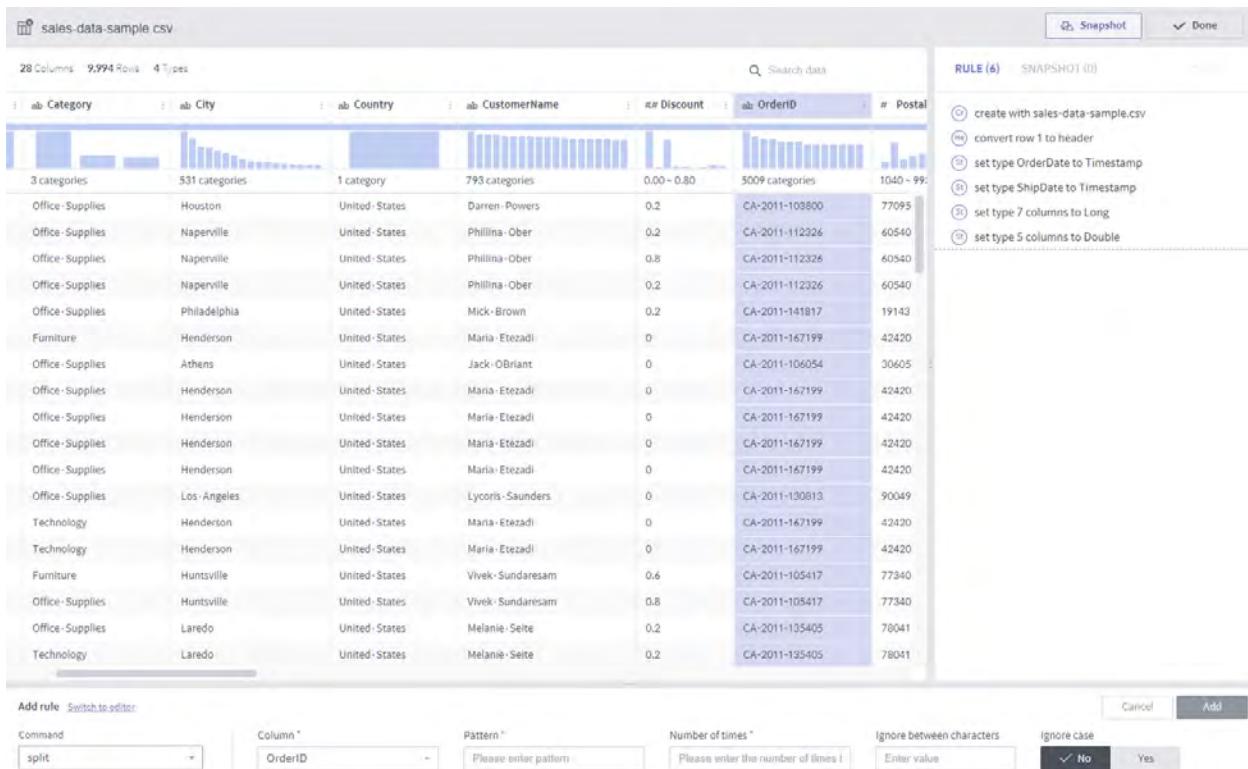
Note: The dataflow is named based on the name of the dataset.

8.4.2 Edit rules

The key task in data preparation is to create rules for data transformation (usually refinement). The transformation rules and input/output specifications are combined to be applied to actual data or other similar data, or scheduling is performed for such tasks.

Below are instructions on creating rules, checking the results, and modifying or deleting rules.

The Edit Rules page consists of the following:



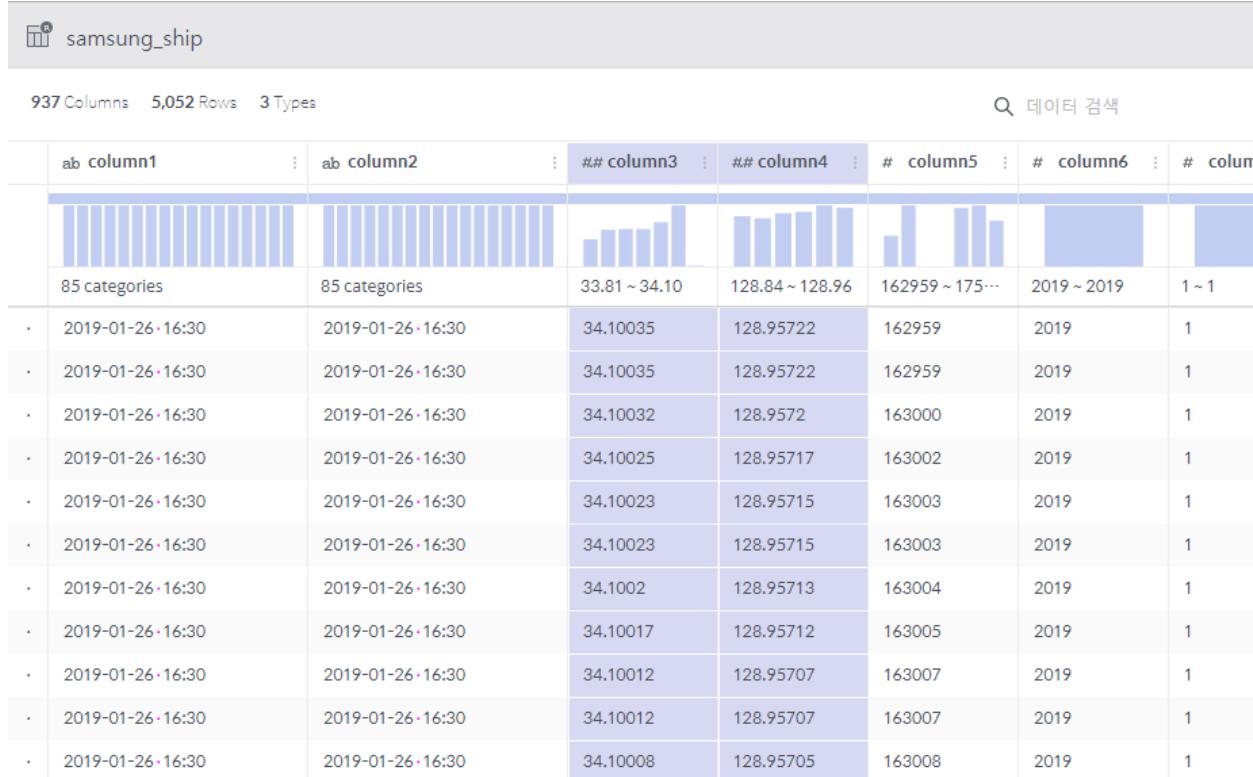
1. Column type, name, and menu button
2. Menu for simple rule creation
3. Rule list and insert button (appears when cursor is placed in between rules)
4. Enabled when undo and redo are available
5. Panel to enter rule details
6. Column value distribution, distinct count, type mismatch, null value, etc.

Create a rule

Using the column header menu

1. Select a target column by clicking the column header.
 - Press the function key to select multiple columns.

- Depending on your OS, click while holding the ^ or key to select/deselect a column (toggle).
- Click while holding the Shift key to select a range.



The screenshot shows a data editor window with the following details:

- Title Bar:** samsung_ship
- Header:** 937 Columns, 5,052 Rows, 3 Types, 데이터 검색 (Search)
- Table Structure:**
 - Columns: ab column1, ab column2, ## column3, ## column4, # column5, # column6, # colum
 - Rows: 85 categories (repeated), dates (e.g., 2019-01-26 16:30), and numerical values (e.g., 34.10035, 128.95722, 162959).

2. Click the  icon in the header of a selected column to open the header menu, and select a transformation command.

- Among the commands, **drop** and **settype** are performed upon clicking.

The screenshot shows a data grid titled "samsung_ship" with 937 columns, 5,052 rows, and 3 types. A context menu is open over the fourth column, showing options like Drop, Alter, Edit, Generate, Sort, Move, and Clear. The data in the grid includes various dates and numerical values.

ab column1	ab column2	## column3	## column4	## column5	# column6	# column7
85 categories 2017-01-20 10:37	85 categories 2017-01-20 10:37	33.81 ~ 34.10				
2019-01-26 16:30	2019-01-26 16:30	34.10035				
2019-01-26 16:30	2019-01-26 16:30	34.10035				
2019-01-26 16:30	2019-01-26 16:30	34.10032				
2019-01-26 16:30	2019-01-26 16:30	34.10025	128.95717	163002	2019	1
2019-01-26 16:30	2019-01-26 16:30	34.10023	128.95715	163003	2019	1
2019-01-26 16:30	2019-01-26 16:30	34.10023	128.95715	163003	2019	1
2019-01-26 16:30	2019-01-26 16:30	34.1002	128.95713	163004	2019	1
2019-01-26 16:30	2019-01-26 16:30	34.10017	128.95712	163005	2019	1
2019-01-26 16:30	2019-01-26 16:30	34.10012	128.95707	163007	2019	1
2019-01-26 16:30	2019-01-26 16:30	34.10012	128.95707	163007	2019	1
2019-01-26 16:30	2019-01-26 16:30	34.10008	128.95705	163008	2019	1

3. To add details, fill out the command input panel below, and click the Add button.



4. Some commands can be performed by selecting a distribution bar.

- Click a distribution bar to filter the data based on the selected range (toggle).
- Click the type mismatch or null value graph to set conditions for those values.

The screenshot shows a data grid with a red box highlighting the distribution bar for the third column. The data in the grid includes various dates and numerical values.

ab column1	ab column2	ab column3	ab column4	ab column5	ab column6	ab column7
85 categories 2017-01-20 10:37	85 categories 2017-01-20 10:37	4265 categories 34.00172	4254 categories 120.74032	4177 categories 103744	1 category 2017	1 category 1
2019-01-26 16:39	2019-01-26 16:39	34.08185	128.94647	163946	2019	1
2019-01-26 16:39	2019-01-26 16:39	34.08182	128.94645	163948	2019	1
2019-01-26 16:39	2019-01-26 16:39	34.08178	128.94643	163949	2019	1
2019-01-26 16:39	2019-01-26 16:39	34.08178	128.94643	163950	2019	1

Using the command input panel

- Select a transformation rule (command) in the command input panel.

The screenshot shows a dialog box with a sidebar containing a list of transformation commands:

- header**: Specifies the value of the specified row as the field name.
- keep**: Only rows that satisfy the condition are kept.
- replace**: Replace values that match the pattern in a particular column with new values.
- rename**: Rename a column title.
- set**: Replace the value of a particular column with the result of the formula.

Below the sidebar is a preview table with the following data:

	Office-Supplies	Henderson	United-States	Maria-Etezadi	0
Office-Supplies	Los-Angeles	United-States	Lycoris-Saunders	0	
Technology	Henderson	United-States	Maria-Etezadi	0	
Technology	Henderson	United-States	Maria-Etezadi	0	
Furniture	Huntsville	United-States	Vivek-Sundaresam	0.6	
Office-Supplies	Huntsville	United-States	Vivek-Sundaresam	0.8	
Office-Supplies	Laredo	United-States	Melanie-Seite	0.2	
Technology	Laredo	United-States	Melanie-Seite	0.2	

At the bottom right are 'Cancel' and 'Add' buttons.

- Add details as needed, and click the Add button.

- Target columns can be selected using the input panel. You can also designate a column by clicking the column header.

The screenshot shows the 'Add rule' dialog with the following fields:

- Add rule**: [Switch to editor](#)
- Command**:
- Column**:
- New column name**:
- Rename multiple columns**:
- Add** button

Inserting into a rule list

- In the list of rules of the right, place the cursor over the boundary where you wish to insert a new rule. The **+ Insert rule** button appears. Press this button.

RULE (6) SNAPSHOT (0)

- (Cr) create with sales-data-sample.csv
- [+] Insert rule**
- (He) convert row 1 to header
- (St) set type OrderDate to Timestamp
- (St) set type ShipDate to Timestamp
- (St) set type 7 columns to Long
- (St) set type 5 columns to Double

2. Select a transformation rule (command) in the command input panel. Add details as needed, and click the **Add** button.
 - When a rule is inserted in this manner, all subsequent rules are affected.
 - Rules that cannot be normally executed are displayed in red. In this case, they will revert to the results obtained in the previous step.



Edit a created rule

Editing a rule

1. In the list of rules on the right, place the cursor over the rule to be edited. The button appears. Press this button.

The screenshot shows a list of rules in the Metatron Data Preparation interface. The first rule, "create with sales-data-sample.csv", is selected, indicated by a blue circle with "Cr". The second rule, "convert row 1 to header", is also listed. The third rule, "set type OrderDate to Timestamp", is selected, indicated by a blue circle with "St" and has a checked checkbox next to it. The fourth rule, "set type ShipDate to Timestamp", is listed. A dashed line separates the selected rules from the others. Below the list are two more rules: "set type 7 columns to Long" and "set type 5 columns to Double".

2. Edit the rule in the command input panel and press the **Done** button.

- When a rule is edited in this manner, all subsequent rules are affected.



Deleting a rule

In the list of rules on the right, place the cursor over the rule to be deleted. The button appears. Press this button.

- When a rule is deleted in this manner, all subsequent rules are affected.

The screenshot shows a list of rule operations for a dataset named "sales-data-sample". The operations listed are:

- (Cr) create with sales-data-sample.csv
- (He) convert row 1 to header
- (St) set type OrderDate to Timestamp
- (St) set type ShipDate to Timestamp
- (St) set type 7 columns to Long
- (St) set type 5 columns to Double

On the right side of the list, there are two small icons: a circular arrow icon for "undo" and a circular arrow with a checkmark icon for "redo".

Undo and redo

On the upper right of the rule list are icons to perform **undo** and **redo**.

The screenshot shows a list of rule operations for a dataset named "sales-data-sample". The operations listed are:

- (Cr) create with sales-data-sample.csv
- (He) convert row 1 to header
- (St) set type OrderDate to Timestamp
- (St) set type ShipDate to Timestamp
- (St) set type 7 columns to Long
- (St) set type 5 columns to Double

On the right side of the list, there are two small icons: a circular arrow icon for "undo" and a circular arrow with a checkmark icon for "redo".

To revert to a state before executing a command, press the button.

- The dataset reverts to the state before the last transformation (including rule creation, modification, and deletion).

- All rules that were affected also revert to their previous states.

To perform the same command again, press the  button.

- Pressing  is faster than following the steps to perform the same command again. It is because the transformation results are stored in memory.

8.4.3 Rule types

This section describes each rule in terms of the following.

- Name of rule
- Required arguments
- Optional arguments
- Description
- Notes

The types of rules supported in data preparation are as follows:

- [drop](#)
- [header](#)
- [settype](#)
- [setformat](#)
- [rename](#)
- [keep](#)
- [delete](#)
- [replace](#)
- [set](#)
- [derive](#)
- [split](#)
- [merge](#)
- [extract](#)

- [countpattern](#)
- [nest](#)
- [unnest](#)
- [flatten](#)
- [aggregate](#)
- [pivot](#)
- [unpivot](#)
- [join](#)
- [union](#)
- [window](#)

In addition to these rules, data preparation provides various expressions, thereby supporting almost every function required for general data preprocessing.

drop

Required arguments

- Column: A list of target columns

Description

- Deletes the selected columns.

header

Required arguments: Row number that contains the column name (1-base)

Description

- This rule sets the content in the designated row as the column name.
- This is useful for reading a CSV file with column names in the first row.
- Unless otherwise specified, data preparation automatically performs header. This rule may be deleted if header results are not desired, but such cases are not common.

settype

Required arguments

- Column: A list of target columns
- New type: Select one out of Long, Double, String, Boolean, and Timestamp

Optional arguments

- Set format: A format string (Joda Time) in the case of timestamp

Description

- This rule changes the type of the selected columns.
- The rule is considered successful even if the result is a type mismatch, which should be separately addressed.

setformat

Required arguments

- Column: A list of target columns
- Set format: A Joda-Time format string

Description

- This rule changes the display format of a Timestamp column.
- The target column must be of the Timestamp type.

Notes

- As shown below, the format input field lists different entries depending on the input. The candidate list is narrowed as more values are entered.

The screenshot shows the Metatron Data Preparation interface. At the top, there's a table with three columns: Date, Category, and City. The first row has dates '2011-01-08T00:00:00.000+00:00', 'Furniture', and 'Huntsville'. The second row has dates '2011-01-08T00:00:00.000+00:00', 'Office-Supplies', and 'Huntsville'. The third row has date '2011-01-10T00:00:00.000+00:00', 'Office-Supplies', and 'Laredo'. Below the table, there's a section for 'Add rule' with a 'Switch to editor' link. The 'Command' dropdown is set to 'setformat'. The 'Column*' dropdown is set to 'OrderDate'. To the right of these dropdowns is a 'Format' dropdown containing several Joda-Time format strings: 'MM-dd-yy', 'MMM-dd-yy', 'MM.dd.yyyy', 'MMM.dd.yyyy', 'MM. dd. yyyy', and 'MM. dd. yyyy'. The first option, 'MM-dd-yy', is highlighted in yellow, indicating it's the current selection.

rename

Required arguments

- Column: A single target column
- New column name: New name

Description

- This rule changes the name of the selected column.
- To rename two or more columns at once, click the **Rename multiple columns** button at the bottom of the command input panel to display the following popup.

Rename

Cancel
Done

sales-data-sample.csv 28 column(s)

Before	After
OrderDate	OrderDate
Category	Category
City	City
Country	Country
CustomerName	CustomerName
Discount	Discount
OrderID	OrderID
PostalCode	PostalCode
ProductName	ProductName

OrderDate	Category	City	Country	Cust
2011-01-04T00:00:00.000+0...	Office Supplies	Houston	United States	Darre...
2011-01-05T00:00:00.000+0...	Office Supplies	Naperville	United States	Phill...
2011-01-05T00:00:00.000+0...	Office Supplies	Naperville	United States	Phill...
2011-01-05T00:00:00.000+0...	Office Supplies	Naperville	United States	Phill...
2011-01-06T00:00:00.000+0...	Office Supplies	Philadelphia	United States	Mick...
2011-01-07T00:00:00.000+0...	Furniture	Henderson	United States	Maria...
2011-01-07T00:00:00.000+0...	Office Supplies	Athens	United States	Jack...
2011-01-07T00:00:00.000+0...	Office Supplies	Henderson	United States	Maria...
2011-01-07T00:00:00.000+0...	Office Supplies	Henderson	United States	Maria...
2011-01-07T00:00:00.000+0...	Office Supplies	Henderson	United States	Maria...
2011-01-07T00:00:00.000+0...	Office Supplies	Los Angeles	United States	Livcor...

keep

Required arguments

- Condition: A conditional expression returning a Boolean value

Description

- All rows are deleted except the rows that return true for the conditional expression.

The screenshot shows the Metatron Data Editor interface. At the top, it displays the file name "sales-data-sample.csv", the number of columns (28), rows (9,994), and types (4). Below this is a preview of the data with columns: OrderDate, Category, City, Country, and CustomerName. The data preview shows various entries such as "2011-01-04 ~ 2014-12-31", "3 categories", "Houston", "United - States", and "Darren - Powers". To the right of the preview is a "RULE (6)" section with a "SNAPSHOT (!)" button and a "Done" button. The rule list contains six items: "create with sales-data-sample.csv", "convert row 1 to header", "set type OrderDate to Timestamp", "set type ShipDate to Timestamp", "set type 7 columns to Long", and "set type 5 columns to Double". At the bottom, there is an "Add rule" button, a "Switch to editor" link, and a "Cancel" or "Add" button.

delete

Required arguments

- Condition: A conditional expression returning a Boolean value

Description

- All rows that return true for the conditional expression are deleted. This is the opposite of [keep](#).

replace

Required arguments

- Column: A list of target columns
- Pattern: A string pattern to be replaced
 - In the case of a constant string: Characters enclosed inside ' ('Houston', 'Naperville', 'Philadelphia' etc.)
 - In the case of a regular expression: Characters enclosed inside / (/[,_]+/, /\s+\$/, etc.)
- New value: A new string expression to replace the specified pattern
 - Constant string
 - Regular expression \$1_\$2_\$3, etc.

Optional arguments

- Ignore between characters: Does not make any replacement for content between the characters entered here
- Match all occurrences: Whether all characters of a word must match
- Ignore case: Whether to make the strings case-insensitive

Description

- String replacement is performed for the selected columns.

Notes

- Do not use ' or / in a **new value**.
- Values from other columns are not available as **new values**. `replace` performs string replacement for content in the selected columns only. (cf. [set](#) rule)

set

The screenshot shows the Metatron Dataflow interface with a 'set' rule applied to a CSV file. The data preview shows columns: OrderDate, Category, City, Country, CustomerName, and Discount. The rule configuration on the right lists six steps: 1. create with sales-data-sample.csv, 2. convert row 1 to header, 3. set type OrderDate to Timestamp, 4. set type ShipDate to Timestamp, 5. set type 7 columns to Long, and 6. set type 5 columns to Double. At the bottom, the 'Add rule' section is shown with 'Command' set to 'set', 'Column' set to 'Category', and 'Expression' set to 'Please enter expression'.

Required arguments

- Column: A list of target columns
- Expression: An expression to be applied to the values of the target column. Values from other columns may be referenced. (cf. [replace](#) rule)

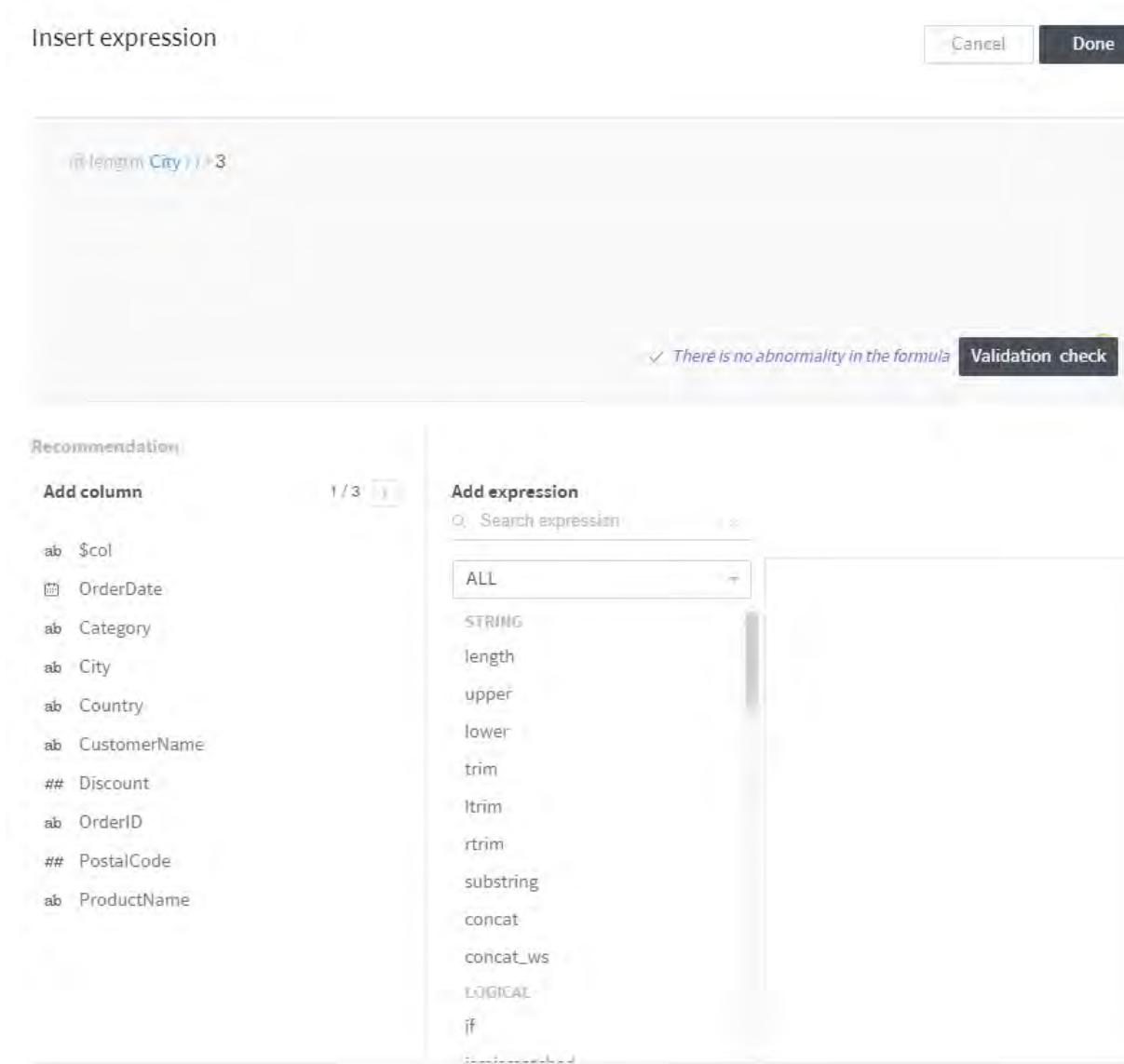
- When multiple columns are involved, use a \$col variable, which will be substituted by the respective target column during each conversion.
- That is, when applying the set command on column1 and column2, \$col becomes column1 during conversion of column1, and \$col becomes column2 during conversion of column2.

Optional arguments

- Use only under the following conditions
 - The set rule is applied only to rows satisfying this condition.
 - This rule may be regarded the same as the WHERE statement in SQL.

Description

- This rule replaces the values in the selected column with results returned by the expression.
- When using a complex expression, click the **Advanced editor** to display the popup shown below:



In the **Advanced editor**, you can edit the expression in a larger window while viewing the column list and a list of functions and their descriptions, and also run a validity check before implementing the expression.

derive

Required arguments

- Expression: An expression whose resulting values are to form a new column. Similar to the [set](#) rule, values from other columns may be referenced.

- New column name

Description

- While similar to the [set](#) rule, this rule creates a new column instead of replacing an existing one.

Notes

- The new column is inserted after the last existing column in the expression.

split

Required arguments

- Column: A list of target columns
- Pattern: A string expression that serves as a separator that splits the target strings. Allows a regular expression as is the case for the [replace](#) rule.
- Number: Number of columns to be divided into.

Description

- Each row is split by the given **Number** – 1.
- When the pattern is no longer matched, the rest columns contain a null.

Notes

- Note that columns are created as many as the **Number** input.

merge

Required arguments

- Column: A list of target columns
- Delimiter: A constant string with which values of different columns are concatenated.
- New column name

Description

- The target columns are merged with the **Delimiter** into a new column.

Notes

- Similar to the [replace](#) rule, enclosing with a ' may be skipped. That is, strings not enclosed by / or ' are automatically enclosed by '.

extract

Required arguments

- Column: A list of target columns
- Pattern: A string pattern to be extracted. Allows a regular expression as is the case for the [replace](#) rule.
- Number: Number of instances to be extracted

Optional arguments

- Ignore between characters: Does not make any replacement for content between the characters entered here
- Ignore case: Whether to make the strings case-insensitive

Description

- A new column(s) with content matching the given pattern is created.

Notes

- When there are multiple target columns, the resulting columns are inserted after each target column.

countpattern

Required arguments

- Column: A list of target columns
- Pattern: A string pattern to be detected. Allows a regular expression as is the case for the [replace](#) rule.

Optional arguments

- Ignore between characters: Does not make any replacement for content between the characters entered here
- Ignore case: Whether to make the strings case-insensitive

Description

- New columns are created based on the number of matches with the pattern.
- This is highly similar to [extract](#). The only difference is that it counts the number of matches, rather than extracting the matched content.

Notes

- When there are multiple target columns, the resulting columns are inserted after each target column.

nest

Required arguments

- Column: A list of target columns
- Type: Map or Array
- New column name

Description

- The target columns are grouped into a new column of the given type.
- Below are examples of grouping columns into an array and map, respectively.

unnest

The screenshot shows the Metatron Data Editor interface with the following details:

- Title Bar:** sales-data-sample.csv
- Toolbar:** Snapshot, Done
- Search Bar:** Search data
- Rule List:** RULE (13) SNAPSHOT (1)
 - create with sales-data-sample.csv
 - convert row 1 to header
 - set type OrderDate to Timestamp
 - set type ShipDate to Timestamp
 - set type 7 columns to Long
 - set type 5 columns to Double
 - convert Category into map
 - convert Category into array
 - move Category_1.map before Category_array
 - drop Category_1.map
 - drop Category_array
 - convert Category, City into array
 - convert Category, City into map
- Data Preview:** A preview of the data showing three columns: 'Category' (3 categories), 'City' (531 categories), and '(cate + city_.map)' (534 categories).
- Command Bar:**
 - Add rule [Switch to editor](#)
 - Command: unnest
 - Column: '(cate + city_.map)' (selected)
 - Select elements: Please enter an element to extract
- Help Tooltip:** Enter the element to extract from the selected column. Array is the index number, and Map is the Key name.

Required arguments

- Column: A single target column
- Select elements: 0-base index for an array, or key value for a map

Description

- A new column is created by extracting the selected elements from an array or a map.

Notes

- The target column must be of the array or map type.

flatten

Required arguments

- Column: A single target column

Description

- Rows are created from elements of an array.

Notes

- The target column must be of the array type.

The screenshot shows the Metatron Data Preparation interface with a rule editor open. The left pane displays a preview of a CSV file named "sales-data-sample.csv" with 29 columns, 9,994 rows, and 5 types. The main table shows data with a column labeled "array_example" containing arrays of strings. The right pane shows the rule history for RULE (16) and the current rule being edited:

```

RULE (16) | SNAPSHOT (1)
create with sales-data-sample.csv
convert row 1 to header
set type OrderDate to Timestamp
set type ShipDate to Timestamp
set type 7 columns to Long
set type 5 columns to Double
convert Category into map
convert Category into array
move Category_1_map before Category_array
drop Category_1_map
drop Category_array
convert Category_City into array
convert Category_City into map
drop `(cate + city_map)`
drop `(cate + city_array)`
convert 4 columns into array
  
```

The rule editor at the bottom shows the command "flatten" selected under "Command" and "array_example" selected under "Column".

If the target array column has four elements as shown in the above example, each original row of the array results in four rows. Non-array columns result in the same columns.

sales-data-sample.csv

29 Columns 39,976 Rows 4 Types

City	Country	CustomerName	array_example	Discount	OrderID	PostalCode
Houston	United- States	Darren- Powers	Office- Supplies	0.2	CA-2011-103800	77095
Houston	United- States	Darren- Powers	Houston	0.2	CA-2011-103800	77095
Houston	United- States	Darren- Powers	United- States	0.2	CA-2011-103800	77095
Houston	United- States	Darren- Powers	Darren- Powers	0.2	CA-2011-103800	77095
Naperville	United- States	Phillina- Ober	Office- Supplies	0.2	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	Naperville	0.2	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	United- States	0.2	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	Phillina- Ober	0.2	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	Office- Supplies	0.8	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	Naperville	0.8	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	United- States	0.8	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	Phillina- Ober	0.8	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	Office- Supplies	0.2	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	Naperville	0.2	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	United- States	0.2	CA-2011-112326	60540
Naperville	United- States	Phillina- Ober	Phillina- Ober	0.2	CA-2011-112326	60540
Philadelphia	United- States	Mick- Brown	Office- Supplies	0.2	CA-2011-141817	19143
Philadelphia	United- States	Mick- Brown	Philadelphia	0.2	CA-2011-141817	19143

Add rule [Switch to editor](#)

Command [Choose Rule Function](#)

Snapshot Done

RULE (17) | SNAPSHOT (1)

- create with sales-data-sample.csv
- convert row 1 to header
- set type OrderDate to Timestamp
- set type ShipDate to Timestamp
- set type 7 columns to Long
- set type 5 columns to Double
- convert Category into map
- convert Category into array
- move Category_1_map before Category_array
- drop Category_1_map
- drop Category_array
- convert Category, City into map
- convert Category, City into map
- drop `(cate + city_map)`
- drop `(cate + city_array)`
- convert 4 columns into array
- convert arrays in array_example to rows

Cancel Add

aggregate

The screenshot shows the Metatron Data Preparation interface. On the left is a preview of a CSV file named "sales-data-sample.csv" with 28 columns, 39,976 rows, and 4 types. The table includes columns for ProductName, Profit, Quantity, Region, Sales, and Segment. A sidebar on the right lists 180 categories and their details. On the right side, there is a "RULE (18)" panel with a "SNAPSHOT (1)" tab. The rule editor shows the following configuration:

```

Add rule Switch to editor
Command: aggregate
Expression: sum("Quantity")
Group by: OrderDate,City
  
```

Buttons at the bottom right of the rule editor are "Cancel" and "Add".

Required arguments

- Expression: A list of aggregate functions
- Group by: A list of columns that group values by.

Description

- A new column is added from the results of grouping by each combination of the elements from the GroupBy columns.
- A column is created for each expression. For example, two columns are created if average and count are designated as expressions.
- The available aggregate functions are as follows:
 - count()

- sum(colname)
- avg(colname)
- min(colname)
- max(colname)

Notes

- Calculations are performed only for sampling results. Therefore, the snapshot?the results for the entire data?may be different.
- Note that () must be inserted when using the count function.
- count(colname) is currently not available.

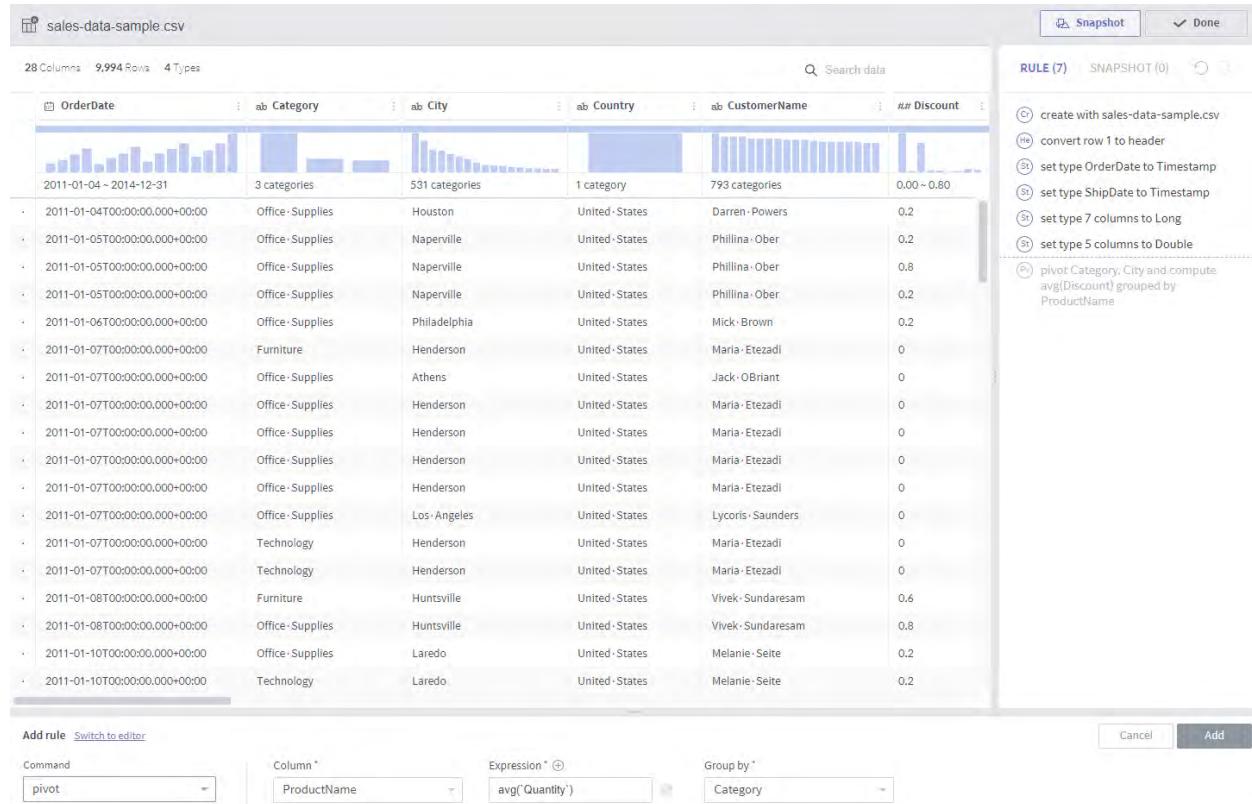
The screenshot shows the Metatron Data Editor interface. On the left, there is a data preview table titled "sales-data-sample.csv" with 3 columns, 4,723 rows, and 3 types. The columns are OrderDate, City, and sum_Quantity. The preview shows various data points, including a summary row for the entire dataset. On the right, there is a "RULE (19)" panel titled "SNAPSHOT (1)". The panel lists 20 rule steps, each with a small icon and a brief description. The steps include operations like creating the CSV file, converting rows to headers, setting column types, moving arrays, and dropping columns. At the bottom of the rule editor, there are buttons for "Cancel" and "Add".

OrderDate	City	# sum_Quantity
2011-01-04 ~ 2014-12-31	531 categories	4 ~ 224
2012-12-08T00:00:00.000+00:00	Houston	20
2014-10-21T00:00:00.000+00:00	Jacksonville	36
2014-12-11T00:00:00.000+00:00	Fairfield	40
2013-06-04T00:00:00.000+00:00	Los-Angeles	8
2012-11-10T00:00:00.000+00:00	Richmond	20
2014-11-05T00:00:00.000+00:00	Columbia	8
2012-10-19T00:00:00.000+00:00	Dallas	4
2013-07-31T00:00:00.000+00:00	Houston	12
2011-11-23T00:00:00.000+00:00	Scottsdale	12
2013-11-25T00:00:00.000+00:00	New-York-City	44
2014-11-05T00:00:00.000+00:00	Burlington	24
2011-03-21T00:00:00.000+00:00	Hamilton	20
2011-06-30T00:00:00.000+00:00	Los-Angeles	20
2014-09-25T00:00:00.000+00:00	Newark	12
2014-09-29T00:00:00.000+00:00	Jacksonville	8
2013-12-24T00:00:00.000+00:00	Bowling-Green	12
2014-06-12T00:00:00.000+00:00	Peoria	56
2014-06-26T00:00:00.000+00:00	New-York-City	4

RULE (19) | SNAPSHOT (1)

- (C) create with sales-data-sample.csv
- (H) convert row 1 to header
- (S_t) set type OrderDate to Timestamp
- (S_t) set type ShipDate to Timestamp
- (S_t) set type 7 columns to Long
- (S_t) set type 5 columns to Double
- (N_s) convert Category into map
- (N_s) convert Category into array
- (M_w) move Category_1_map before Category_array
- (D_b) drop Category_1_map
- (D_b) drop Category_array
- (N_e) convert Category_City into array
- (N_e) convert Category_City into map
- (D_b) drop `(cate+city_map)`
- (D_b) drop `(cate+city_array)`
- (N_s) convert 4 columns into array
- (F) convert arrays in array_example to rows
- (D_b) drop array_example
- (A_g) aggregate with sum(Quantity) grouped by OrderDate, City

pivot



Required arguments

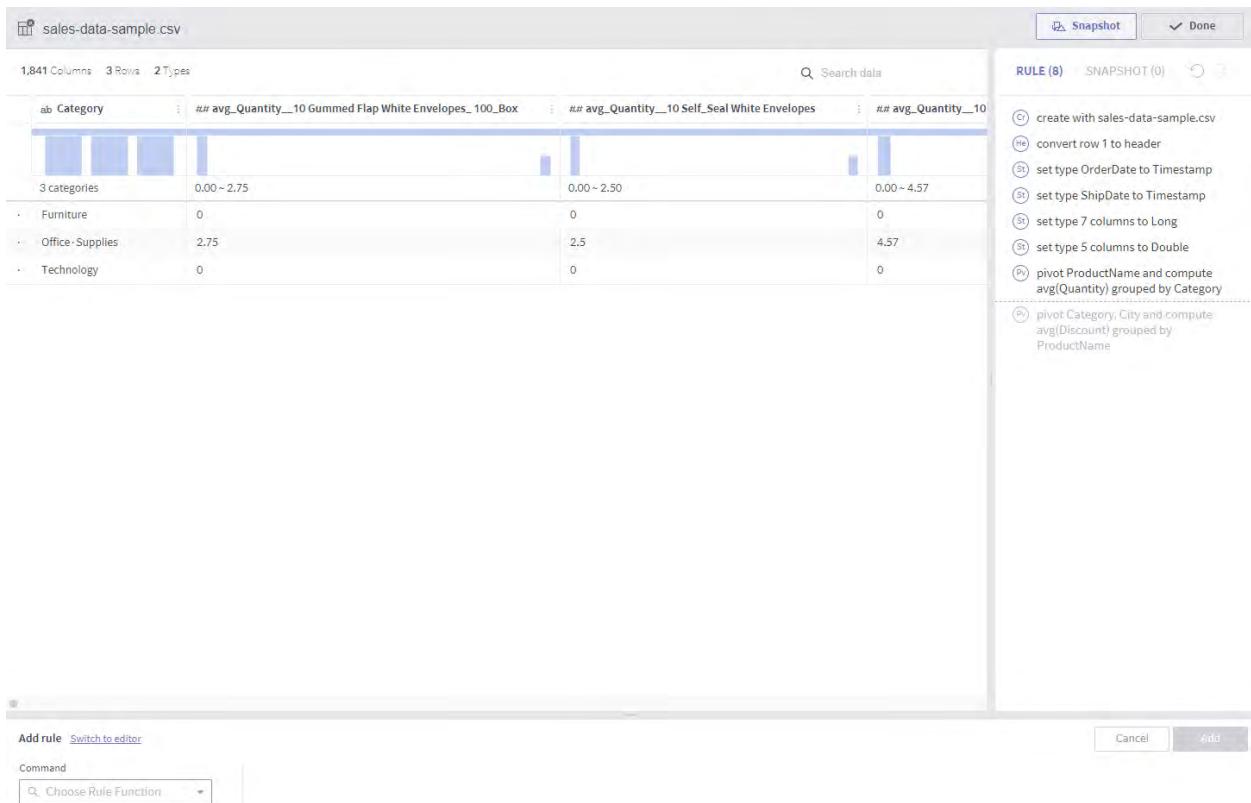
- Column: A list of columns subject to pivoting
- Expression: A list of expressions whose resulting values form new columns (only aggregate functions are available)
- Group by: A list of columns that group values by.

Description

- Group By is performed for each combination of target columns and GroupBy columns. A dataset having the results as column values is created.
- A set of columns is created for each expression. For example, if average and count are designated as expressions and the values in the pivoted columns are divided into ten groups, a total of 20 columns will be created.

Notes

- This is used when performing GroupBy on at least two columns. (1 pivoted column, 1 GroupBy column)
- Here, **Rename multiple columns** is useful as column names tend to get longer.



unpivot

The screenshot shows the Metatron Data Preparation interface. At the top, there's a file navigation bar with 'sales-data-sample.csv'. Below it, a summary bar shows '1,841 Columns', '3 Rows', and '2 Types'. A search bar and a 'Snapshot' button are also present. On the right, a sidebar titled 'RULE (8) - SNAPSHOT (0)' lists several steps: 'create with sales-data-sample.csv', 'convert row 1 to header', 'set type OrderDate to Timestamp', 'set type ShipDate to Timestamp', 'set type 7 columns to Long', 'set type 5 columns to Double', 'pivot ProductName and compute avg(Quantity) grouped by Category', and 'pivot Category, City and compute avg(Discount) grouped by ProductName'. The main area displays a table with three columns: 'Category' (with categories Furniture, Office-Supplies, Technology), 'avg_Quantity__10_Gummed Flap White Envelopes_100_Box' (values 0, 2.75, 0), and 'avg_Quantity__10_Self_Seal White Envelopes' (values 0, 2.5, 0). Below the table, a detailed configuration window for the 'unpivot' command is shown, with fields for 'Column' (set to 'avg_Quantity__10_Gummed...'), 'GroupEvery' (set to '1'), and buttons for 'Cancel' and 'Add'.

Required arguments

- Column: A list of target columns to be converted into values in new columns
- GroupEvery: Number of columns (defaults to 1)

Description

- Two columns are created?one contains the selected column names and the other contains their values. (If GroupEvery is set to 1)
- If GroupEvery is the same as the number of selected columns, each resulting pair of columns contains the name and values of its respective original column. Therefore, If 10 columns are unpivoted with the GroupEvery argument set to 10, for example, a total of 20 columns are created.

Notes

- Using the GroupEvery argument set to a factor of the number of columns will soon be supported.

{Where GroupEvery is set to 1}

9 Columns 798 Rows 1 Types

Search data

RULE (6) SNAPSHOT (0)

ab column5 ab column6 ab column7 ab column8 ab column9 ab key1 ab value1

32 categories 96 categories 14 categories 4 categories 306 categories 2 categories 136 categories

	accler	year	origin	carname	column1	mpg
504	12	70	1	chevrolet·chevelle·malibu	column2	cyl
504	12	70	1	chevrolet·chevelle·malibu	column1	18
693	11.5	70	1	buick·skylark·320	column1	8
693	11.5	70	1	buick·skylark·320	column2	15
436	11	70	1	plymouth·satellite	column1	8
436	11	70	1	plymouth·satellite	column2	18
433	12	70	1	amc·rebel·sst	column1	16
433	12	70	1	amc·rebel·sst	column2	8
449	10.5	70	1	ford·torino	column1	17
449	10.5	70	1	ford·torino	column2	8
341	10	70	1	ford·galaxie·500	column1	15
341	10	70	1	ford·galaxie·500	column2	8
354	9	70	1	chevrolet·impala	column1	14
354	9	70	1	chevrolet·impala	column2	8
312	8.5	70	1	plymouth·fury·iii	column1	8
312	8.5	70	1	plymouth·fury·iii	column2	14

Add rule Switch to editor

Command Choose Rule Function

Cancel Add

{Where GroupEvery is set to the same as the number of columns}

JM_Set1

11 Columns 399 Rows 1 Types

Search data:

jmn7	ab column8	ab column9	ab key1	ab value1	ab key2	ab value2
categories	4 categories	306 categories	1 category	130 categories	1 category	6 categories
origin	carname	column1	mpg	column2	cyl	
1	chevrolet-chevelle-malibu	column1	18	column2	8	
1	buick-skylark-320	column1	15	column2	8	
1	plymouth-satellite	column1	18	column2	8	
1	amc-rebel-sst	column1	16	column2	8	
1	ford-torino	column1	17	column2	8	
1	ford-galaxie-500	column1	15	column2	8	
1	chevrolet-impala	column1	14	column2	8	
1	plymouth-fury-iii	column1	14	column2	8	
1	pontiac-catalina	column1	14	column2	8	
1	amc-ambassador-dpl	column1	15	column2	8	
1	dodge-challenger-se	column1	15	column2	8	
1	plymouth-'cuda-340	column1	14	column2	8	
1	chevrolet-monte-carlo	column1	15	column2	8	
1	buick-estate-wagon-(sw)	column1	14	column2	8	
3	toyota-corona-mark-ii	column1	24	column2	4	
1	plymouth-duster	column1	22	column2	6	
1	amc-hornet	column1	18	column2	6	

Add rule [Switch to editor](#)

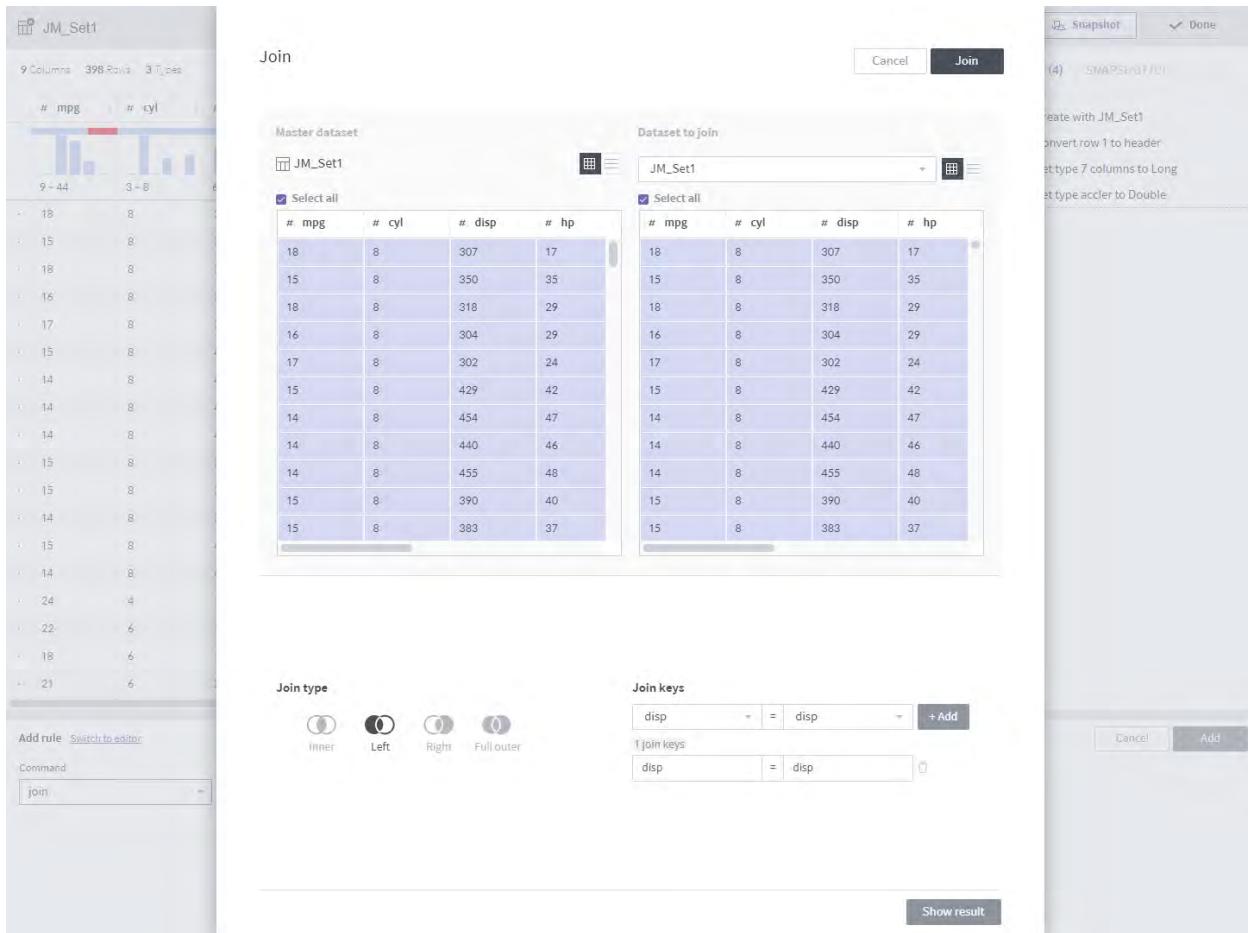
Command [Choose Rule Function](#)

RULE (7) SNAPSHOT [0]

- Cr create with JM_Set1
- Up convert column1, column2 into rows
- Up convert column1, column2 into rows
- He convert row 1 to header
- St set type:7 columns to Long
- Sd set type:acceler to Double
- Ag aggregate with avg(wt) grouped by year, origin

Cancel Add

join



Unlike other rules, join has a separate popup.

Required arguments (select in a popup or enter a value)

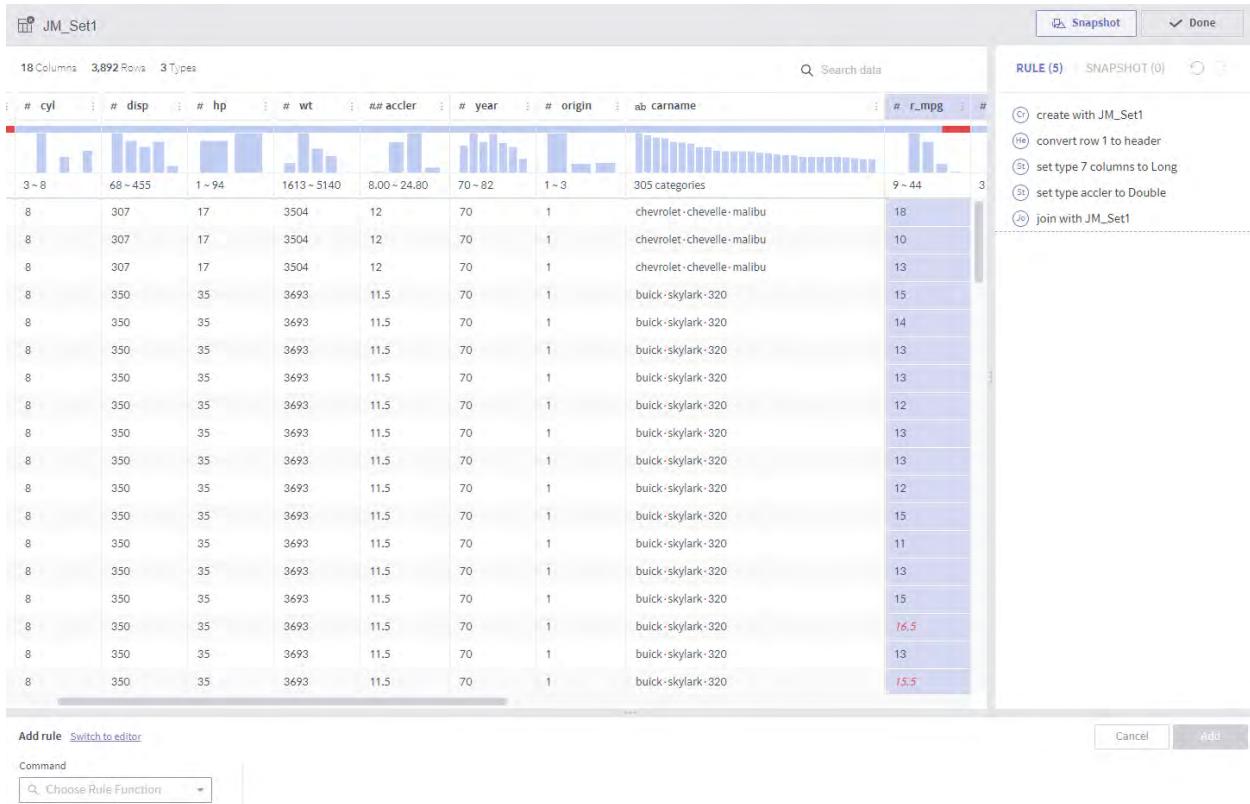
- Dataset to join: A wrangled dataset in the same dataflow
- Columns to join (toggle)
- Join keys: Multiple values may be entered
- Join type: Only inner join supported now

Description

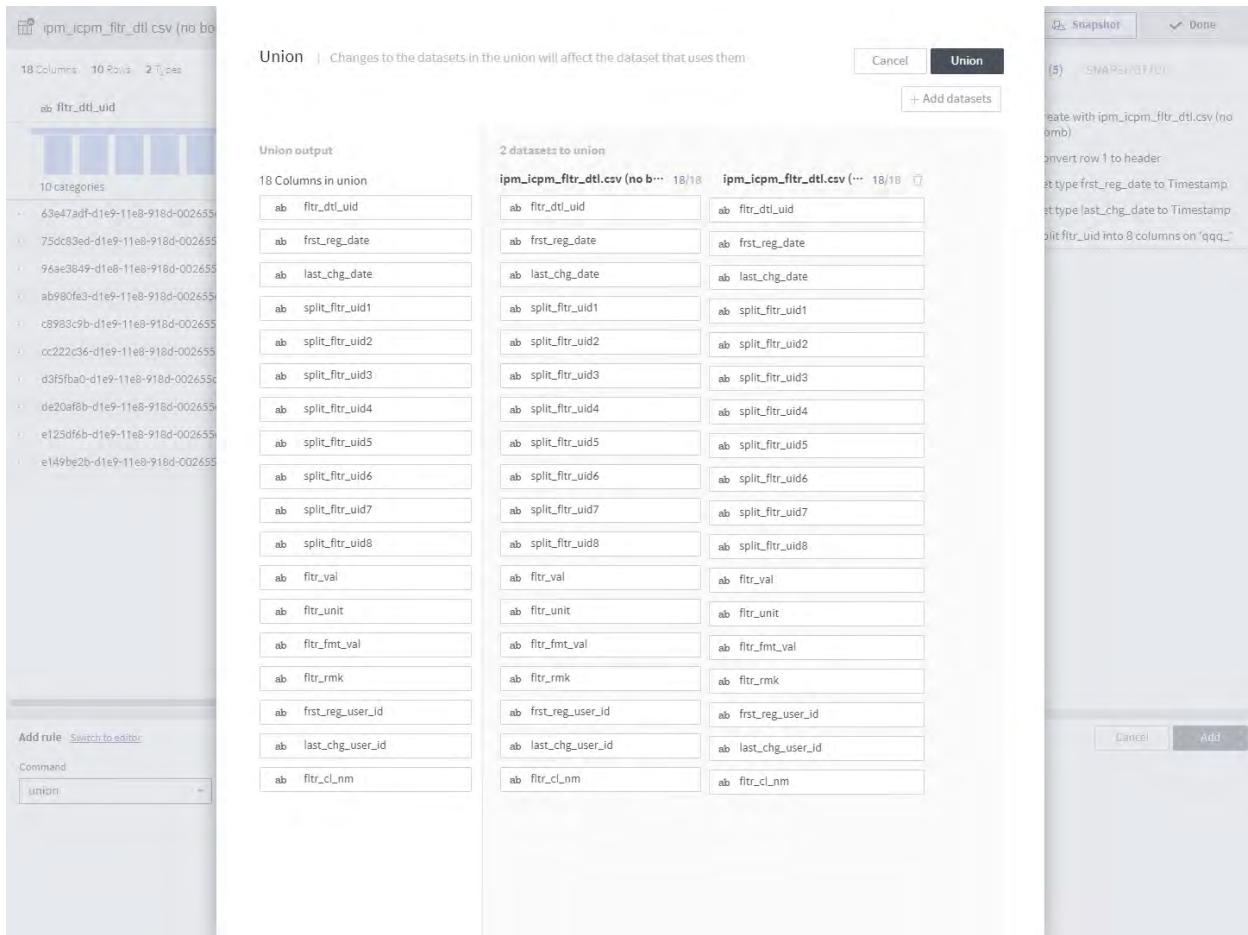
- Joins to the target dataset to create new columns.
- This rule is the same as `join` used by a relational database.
- The results can be previewed by clicking the **Show result** button.

Notes

- The join keys must be included in the columns to join.



union



Similar to [join](#), union has a separate popup.

Required arguments (select in a popup)

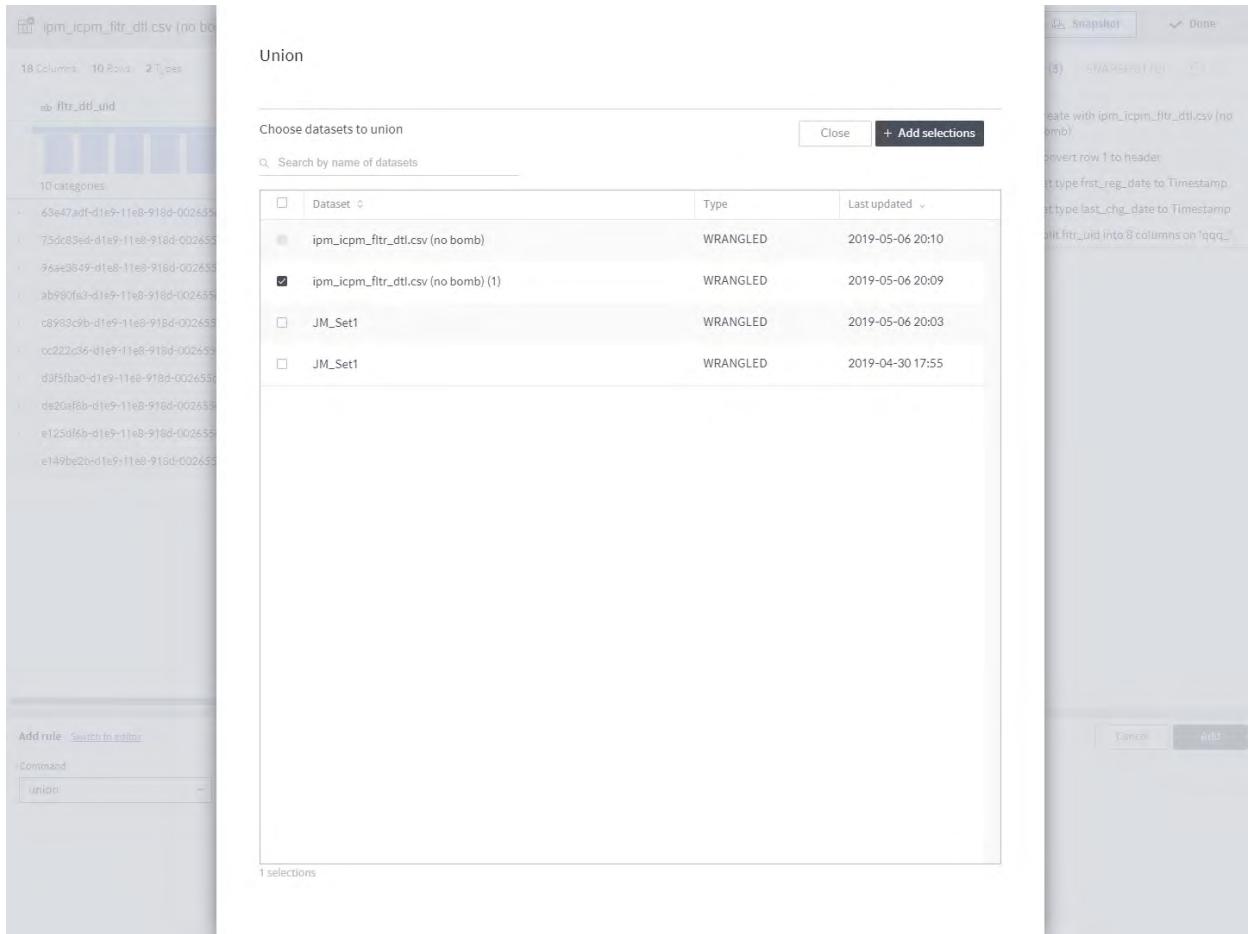
- Datasets to union: Multiple selections allowed.

Description

- The content of the selected datasets is also processed.
- This rule is the same as `union all` used by a relational database.

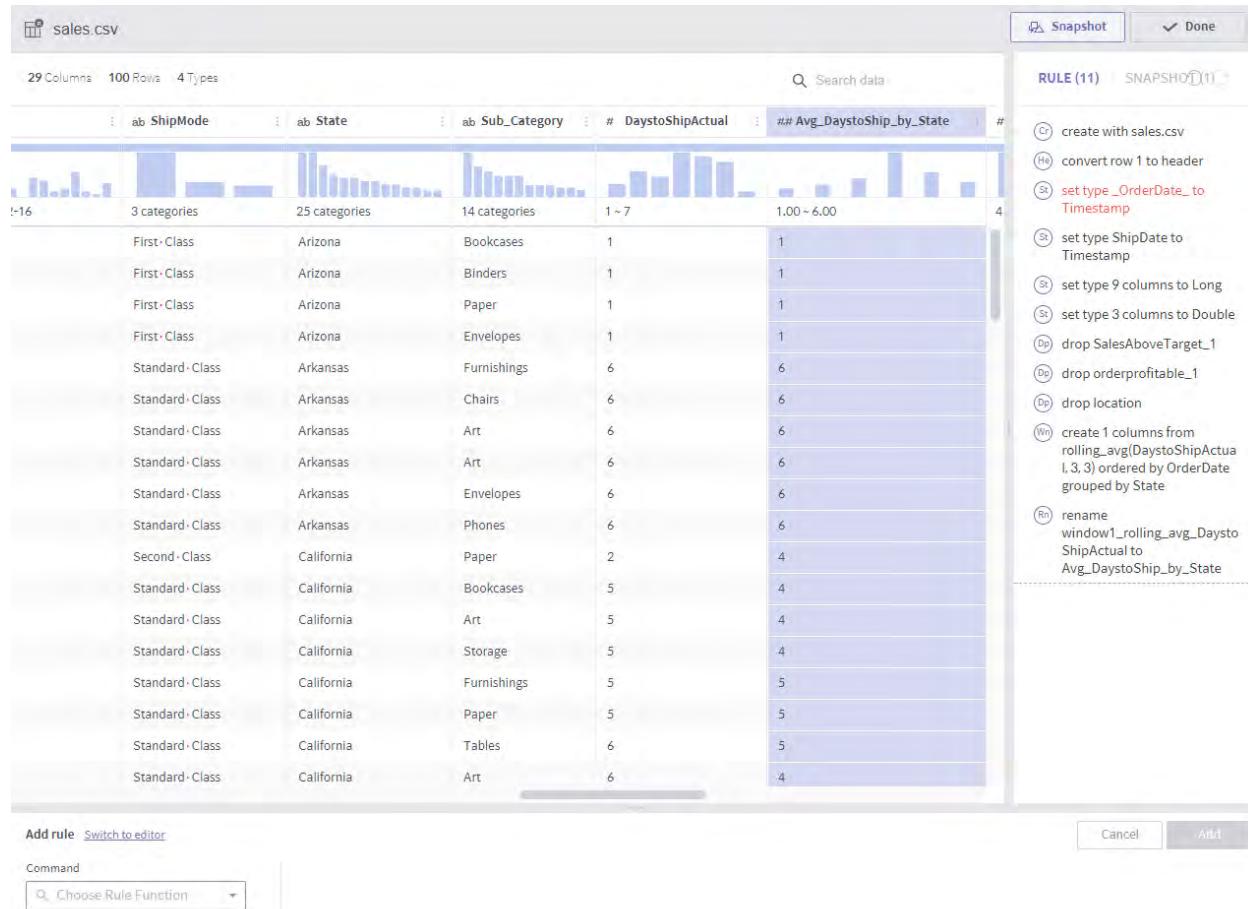
Notes

- The target datasets must coincide with the dataset that unions them in terms of column name, type, and number of columns.



window

The screenshot shows the Metatron Data Editor interface. On the left, there's a preview of a CSV file named "sales.csv" with 28 columns, 100 rows, and 4 types. The columns are labeled: ab OrderDate, ab Category, ab City, ab Country, and ab CustomerName. Below the preview are 20 rows of sample data. On the right, a sidebar titled "RULE (9)" lists various data transformation steps: create with sales.csv, convert row 1 to header, set type _OrderDate_ to Timestamp, set type ShipDate to Timestamp, set type 9 columns to Long, set type 3 columns to Double, drop SalesAboveTarget_1, drop orderprofitable_1, and drop location. At the bottom, there's a "Command" section with dropdowns for "window", "Expression" (set to "rolling_avg('DaysToShipActual')"), and "Group by" (set to "State"). Buttons for "Cancel" and "Add" are also present.



Required arguments

- Expression: A list of window functions
- Group by: A list of columns that group values by. Row order created within each group. If not specified, the whole data is sorted based on the Sort by setting.
- Sort by: Specifies columns by which the order of rows is determined. If not specified, data is sorted in the order of being inputted.

Description

- Column values are created by calculating with the values of the preceding and following rows.
- The rows are grouped first and then sorted within each group in the specified column order.
 - In the above example, each row value is averaged with the three preceding and following rows

within the same State group.

- If an immediately preceding row does not have the same state, earlier rows are searched.
- The currently available window functions are as follows:
 - `row_number()`
 - `lead(column, int)`
 - `lag(column, int)`
 - `rolling_sum(column, int, int)`
 - `rolling_avg(column, int, int)`
- In addition to window functions, aggregate functions may be used.

Notes

- When using window functions, error messages may not be properly displayed in the event of insufficient arguments.

8.4.4 Function list

You can create rules using functions. This can be a very useful method

This section describes each function in terms of the following.

- Category
- Description
- Function interface
- Arguments
- Return type
- Example
- Remarks

The following functions are currently supported by data preparation

- `length`
- `if`

- [isnull](#)
- [isnan](#)
- [upper](#)
- [lower](#)
- [trim](#)
- [ltrim](#)
- [rtrim](#)
- [substring](#)
- [concat](#)
- [concat_ws](#)
- [year](#)
- [month](#)
- [day](#)
- [hour](#)
- [minute](#)
- [second](#)
- [millisecond](#)
- [now](#)
- [add_time](#)
- [sum](#)
- [avg](#)
- [max](#)
- [min](#)
- [count](#)
- [math.abs](#)
- [math.acos](#)

- `math.asin`
- `math.atan`
- `math.cbrt`
- `math.ceil`
- `math.cos`
- `math.cosh`
- `math.exp`
- `math.expm1`
- `math.getExponent`
- `math.round`
- `math.signum`
- `math.sin`
- `math.sinh`
- `math.sqrt`
- `math.tan`
- `math.tanh`
- `time_diff`
- `timestamp`
- `row_number`
- `rolling_sum`
- `rolling_avg`
- `lag`
- `lead`
- `ismismatched`
- `contains`
- `startswith`

- `endswith`

Functions can be supplemented on an ongoing basis.

`length`

Category

- String Function

Description

- Returns the length of the input string

Function interface

- `length(string_value)`

Arguments

- `string_value`: the string whose length you want to find.

Return type

- Integer

Example

- `length(first_name)`

`if`

Category

- Logical Function

Description

- Examine the conditional statement and return a value corresponding to TRUE or FALSE.

Function interface

- `if(condition)`
- `if(condition, true_value, false_value)`

Arguments

- condition: The condition to check for true / false
- true_value: The value returned if the conditional statement is true.
- false_value: The value returned if the conditional statement is false.

Return type

- Any

Example

- `if(gender=='male') : TRUE`
- `if(age<18, 'kid', 'adult') : 'adult'`

Remarks

- If true_value/false_value does not exist, it returns TURE or FALSE as a result of Boolean type.
- true_value와 false_value의 데이터 타입은 동일해야 합니다.

isnull

Category

- Logical Function

Description

- Determines whether the value of the input column is null. Returns TRUE if null, or FALSE.

Function interface

- `isnull(condition)`

Arguments

- condition: The column to determine if null.

Return type

- Boolean

Example

- `isnull(telephone) : FALSE`

isnan

Category

- Logical Function

Description

- Determines if input value is NaN (Not-a-Number). Returns TRUE if NaN, FALSE otherwise.

Function interface

- `isnan(condition)`

Arguments

- `condition`: The column or formula for which to determine NaN.

Return type

- Boolean

Example

- `isnan(1000/ratio)`

Remarks

- The result of the condition must be a Double Value.

upper

Category

- String Function

Description

- Returns all uppercase letters of the alphabet entered.

Function interface

- `upper(string_value)`

Arguments

- `string_value`: The string to replace with an uppercase letter.

Return type

- String

Example

- `upper(last_name)`
- `upper('Hello world')` : 'HELLO WORLD'

lower

Category

- String Function

Description

- Returns all lowercase letters of the entered string.

Function interface

- `lower(string_value)`

Arguments

- `string_value`: the string you want to replace with lowercase.

Return type

- String

Example

- `lower(last_name)`
- `lower('Hello WORLD')` : 'hello world'

trim

Category

- String Function

Description

- Returns the spaces before and after the input string.

Function interface

- `trim(string_value)`

Arguments

- `string_value`: The string to remove whitespace from.

Return type

- `String`

Example

- `trim(comment)`
- `trim(' . Hi! ') : ' . Hi! '`

ltrim

Category

- String Function

Description

- Remove and return the space before the input string.

Function interface

- `ltrim(string_value)`

Arguments

- `string_value`: The string to remove whitespace from.

Return type

- `String`

Example

- `ltrim(comment)`
- `ltrim(' . Hi! ') : ' . Hi! '`

rtrim

Category

- String Function

Description

- Returns the space after the input string.

Function interface

- `rtrim(string_value)`

Arguments

- `string_value`: The string to remove whitespace from.

Return type

- String

Example

- `rtrim(comment)`
- `rtrim(' . Hi! ') : ' . Hi! '`

substring

Category

- String Function

Description

- Returns part of the input string.

Function interface

- `substring(string_value, begin_index, offset)`
- `substring(string_value, begin_index)`

Arguments

- `string_value`: The string to edit.
- `begin_index`: Start index of the part to extract from the target string. The beginning of the string is 0. If you enter a negative number, it goes back to the last character of the string.
- `offset`: The length of the string to extract from the target string. If not entered, extracts from `begin_index` to the end of the string.

Return type

- String

Example

- `substring(user_id, 0, 5)`
- `substring('hello world', 1, 7) : 'ello w'`
- `substring('metatron', -2) : 'on'`

concat

Category

- String Function

Description

- 입력된 복수의 문자열을 연결하여 반환합니다.

Function interface

- Concatenate and return multiple input strings.

Arguments

- `string_value (X)`: String to concatenate. You can enter multiple n items.

Return type

- String

Example

- `concat(first_name, ' - ', last_name) : 'Jane-Doe'`
- `concat('1980', '02') : '198002'`

concat_ws

Category

- String Function

Description

- Concatenates multiple input strings and returns a Separator between them.

Function interface

- concat(separator, string_value1, string_value2)

Arguments

- separator: Separator to insert between strings to be concatenated.
- string_value (X): String to concatenate. You can enter multiple n items.

Return type

- String

Example

- concat_ws(‘,’, first_name, last_name) : ‘Jane, Doe’
- concat_ws(‘-‘, ‘010’, ‘1234’, ‘5678’) : ‘010-1234-5678’

year

Category

- Timestamp Function

Description

- Returns a value corresponding to the year from the entered Timestamp value.

Function interface

- year(timestamp_value)

Arguments

- timestamp_value: 연도를 추출하고자 하는 timestamp

Return type

- Integer

Example

- year(birthday)

month

Category

- Timestamp Function

Description

- Returns the value corresponding to the month in the entered Timestamp value.

Function interface

- `month(timestamp_value)`

Arguments

- `timestamp_value`: the timestamp from which you want to extract the month

Return type

- Integer

Example

- `month(birthday)`

day

Category

- Timestamp Function

Description

- Returns a value corresponding to day from an entered Timestamp value.

Function interface

- `day(timestamp_value)`

Arguments

- `timestamp_value`: the timestamp from which you want to extract the day

Return type

- Integer

Example

- day(birthday)

hour

Category

- Timestamp Function

Description

- Returns a value corresponding to a time from an entered Timestamp value.

Function interface

- hour(timestamp_value)

Arguments

- timestamp_value: timestamp from which you want to extract time

Return type

- Integer

Example

- hour(last_login)

minute

Category

- Timestamp Function

Description

- Returns a value corresponding to minutes from the entered Timestamp value.

Function interface

- minute(timestamp_value)

Arguments

- timestamp_value: the timestamp from which you want to extract minutes

Return type

- Integer

Example

- minute(last_login)

second

Category

- Timestamp Function

Description

- Returns the value corresponding to seconds from the entered Timestamp value.

Function interface

- second(timestamp_value)

Arguments

- timestamp_value: the timestamp from which you want to extract seconds

Return type

- Integer

Example

- second(last_login)

millisecond

Category

- Timestamp Function

Description

- Returns the value corresponding to milliseconds (1/1000 second) from the entered Timestamp value.

Function interface

- millisecond(timestamp_value)

Arguments

- timestamp_value: the timestamp from which you want to extract milliseconds

Return type

- Integer

Example

- millisecond(last_login)

now

Category

- Timestamp Function

Description

- Returns the current time based on the entered Timezone.

Function interface

- now()
- now(timezone)

Arguments

- timezone: 현재시간을 구하고자 하는 Timezone의 full-name.

Return type

- Integer

Example

- now()
- now('Asia/Seoul')

Remarks

- If no Timezone value is entered, returns the time in UTC.

add_time

Category

- Timestamp Function

Description

- Returns the value added or subtracted from the input Timestamp value.

Function interface

- `add_time(timestamp, delta, time_unit)`

Arguments

- `timestamp`: the original timestamp value being targeted
- `delta`: the date / time value to add or subtract
- `time_unit`: The unit of date / time to add or subtract (in string). year, month, day, hour, minute, second, millisecond.

Return type

- Integer

Example

- `add_time(end_date, 10, 'day')`
- `add_time(end_date, -1, 'month')`

sum

Category

- Aggregation Function

Description

- Returns the sum of the target values.

Function interface

- `sum(target_col)`

Arguments

- target_col: Target column to sum

Return type

- Double

Example

- `sum(profit)`

Remarks

- Only available for aggregation and window rules.

avg

Category

- Aggregation Function

Description

- Returns the average of the target values.

Function interface

- `avg(target_col)`

Arguments

- target_col: Target column to average

Return type

- Double

Example

- `avg(profit)`

Remarks

- Only available for aggregation and window rules.

max

Category

- Aggregation Function

Description

- Returns the largest of the target values.

Function interface

- `max(target_col)`

Arguments

- `target_col`: Target column to get the maximum value

Return type

- Double

Example

- `max(profit)`

Remarks

- Only available for aggregation and window rules.

min

Category

- Aggregation Function

Description

- Returns the smallest of the target values.

Function interface

- `min(target_col)`

Arguments

- `target_col`: Target column to get the minimum value

Return type

- Double

Example

- `min(profit)`

Remarks

- Only available for aggregation and window rules.

`count`

Category

- Aggregation Function

Description

- Returns the number of rows in the target.

Function interface

- `count()`

Return type

- Double

Example

- `count()`

Remarks

- Only available for aggregation and window rules.

`math.abs`

Category

- Math Function

Description

- Returns the absolute value of the entered value.

Function interface

- `math.abs(value)`

Arguments

- value: A number whose absolute value you want to find.

Return type

- Double

Example

- `math.abs(-10) : 10`

math.acos

Category

- Math Function

Description

- Returns the arc cosine of the entered value.

Function interface

- `math.acos(value)`

Arguments

- value: The cosine of which you want to find the arc cosine.

Return type

- Double

Example

- `math.acos(-1) : 3.141592653589793`

math.asin

Category

- Math Function

Description

- Returns the arc sine of the entered value.

Function interface

- `math.asin(value)`

Arguments

- `value`: The sine of which you want to find the arc sine, in the range -1 to 1.

Return type

- `Double`

Example

- `math.asin(-1) : -1.5707963267948966`

math.atan

Category

- Math Function

Description

- Returns the arc sine of the entered value.

Function interface

- `math.atan(value)`

Arguments

- `value`: The sine of which you want to find the arc sine, in the range -1 to 1.

Return type

- `Double`

Example

- `math.asin(-1) : -1.5707963267948966`

math.cbrt

Category

- Math Function

Description

- Returns the cube root of the entered value.

Function interface

- `math.cbrt(value)`

Arguments

- `value`: The number whose cube root you want to find.

Return type

- Double

Example

- `math.cbrt(5) : 1.709975946676697`

`math.ceil`

Category

- Math Function

Description

- Returns the value rounded up to be a multiple of day.

Function interface

- `math.ceil(value)`

Arguments

- `value`: The number you want to round to one's place.

Return type

- Double

Example

- `math.ceil(15.142) : 16`

`math.cos`

Category

- Math Function

Description

- Returns the cosine of the entered value.

Function interface

- `math.cos(value)`

Arguments

- `value`: the radian angle to get the cosine of

Return type

- Double

Example

- `math.cos(45) : 0.5253219888177297`

math.cosh

Category

- Math Function

Description

- Returns the hyperbolic cosine of the entered value.

Function interface

- `math.cosh(value)`

Arguments

- `value`: The number whose hyperbolic cosine is to be obtained.

Return type

- Double

Example

- `math.cosh(9) : COSH(9) => 4051.5420254925943`

math.exp

Category

- Math Function

Description

- Returns the natural logarithm of e raised to the power of the input value.

Function interface

- `math.exp(value)`

Arguments

- `value`: The number of times to want to log the natural logarithm e.

Return type

- Double

Example

- `math.exp(4) : 54.598150033144236`

math.expm1

Category

- Math Function

Description

- Returns the natural logarithm e, multiplied by the value entered, minus one.

Function interface

- `math.expm1(value)`

Arguments

- `value`: The number of times to want to log the natural logarithm e.

Return type

- Double

Example

- `math.expm1(4)` : 53.598150033144236

math.getExponent

Category

- Math Function

Description

- Returns the largest of exp values that satisfy $2^{\text{exp}} \leq N$ for the entered value N .

Function interface

- `math.getExponent(value)`

Arguments

- `value`: The number corresponding to N when looking for an exp value that satisfies $2^{\text{exp}} \leq N$.

Return type

- Double

Example

- `math.getExponent(9)` : 3

math.round

Category

- Math Function

Description

- Returns the value rounded to the ones place.

Function interface

- `math.round(value)`

Arguments

- `value`: the number to be rounded to

Return type

- Double

Example

- `math.round(14.2) : 14`

math.signum

Category

- Math Function

Description

- Returns the sign of the entered value.

Function interface

- `math.signum(value)`

Arguments

- `value`: the number to extract the sign of

Return type

- Double

Example

- `math.signum(-24) : -1`

Remarks

- If the number entered is 1, it is 1, 0 is 0, and -1 if it is negative.

math.sin

Category

- Math Function

Description

- Returns the sine of the entered value.

Function interface

- `math.sin(value)`

Arguments

- `value`: the radian angle for which you want to find the sine

Return type

- `Double`

Example

- `math.sin(90) : 0.8939966636005579`

`math.sinh`

Category

- Math Function

Description

- Returns the hyperbolic sine of the entered value.

Function interface

- `math.sinh(value)`

Arguments

- `value`: the number whose hyperbolic sine is to be obtained

Return type

- `Double`

Example

- `math.sinh(1) : 1.1752011936438014`

`math.sqrt`

Category

- Math Function

Description

- Returns the square root of the entered value.

Function interface

- `math.sqrt(value)`

Arguments

- `value`: the number whose square root you want to find

Return type

- Double

Example

- `math.sqrt(4) : 2`

`math.tan`

Category

- Math Function

Description

- Returns the tangent of the entered value.

Function interface

- `math.tan(value)`

Arguments

- `value`: the radian angle for the tangent value

Return type

- Double

Example

- `math.tan(10) : 0.6483608274590866`

`math.tanh`

Category

- Math Function

Description

- Returns the hyperbolic tangent of the entered value.

Function interface

- `math.tanh(value)`

Arguments

- `value`: The angle to get the hyperbolic tangent of.

Return type

- Double

Example

- `math.tanh(4) : 0.999329299739067`

time_diff

Category

- Timestamp Function

Description

- Calculates and returns the difference between two input Timestamp values in milliseconds.

Function interface

- `time_diff(timestamp1, timestamp2)`

Arguments

- `timestamp1`: C = B - A에서 A에 해당하는 시간 값.
- `timestamp1`: C = B - A, the timestamp of B

Return type

- Double

Example

- `time_diff(order_date, shipped_date)`

Remarks

- result value = timestamp2 – timestamp1

timestamp

Category

- Timestamp Function

Description

- Create a new Timestamp value.

Function interface

- `timestamp(value, format)`

Arguments

- `value`: Date/Time value to create as timestamp value.
- `format`: The time format of the value value.

Return type

- `Timestamp`

Example

- `timestamp('2011-01-01', 'yyyy-MM-dd')` : 2011-01-01T00:00:00.000Z

row_number

Category

- Window Function

Description

- Generates serial numbers of rows arranged in order in the partition.

Function interface

- `row_number()`

Return type

- Long

Example

- `row_number()`

Remarks

- Only available with Window Rule.

`rolling_sum`

Category

- Window Function

Description

- Returns the sum of the values of the specified number of rows before and after within the partition.

Function interface

- `rolling_sum(target_col, before, after)`

Arguments

- `target_col`: Target column name to sum.
- `before`: Number of preceding rows to sum.
- `after`: The number of trailing rows to sum.

Return type

- Long/Double

Example

- `rolling_sum (profit, 3, 3)`: Combines profits for a total of seven rows, including three rows before and after the same partition.

Remarks

- Only available with Window Rule.

rolling_avg

Category

- Window Function

Description

- Returns the average of the values of the specified number of rows before and after in the partition.

Function interface

- `rolling_avg(target_col, before, after)`

Arguments

- `target_col`: The target column name for which you want to average.
- `before`: The number of preceding rows to average.
- `after`: number of trailing rows to average.

Return type

- Long/Double

Example

- `rolling_avg (profit, 3, 3)`: average of 7 rows' profits including 3 rows before and after the same partition

Remarks

- Only available with Window Rule.

lag

Category

- Window Function

Description

- Returns the value of the row that is earlier than the specified number in the partition.

Function interface

- `lag(target_col, before)`

Arguments

- target_col: Target column name.
- before: A number that specifies how far back to return the current row.

Return type

- Long/Double

Example

- lag (profit, 2): Returns the profit value of the row above 2 lines in the same partition. If there is no value above line 2, it returns null.

Remarks

- Only available with Window Rule.

lead

Category

- Window Function

Description

- Returns the value of Row after the specified number within the partition.

Function interface

- lead(target_col, after)

Arguments

- target_col: Target column name.
- after: A number that specifies how far behind the current row to return.

Return type

- Long/Double

Example

- lead (profit, 2): returns the profit value of a row below 2 lines in the same partition. If there is no value under line 2, it returns null.

Remarks

- Only available with Window Rule.

ismismatched

Category

- Logical Function

Description

- Returns whether the Value of the specified column matches a specific Column Type.

Function interface

- `ismismatched(target_col, column_type)`

Arguments

- `target_col`: Column name to check type.
- `column_type`: Type to check for match. (Type as string) String, Boolean, Timestamp, Long, Double

Return type

- Boolean

Example

- `ismismatched (birth_date, timestamp)`: false if the value of the row is timestamp, true otherwise.

contains

Category

- String Function

Description

- Returns whether the Value of the specified column contains a specific string.

Function interface

- `contains(target_col, search_word)`

Arguments

- `target_col`: The column name to search for a string.

- `search_word`: The string to search for in the column.

Return type

- Boolean

Example

- `contains (name, 'son')`: True if name contains son. ‘Micheal Jackson’, ‘Son Heung Min’, etc.

startswith

Category

- String Function

Description

- Returns whether the Value of the specified column starts with a specific string.

Function interface

- `startswith(target_col, search_word)`

Arguments

- `target_col`: The column name to search for a string.
- `search_word`: The string to search for in the column.

Return type

- Boolean

Example

- `startswith (name, 'kim')`: True if name starts with ‘kim’. Kim Chul-soo, Kim Soo-ji, etc.

endswith

Category

- String Function

Description

- Returns whether the Value of the specified column ends a specific string.

Function interface

- `endswith(target_col, search_word)`

Arguments

- `target_col`: The column name to search for a string.
- `search_word`: The string to search for in the column.

Return type

- Boolean

Example

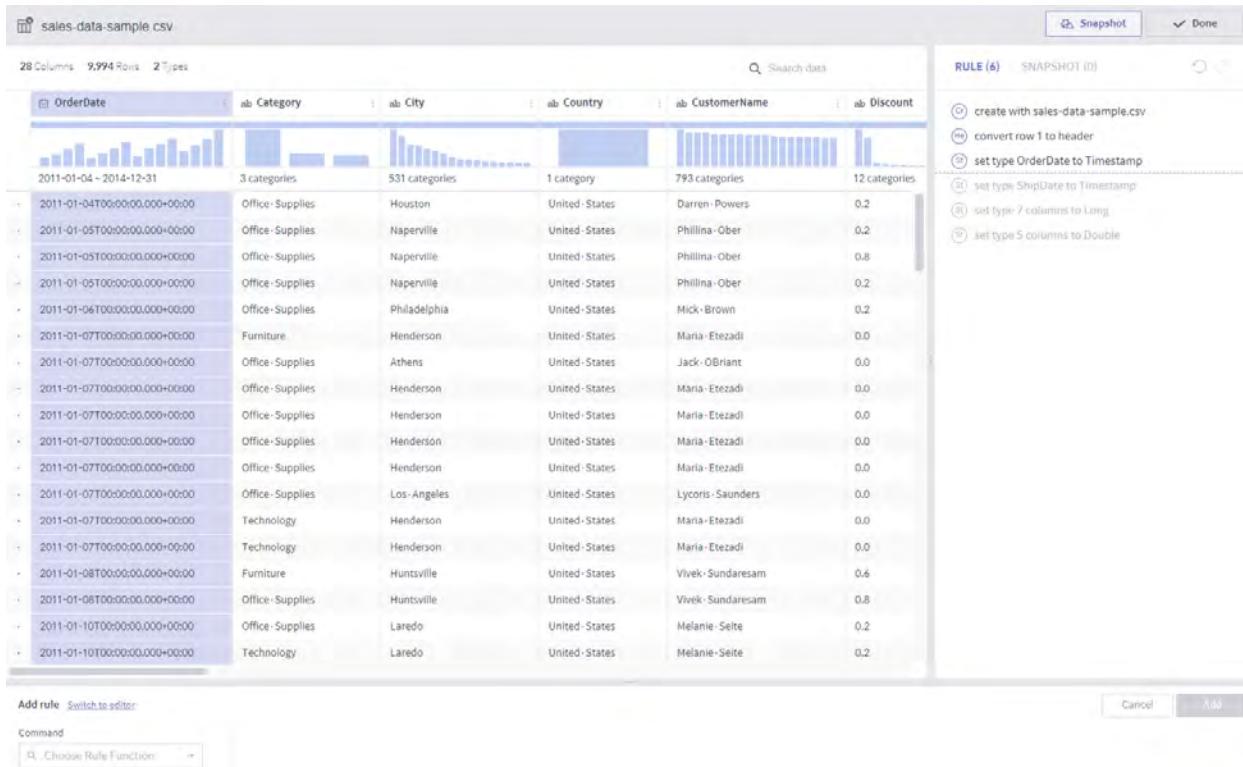
- `endswith(customer_code, 'M')`: True if `customer_code` ends with M ‘1340M’, ‘0020M’, etc.

8.4.5 Create a data snapshot

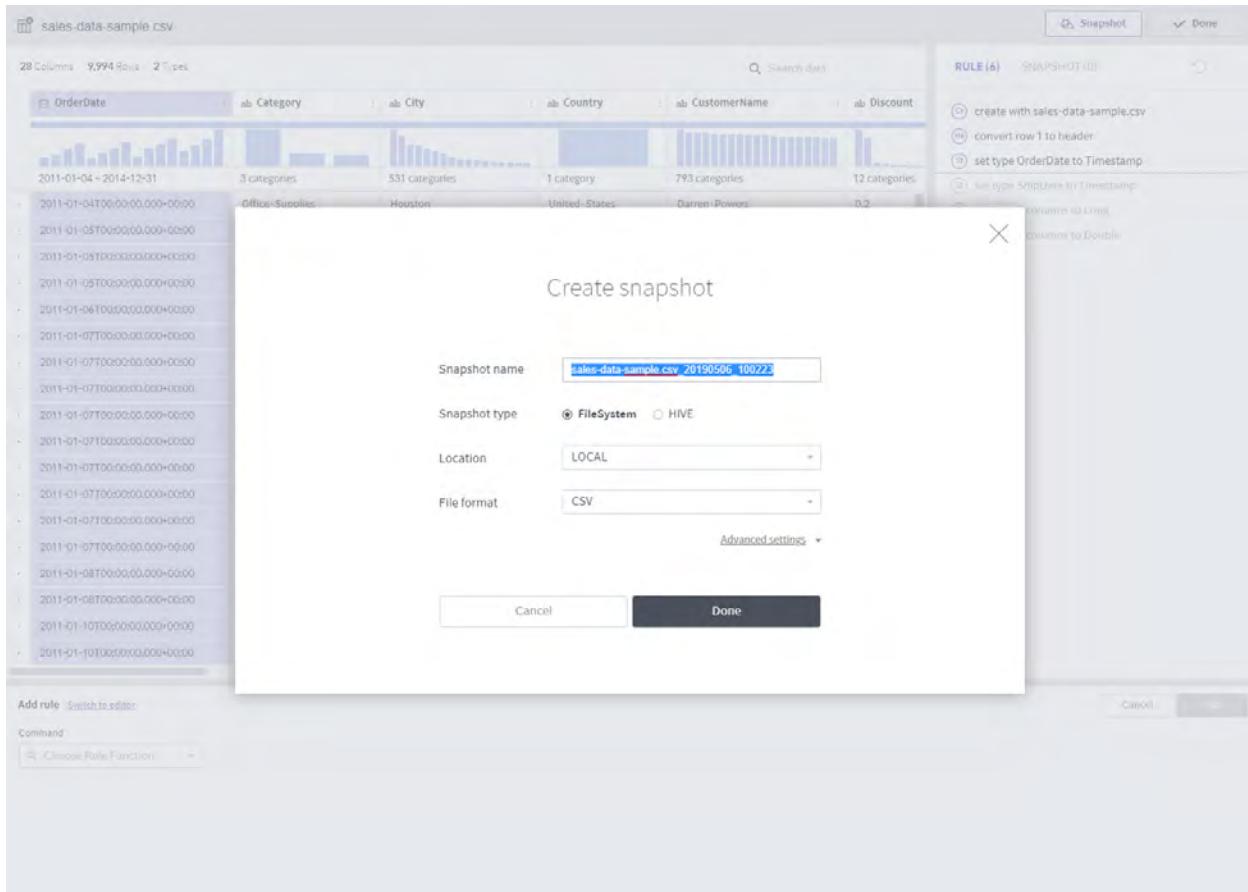
When rule editing is complete, you can create a data snapshot of the finalized dataset, which can then be downloaded to your local PC or ingested into the Metatron engine. Running the data snapshot applies the rules to the entire data, which, in the process of rule editing, applied to a sample dataset of less than 10,000 rows.

Below are instructions on creating a snapshot:

1. Click the **Data Snapshot** button on the upper right of the [Edit rules](#) window.



- When a popup is displayed to set snapshot options, select either FileSystem or HIVE (STAGING_DB) under Snapshot type.



- If FileSystem is selected as the snapshot location, the snapshot will be created as **CSV** or **JSON**.

X

Create snapshot

Snapshot name: sales-data-sample.csv_20190506_100223

Snapshot type: FileSystem HIVE

Location: LOCAL

File format: CSV

CSV
JSON

- The HIVE option is available only when STAGING_DB is enabled. A snapshot is created in the table when you designate a schema name and table name.



Create snapshot

Snapshot name

Snapshot type FileSystem HIVE

DB name

Table name

[Advanced settings](#)

3. When the snapshot is created, you can view the snapshot status and related information in the same window.

Snapshot Done

RULE (6) **SNAPSHOT (1)**

Success
✓  sales-data-sample.csv_20190506_100223 >
2019-05-06 19:03:20

 Go to snapshot list

8.5 Use data snapshot results

A **data snapshot** created through a dataflow can be used as follows:

- Check the data snapshot results
- Ingest into the Metatron engine
- Download as a CSV file

8.5.1 Check the data snapshot results

The status of snapshot creation can be classified as follows:

- **Success** = SUCCEEDED
- **Failed** = FAILED
- **Preparing** = INITIALIZING, RUNNING, WRITING, TABLE_CREATING, CANCELING

You can view the details of snapshot creation through the two paths below:

- Go to the snapshot list under **MANGEMENT** > **Data Preparation** > **Data Snapshot**.

Name	Dataflow Dataset	Status	Elapsed time	Created
sales-data-sample.csv_20190506_100223	Sales Analysis sales-data-sample.csv	✓	00:00:01.00	2019-05-06 19:03 by ...
JM_Set1_20190430_055433	JM_Set1_0430_1449 JM_Set1	✓	00:00:00.00	2019-04-30 14:54 by j...
JM_Set1_20190430_055143	JM_Set1_0430_1449 JM_Set1	✓	00:00:02.00	2019-04-30 14:52 by j...

- Click the **Snapshot (#)** tab on the right of the [Edit rules](#) page in **Dataflow**

The screenshot shows a data snapshot summary page. At the top right are buttons for "Snapshot" and "Done". Below them, a section titled "RULE (6) SNAPSHOT (1)" shows a "Success" status with a green checkmark. It displays a file icon and the path "sales-data-sample.csv_20190506_100223" along with the timestamp "2019-05-06 19:03:20". To the left is a data preview grid with columns: "ab Country", "ab CustomerName", "ab Discount", and "ab OrderID". The first row shows counts: "1 category", "793 categories", "12 categories", and "5009 categories". Subsequent rows show specific data points like "United States" and "Phillina Ober".

In the snapshot details page, you can view details such as data validity ratio and a grid of the created snapshot, and download the results as a CSV file ([Download as CSV](#)).

This screenshot shows the details of the "sales-data-sample.csv_20190506_100223" snapshot. On the left is a data grid with columns: "ab OrderDate", "ab Category", "ab City", "ab Country", "ab CustomerName", "ab Discount", "ab OrderID", and "ab PostalCode". The grid contains 20 rows of sample data. To the right is a sidebar with various metadata:

- sales-data-sample.csv_20190506_100223**
- Snapshot type:** FILE (CSV)
- File URI:** file:///data/metatron-discovery/dataprep/sn...
- Summary:** 9,994 row(s), 28 column(s)
- Elapsed time:** 00:00:01.00
- Created:** 2019-05-06 19:03
- Dataset:** sales-data-sample.csv in Sales Analysis
- Created:** 2019-05-06 18:52
- Updated:** 2019-05-06 19:00
- Origin:** imported dataset
- Data source:** sales-data-sample.csv
- Created:** 2019-05-06 18:41

At the bottom left is a "Rule list" section with several items listed, and at the bottom right is a "Create datasource" button.

If valid data has not been created, the snapshot details page displays an error log.

Error log

```

java.util.ArrayList.rangeCheck(ArrayList.java:657)
java.util.ArrayList.get(ArrayList.java:433)
app.metatron.discovery.domain.dataprep.teddy.DataFrame.getColName(DataFrame.java:218)
app.metatron.discovery.domain.dataprep.teddy.DfJoin.gatherPredicates(DfJoin.java:73)
app.metatron.discovery.domain.dataprep.teddy.DfJoin.prepare(DfJoin.java:116)
app.metatron.discovery.domain.dataprep.teddy.TeddyExecutor.applyRuleStrings(TeddyExecutor.java:465)
app.metatron.discovery.domain.dataprep.teddy.TeddyExecutor.transformRecursive(TeddyExecutor.java:429)
app.metatron.discovery.domain.dataprep.teddy.TeddyExecutor.createUriSnapshot(TeddyExecutor.java:292)
app.metatron.discovery.domain.dataprep.teddy.TeddyExecutor.run(TeddyExecutor.java:169)
app.metatron.discovery.domain.dataprep.teddy.TeddyExecutor$5FastClassBySpringCGLIB$8a9fffb2.invoke()
org.springframework.cglib.proxy.MethodProxy.invoke(MethodProxy.java:204)
org.springframework.aop.framework.CglibAopProxy$CglibMethodInvocation.invokeJoinpoint(CglibAopProxy.java:738)
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:157)
org.springframework.aop.interceptor.AsyncExecutionInterceptor$1.call(AsyncExecutionInterceptor.java:115)
java.util.concurrent.FutureTask.run(FutureTask.java:266)
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
java.lang.Thread.run(Thread.java:748)

```

Trello Action Log (Saved) _20190405_083124

Snapshot type:	FILE (CSV)
File URI:	file:///data/metatron-discovery/dataprep/sn...
Elapsed time:	00:00:00.00
Created:	2019-04-05 17:31

Dataset

Trello Action Log (Saved) in 3.2 필터링 히스토리	
Created:	2019-04-01 01:59
Updated:	2019-04-05 14:37

Origin imported dataset

Datasource:	Trello Action Log
Created:	2019-04-01 01:55

Rule list

- (10) Rule list
- ① create with Trello Action Log
- ② set type date to Timestamp
- ③ join with Trello ID table
- ④ move date before id
- ⑤ move name after date
- ⑥ move type after name

8.5.2 Ingest into the Metatron engine

(upcoming feature)

8.5.3 Download as a CSV file

In the details page of a successfully created snapshot, the **Download as CSV** option is enabled.

The screenshot shows the Metatron Discovery interface with a data snapshot titled "sales-data-sample.csv_20190506_100223". The main area displays a grid of data with columns: ab_OrderDate, ab_Category, ab_City, ab_Country, ab_CustomerName, ab_Discount, ab_OrderID, and ab_PostalCode. A red box highlights the "Download as CSV" button. To the right, there's a sidebar with dataset information, including snapshot type (FILE (CSV)), file URI (file:///data/metatron-discovery/dataprep/snapshots/sales-data-sample.csv_20190506_100223), and creation details (Created: 2019-05-06 19:03). Below the sidebar, there's a "Rule list" section with several items and a "Create datasource" button.

The downloaded file is a standard CSV, with each value separated by a “comma” and each row by a “new line.”

The screenshot shows a spreadsheet application with a table titled "test - Sheet1 (1)". The table has columns labeled "column1" through "column5". The data rows include:

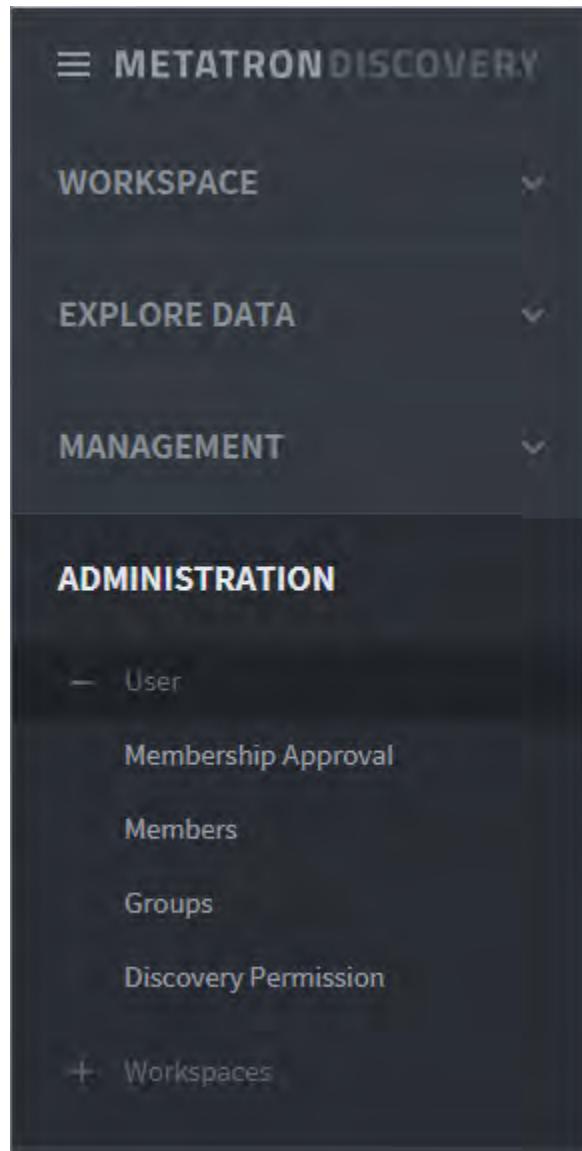
column1	column2	column3	column4	column5_1	column5
test1	test2	test3	test4	{column4=test4, column3=test3}	{"a":"a","b":"b"}
1	2	3		{column4=, column3=3.0}	{"a":"a","b":"b"}
1		3	4	{column4=4.0, column3=3.0}	{"a":"a","b":"b"}
1	2			{column4=, column3=}	{"a":"a","b":"b"}
		3		{column4=, column3=3.0}	{"a":"a","b":"b"}

CHAPTER
NINE

ACCOUNT MANAGEMENT

The administrator can set and manage the membership and permissions of Metatron Discovery users, and these tasks are facilitated by the **Group** functionality.

To manage users, click ADMINISTRATION → USER on the left-hand panel of the main page and select a submenu you want to use.



9.1 Membership Approval

This menu shows applications for membership. As shown below, the list includes the applications that have been rejected or waiting for approval. But, approved users are listed here but can be found in the **Members** menu.

Users

The screenshot shows a user management interface with the following details:

- Header:** Membership Approval, Members, Groups, Discovery Permission.
- Status Filter:** All (radio button selected), Pending, Rejected.
- Request Date Filter:** All, Today, Last 7 days, yyyy-MM-dd hh:mm -> yyyy-MM-dd hh:mm, Apply.
- Search Bar:** Search by username or full name.
- Table Headers:** Username, Full Name, Email, Request date, Status.
- Data Rows:**

Username	Full Name	Email	Request date	Status
applicant	EE	applicant24@gmail.com	2019-08-21 22:39	<input checked="" type="button"/> Approve <input type="button"/> Reject
tester_00	tester_00	skt.metatron@gmail.com	2019-05-14 13:08	Rejected ⓘ
tester_00	tester_00	skt.metatron@gmail.com	2019-05-14 12:53	Rejected ⓘ
tester_00	tester_00	skt.metatron@gmail.com	2019-05-14 11:37	Rejected ⓘ
sehwa.lee	sehwa.lee	sehwa.lee@sk.com	2019-05-09 18:43	Rejected ⓘ
admin_test	aaa	kyungtaak@gamil.com	2019-04-29 10:13	Rejected ⓘ
asd	ASD	asd@asd.com	2018-12-10 13:12	Rejected ⓘ
sbparks	sbparks	sbparks@sbparks.sbparks	2018-12-06 13:23	Rejected ⓘ
tester	Tester	test@test.com	2018-11-23 13:11	Rejected ⓘ
pp333	pp333	ppeee@test.com	2018-11-22 14:51	Rejected ⓘ
pp333	pp333	pp333@pp333.pp333	2018-11-15 15:52	Rejected ⓘ
pp222	pp222	pp222@pp222.pp222	2018-11-15 15:51	Rejected ⓘ
pp111	pp111	pp111@pp111.pp111	2018-11-15 15:51	Rejected ⓘ
pp000	pp000	pp000@pp000.pp000	2018-11-15 15:51	Rejected ⓘ
p888	p888	p888@p888.p888	2018-11-15 15:50	Rejected ⓘ
ppp3333	ppp3333	ppp3333@cc.com	2018-11-15 15:02	Rejected ⓘ
- Pagination:** Show up to 20.

9.2 Members

This menu allows you to view and manage registered users.

Users can sign up for Metatron Discovery in one of the following two ways:

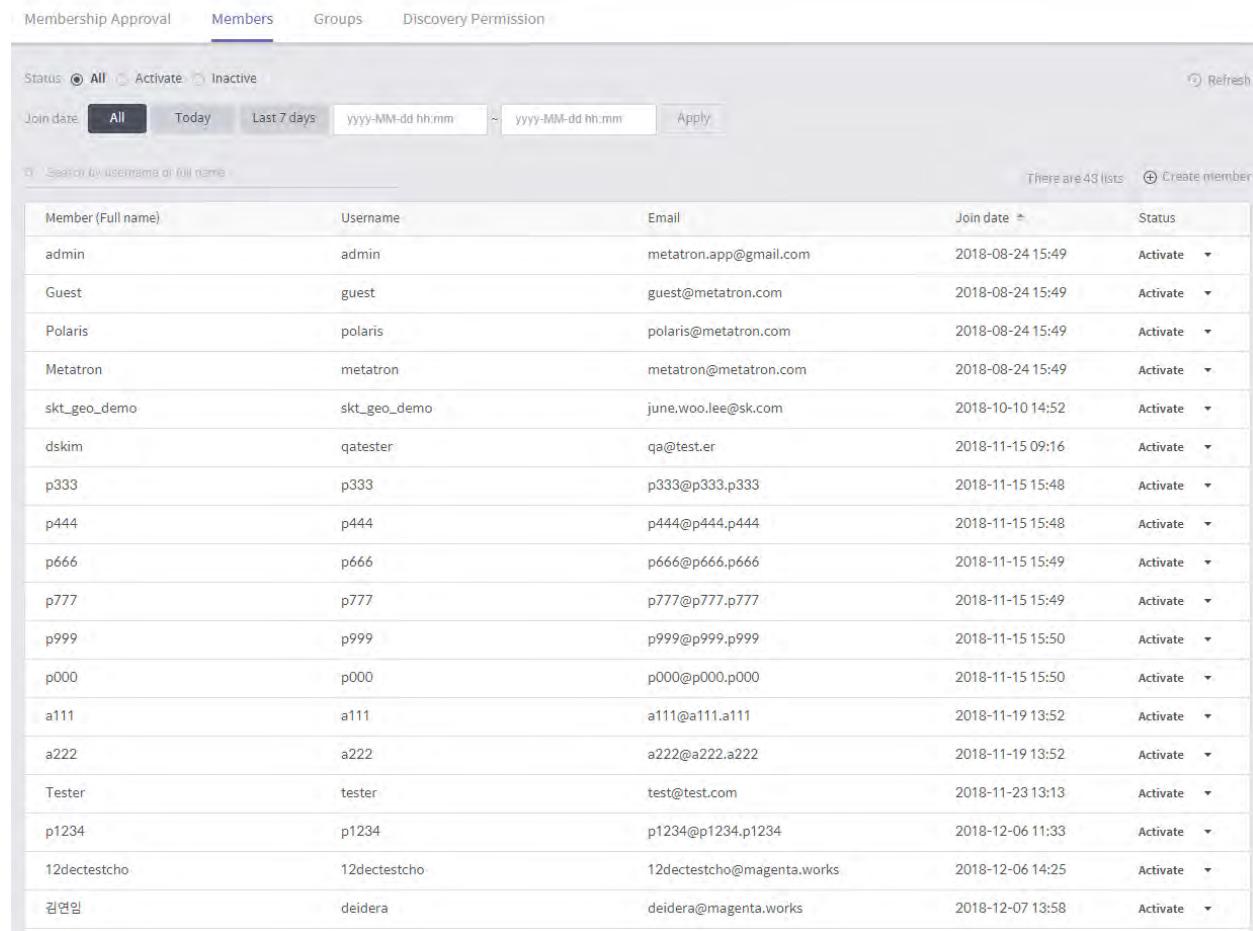
- Administrator's approval of a user's application for membership (see [Membership Approval](#))
- Registration by the administrator (see [Register a member](#))

9.2.1 Members home

The Members home shows a list of Metatron Discovery members. The member list can be filtered by various criteria, and clicking on an entry in the list allows you to view and edit the selected member's

information.

Users



The screenshot shows a user management interface with the following details:

- Header:** Membership Approval, Members (selected), Groups, Discovery Permission.
- Status Filter:** All (selected), Activate, Inactive.
- Date Range:** Join date (All, Today, Last 7 days), Search fields for yyyy-MM-dd hh:mm and yyyy-MM-dd hh:mm, Apply button.
- Search:** Search by username or full name.
- Count:** There are 43 lists.
- Create:** Create member.

Member (Full name)	Username	Email	Join date	Status
admin	admin	metatron.app@gmail.com	2018-08-24 15:49	Activate ▾
Guest	guest	guest@metatron.com	2018-08-24 15:49	Activate ▾
Polaris	polaris	polaris@metatron.com	2018-08-24 15:49	Activate ▾
Metatron	metatron	metatron@metatron.com	2018-08-24 15:49	Activate ▾
skt_geo_demo	skt_geo_demo	june.woo.lee@sk.com	2018-10-10 14:52	Activate ▾
dskim	qatester	qa@test.er	2018-11-15 09:16	Activate ▾
p333	p333	p333@p333.p333	2018-11-15 15:48	Activate ▾
p444	p444	p444@p444.p444	2018-11-15 15:48	Activate ▾
p666	p666	p666@p666.p666	2018-11-15 15:49	Activate ▾
p777	p777	p777@p777.p777	2018-11-15 15:49	Activate ▾
p999	p999	p999@p999.p999	2018-11-15 15:50	Activate ▾
p000	p000	p000@p000.p000	2018-11-15 15:50	Activate ▾
a111	a111	a111@a111.a111	2018-11-19 13:52	Activate ▾
a222	a222	a222@a222.a222	2018-11-19 13:52	Activate ▾
Tester	tester	test@test.com	2018-11-23 13:13	Activate ▾
p1234	p1234	p1234@p1234.p1234	2018-12-06 11:33	Activate ▾
12dectestcho	12dectestcho	12dectestcho@magenta.works	2018-12-06 14:25	Activate ▾
김연임	deidera	deidera@magenta.works	2018-12-07 13:58	Activate ▾

9.2.2 View and edit member information

Clicking on a member in the list opens the member information page shown below:

Information

Full Name	admin
Username	admin
Email	metatron.app@gmail.com
Permission	Manage system, Manage and monitor datasource, Use shared workspace, Use personal workspace, Manage workspace custom schema
Phone	0000000000

Groups (1)

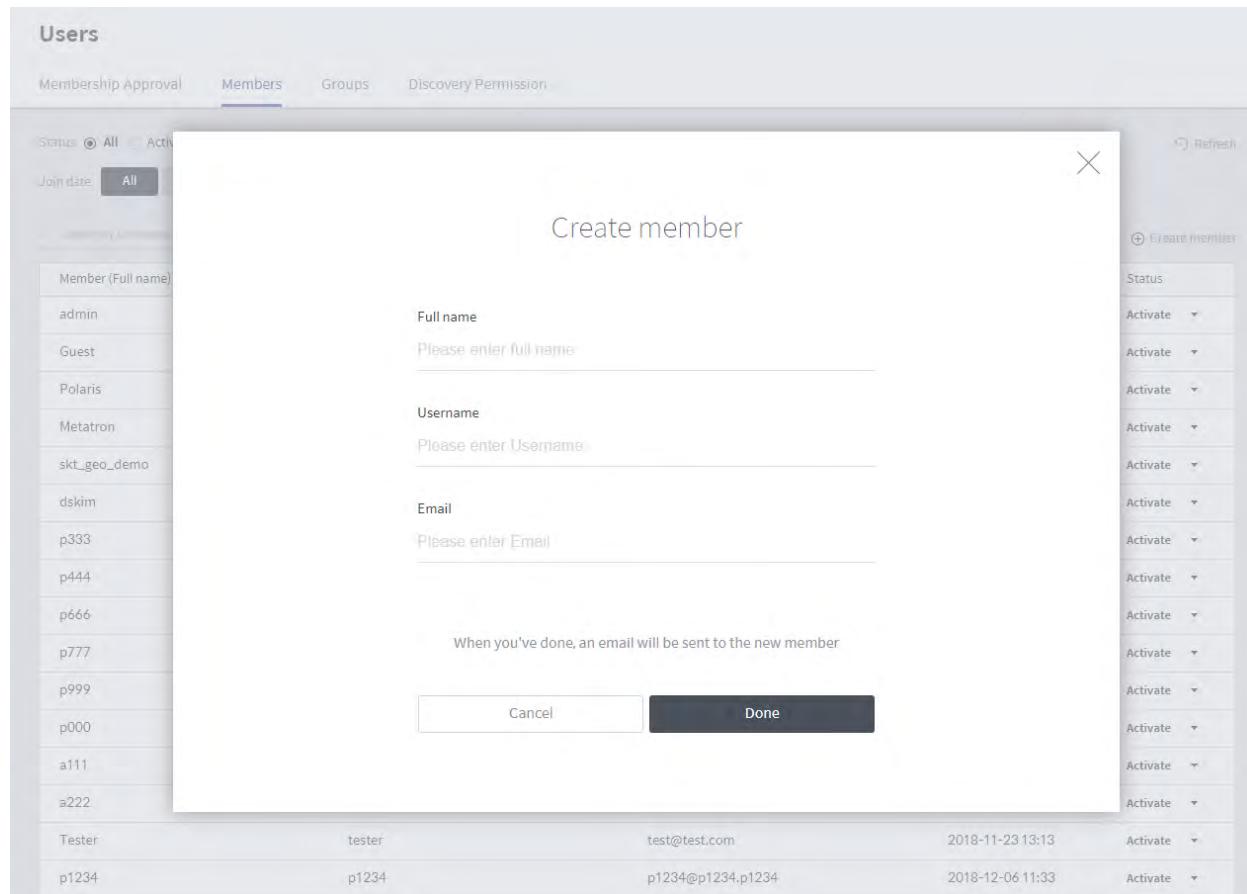
System-Admin	Manage system, Manage and monitor datasource, Use shared workspace, Use personal workspace, Manage workspace custom schema
--------------	--

This page displays some basic information and allows a number of settings.

- **Status setting (Active/Inactive):** An inactive member cannot log in to the system.
- **Reset password:** By clicking this, a user who has forgotten the password can receive an email to reset it.
- **Group setting:** Click on the icon to add or delete groups to which the member belongs. See [Groups](#) for details on the user group.

9.2.3 Register a member

Click the Create member button on the top right of the page to pop up the member creation dialog box below.



Enter the member's real name, ID, and email address to register the member, and the membership details will be sent to the email address.

9.3 Groups

By grouping Metatron Discovery users, you can use the following convenient features:

- Batch setting of a permission for all users in a group
- Sending an email to all users in a group

9.3.1 Groups home

The Groups home shows the user groups currently registered in Metatron Discovery. The group list can be filtered by various criteria, and clicking on an entry in the list allows you to view and edit the selected

group's information.

Users

Membership Approval		Members	Groups	Discovery Permission
Create date	All	Today	Last 7 days	yyyy-MM-dd hh:mm ~ yyyy-MM-dd hh:mm
				Apply
<input type="text"/> Search by Username or full name				There are 30 lists
				<input type="button"/> Create group
Group	Description	Members		Create date
Data-Manager		12		2018-08-24 15:49
General-User		62		2018-08-24 15:49
System-Admin		9		2018-08-24 15:49
#1425		0		2019-03-07 10:42
11222	1122222	0		2018-11-15 15:46
14	14	0		2018-11-15 15:46
1414.14142		0		2019-03-07 10:46
15	15	0		2018-11-15 15:46
16	16	0		2018-11-15 15:46
17	17	0		2018-11-15 15:46
18	18	0		2018-11-15 15:46
19	19	0		2018-11-15 15:47
2	2	0		2018-11-15 15:44
20	20	1		2018-11-15 15:47
21	21	0		2018-11-15 15:47
22	22	0		2018-11-15 15:47
3	3	0		2018-11-15 15:46
4	4	0		2018-11-15 15:46

9.3.2 View and edit group information

Clicking on a group in the list opens the group information page shown below:

[Data-Manager](#)

Created on 2018-08-24 15:49 by **admin** Last update on 2019-06-12 17:04 by **admin**

Information

Name:	Data-Manager
Description:	
Permission:	Manage and monitor datasource, Use shared workspace, Use personal workspace

Members(10) email to all users

polaris
qatester
deidera
demo
heesoo
jungil.park
choong
sting
kyungtaak
SKH

This page provides the following functions:

- Check the selected group's basic information, assigned permissions, and members.
- Click on the icon to add or delete members to or from the group.
- Click the email to all users button to send an email to all members of the group.

9.3.3 Register a group

Click the Create group button on the top right of the page to pop up the group creation dialog box below.

The screenshot shows the 'Users' section of the Metatron interface. The 'Groups' tab is active. A modal window titled 'Create group' is open, containing fields for 'Name' and 'Description'. The background table lists groups with columns for ID, Name, Description, and creation date.

ID	Name	Description	Created At	Action
2	2		2018-11-15 15:44	<input type="checkbox"/> Delete
10	20		2018-11-15 15:47	<input type="checkbox"/> Delete
21	21		2018-11-15 15:47	<input type="checkbox"/> Delete
22	22		2018-11-15 15:47	<input type="checkbox"/> Delete

Enter a name and description for the group and click Done to create the new group.

9.4 Discovery Permissions

Metatron Discovery supports four types of permissions shown below, thereby enabling the administrator to grant different user privileges. This menu allows permission settings for individual members or groups.

Users

Membership Approval	Members	Groups	Discovery Permission
There are 4 lists			
Discovery Permission	Description	Member	Group
Manage and monitor datasource	Access with data management menu. Able to create and manage data. In addition, users with this permission can...	0	2
Manage workspace custom schema	Create and manage custom schemas in owner workspaces.	0	1
Use personal workspace	Have a private workspace that only you can access, and you are authorized for its administration.	0	3
Use shared workspace	Create a new shared workspace and access your shared workspace.	1	3

Click on one of the four permissions presented on the home to list the individual members and groups assigned the selected permission.

≡ METATRON DISCOVERY

← Manage and monitor datasource Access with data management menu. Able to create and manage data. In addition, users with this permission can...

Information

Name	Manage and monitor datasource
Description	Access with data management menu. Able to create and manage data. In addition, users with this permission can monitor the usage of data.

Users

Members (0)	No member
Groups (2)	▼ Data-Manager and 1 more groups.

In the **Member** or **Group** section, click on the icon to pop up the following settings dialog box where you can set which members or groups will be assigned the permission.

Set shared member & group

[Cancel](#)[Done](#)

Member 0

Group 2

Search by Username or full name

All (15/43)

#error (test)

12dectestcho (12dectestcho)

a111 (a111)

a222 (a222)

admin (admin)

al.lee (al.lee)

choong (choong)

DD (member)

delete_user2 (delete_user2)

delete_user3 (delete_user3)

Demo (demo)

dskim (qatester)

eeee (eeee)

Guest (guest)

hive (hive)

0 selections

Full Name	UserName
-----------	----------

CHAPTER
TEN

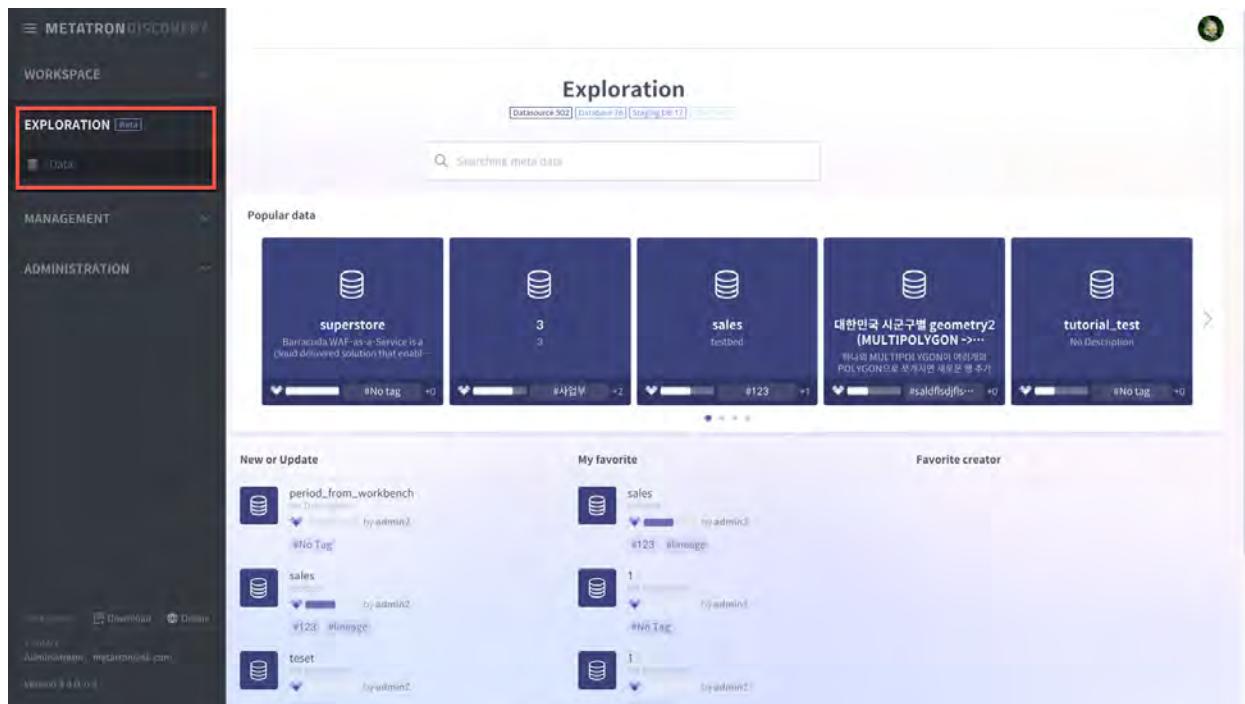
DATA EXPLORATION

The administrator can set · manage the membership and permissions of Metatron Discovery users. By using **group** feature, you can be more effective with administration.

For Data Exploration, click Exploration from the left panel of the main view and select the submenu you want. Also, for smooth data exploration of users, Admin should manage the Metadata. Click Management → Exploration and select a submenu.

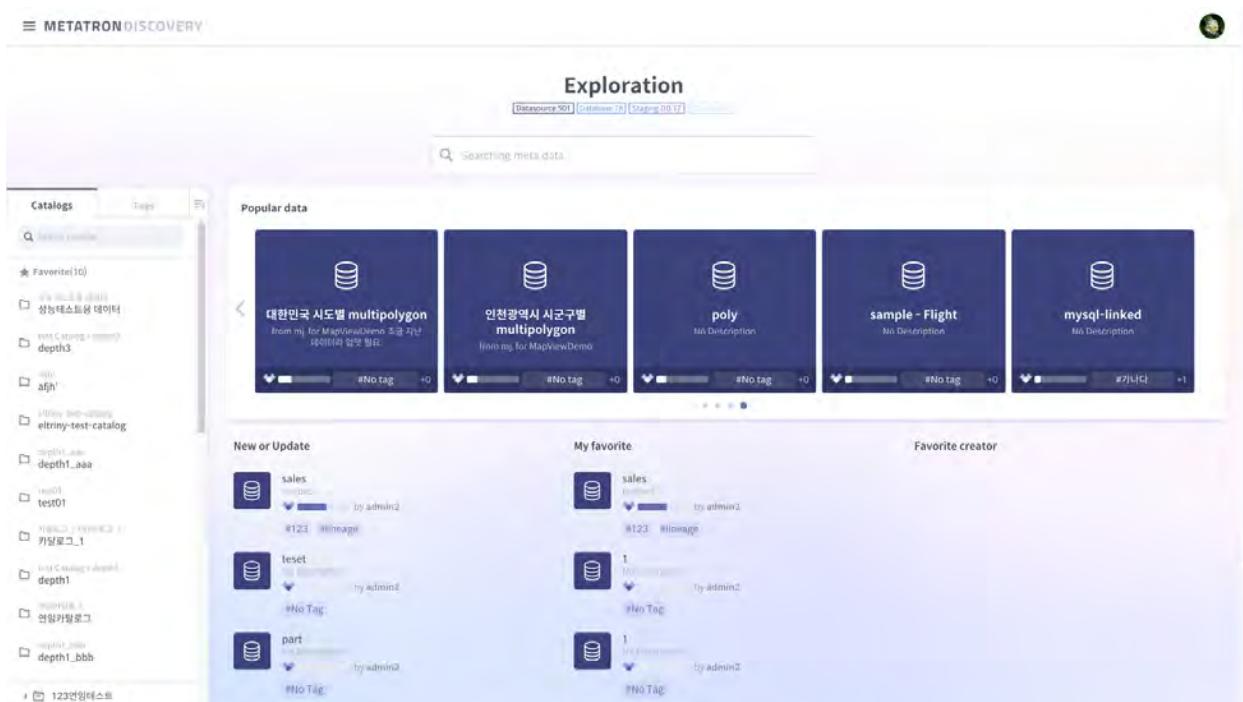
10.1 Data Exploration

The aim of providing data exploration feature is to enable easy data search wherever the data is located at, and for visualizing the found data.



10.1.1 Data Exploration Overview

At the Overview, you can manage data of the current source DB, StagingDB(Slave DB) – provided by Metatron Discovery – and the data in the Engine(Druid).



10.1.2 Data Exploration Detail View

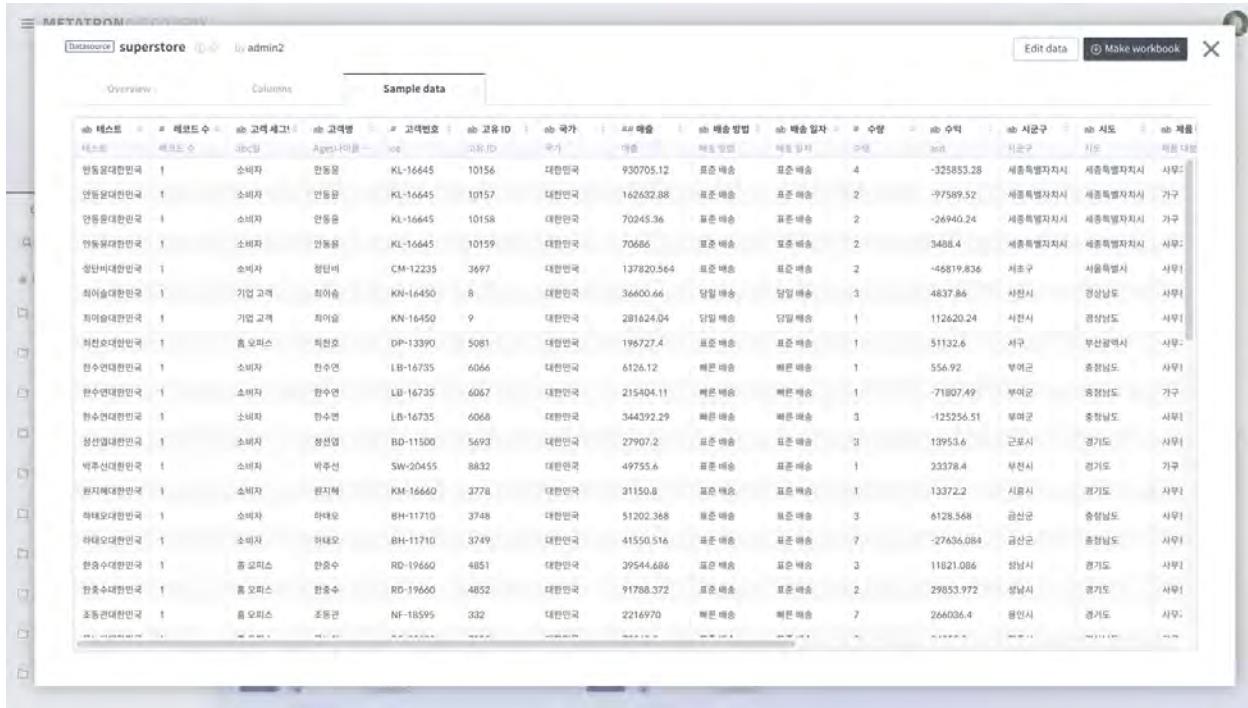
With Data Exploration, you can find the data you want fast.

The screenshot shows the Metatron Data Explorer interface with the 'superstore' datasource selected. The main area displays the 'Overview' tab, which includes sections for 'Data name', 'Description', 'Tags', 'Data Popularity', and 'Catalogs'. A 'Recently Updated' section lists recent changes made by 'admin2'. Below this is a 'Recently Used' section showing five thumbnail previews of different data visualizations.

Data information is provided with 3 main sections: Overview, Column Scheme, Sample Data. According to each data types, workbook(for Datasource type), workbench(for DB type) action button is enabled.

The screenshot shows the Metatron Data Explorer interface with the 'superstore' datasource selected. The main area displays the 'Columns' tab, which lists various columns with their details such as Role, Column popularity, Column name, Logical column name, Optionality, Type, Codable, and Description. The columns listed include 테스트, 레코드 수, 고객 세그먼트, 고객별, 고객번호, 고유 ID, 국가, 매출, 배송 방법, 배송 일자, 수령, 수익, 시군구, and 시도.

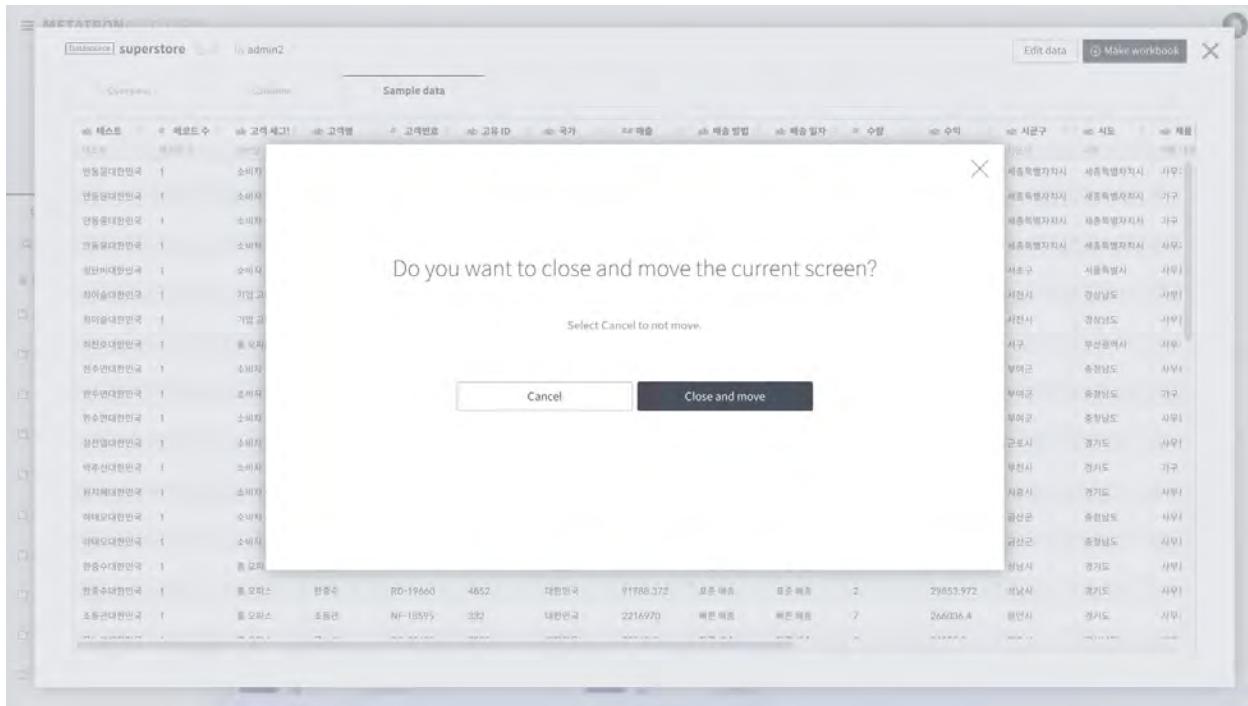
Sample data list displays up to 100 rows. If you are authorized, you can view more and download via ‘Management > Exploration’. If you have ‘Edit data’ button on the top right of the detail view it means that you are authorized. The button leads you to ‘Management > Exploration’.



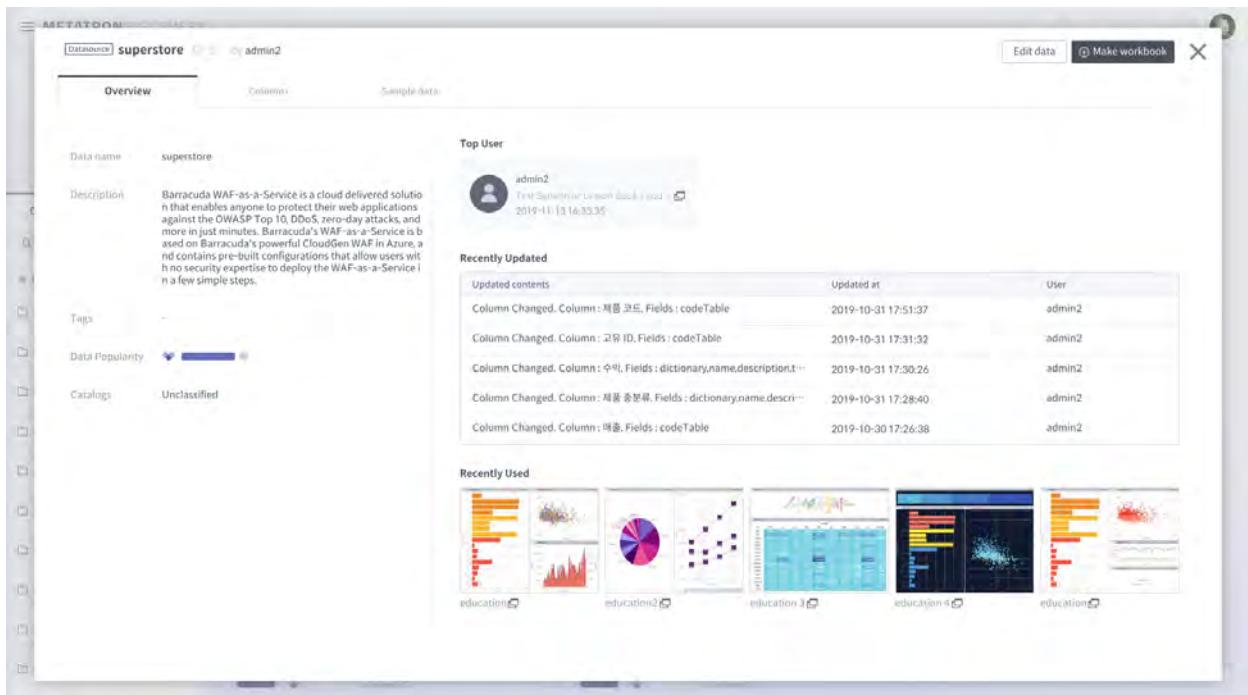
The screenshot shows the Metatron Data Explorer interface with the following details:

- Top Bar:** METATRON, superstore, by admin2, Edit data, Make workbook, X.
- Left Sidebar:** Overview, Columns, Sample data.
- Table Headers:** sb 테스트, # 레코드 수, sb 고객 세그먼트, sb 고객명, # 고객번호, sb 고유 ID, sb 국가, # 매출, sb 배송 방법, sb 배송 일자, # 수령, sb 수익, sb 시군구, sb 시도, sb 제작.
- Table Rows:** A large list of 20 rows, each containing data such as customer segment, name, ID, country, sales, delivery method, delivery date, shipping address, and profit.

When you jump to other menu, an alert like below appears.



The second below image is the view when proceeded to ‘Management > Exploration’. Here, you can view more meta information in detail and manage them as the administrator.



The screenshot shows the Metatron Discovery interface. At the top, there's a navigation bar with a logo and a user icon. Below it, a search bar contains the text 'superstore'. The main content area has tabs for 'Information', 'Data Grid', and 'Catalogs'. Under 'Information', there's a section for 'Metadata information' with a summary card for 'superstore'. This card includes a popularity bar, tags, a catalog button, and a detailed description about Barracuda WAF-as-a-Service. Below this is a 'Source information' section with details like Data type: Datasource, Status: DISABLED, Created at: 2019-04-16 16:01 by admin2, and Updated at: 2019-10-30 15:19 by admin2.

Find data fast with search and filter features such as catalog and tag.

The screenshot shows the Metatron Exploration interface. At the top, there's a navigation bar with a logo and a user icon. Below it, a search bar contains the text 'All'. The main content area has a 'Catalogs' sidebar with a search bar and a list of catalogs: 'Superstore', '서기일로그1', and '10/10/2019'. The main panel displays a table of data sources. The columns are: Data type, Name, Table, Data Popularity, Modifier, and Updated. The table lists 20 entries, each with a popularity bar and a timestamp. Some entries have a yellow background, indicating they are selected or highlighted.

Data type	Name	Table	Data Popularity	Modifier	Updated
Datasource	3.2 stable test		low	Administrator	2019-07-23 14:57
Datasource	BOOKS		medium	Administrator	2019-07-26 17:50
Datasource	Book s		high	Administrator	2019-07-26 17:55
Datasource	NYC Job s		medium	Administrator	2019-07-26 17:45
Datasource	zzzz s		medium	Administrator	2019-10-22 14:21
Staging DB	abc		medium	Administrator	2019-07-23 16:02
Datasource	book_inge s ted		medium	Administrator	2019-11-12 15:23
Datasource	book_linked2		medium	Administrator	2019-11-12 15:22
Datasource	cklg_s sample		medium	Administrator	2019-07-23 16:01
Datasource	encoding 10/10/2019		medium	Administrator	2019-10-22 08:24
Datasource	futbol_femenino		medium	Administrator	2019-07-26 17:47
Datasource	po s tgresql view test		medium	Administrator	2019-10-07 17:37
Datasource	real estate_trade 2019-03-05	#realestate + 2	medium	Administrator	2019-07-23 14:21
Datasource	Sales 9/2018-2019-COMI	#sales	high	Administrator	2019-10-22 08:58
Datasource	Southkorea_multipolygon		medium	Administrator	2019-07-23 14:28
Datasource	test 10/10/2019 08:00:00		medium	Administrator	2019-10-22 08:23

In Metatron Discovery, you can manage data with catalogs. Classify catalogs according to classifications such as groups, and use the catalogs to fast search data.

The screenshot shows the 'Exploration' view in Metatron Discovery. On the left, there's a sidebar with 'Catalogs' and 'Tags' tabs, and a search bar. The main area displays a table of catalogs. The table has columns for 'Data type', 'Name', 'Tags', 'Data Popularity', 'Modifier', and 'Updated'. There are two entries:

Data type	Name	Tags	Data Popularity	Modifier	Updated
Staging DB	abc		Low	Administrator	2019-07-23 16:02
Datasource	Sales	# sales	High	Administrator	2019-10-22 08:58

At the bottom right, there's a 'Show up to' dropdown set to 20.

10.1.3 Favorite Data view

This feature is in preparation.

10.1.4 Data Creator view

This feature is in preparation.

10.2 Metadata Management

Metadata was created to manage the data displayed on Exploration view and analyze them in more detail.

The screenshot shows the Metatron Discovery interface with the following details:

- Left Sidebar:** Contains sections for WORKSPACE, EXPLORATION (with Data), MANAGEMENT (including Exploration, Metadata, Column Dictionary, Code Table, Catalog), DATA STORAGE, DATA PREPARATION, NOTEBOOK, DATA MONITORING, ENGINE MONITORING (with Data), and INTERPRETER.
- Top Bar:** Includes tabs for Query, Code Table, Catalog, and a search bar.
- Main Area:** A table listing items with columns for Tags, Data Popularity, and Updated. The table shows 595 items, with the last few rows listed below:

Tags	Data Popularity	Updated
123 + 1	High	2019-11-15 09:21 by admin2
	Medium	2019-11-14 16:46 by admin2
	Low	2019-11-13 16:13 by admin2
	Medium	2019-11-13 16:11 by admin2
	High	2019-11-12 16:51 by admin2
	Medium	2019-11-12 16:34 by admin2
	Low	2019-11-12 16:27 by admin2
	Medium	2019-11-12 15:29 by admin2
	High	2019-11-12 14:34 by admin2
	Medium	2019-11-12 08:31 by admin2
	Low	2019-11-08 10:11 by admin2
	Medium	2019-11-08 10:07 by admin2
	High	2019-11-07 14:12 by admin2
- Bottom Navigation:** Includes links for User Center, Download, and Online.

The screenshot shows the Metatron Discovery interface with the following details:

- Top Bar:** Shows the URL [superstore](#) and a message: "Barracuda WAF-as-a-Service is now available in Azure! Metatron has added support for it in the Data Source section."
- Left Sidebar:** Contains sections for superstore, INFORMATION, Data Grid, and Column Analysis.
- Right Panel - Metadata Information:**
 - Metadata Information:** Shows a progress bar for Popularity (blue, almost full).
 - Tags:** A list of tags including `datasource`, `superstore`, and `datasource`.
 - Catalogs:** A button labeled "Add".
 - Description:** Text: "Barracuda WAF-as-a-Service is a cloud delivered solution that enables anyone to protect their web applications against the OWASP Top 10, DDoS, zero-day attacks, and more in just minutes. Barracuda's WAF-as-a-Service is based on Barracuda's powerful CloudGen WAF in Azure, and contains pre-built configurations that allow users with no security expertise to deploy the WAF-as-a-Service in a few simple steps." A "Go to Datasource" button is also present.
- Bottom Section - Source Information:**

Data type	Datasource
Status	DISABLED
Created at	2019-04-16 16:01 by admin2
Updated at	2019-10-30 15:19 by admin2

10.3 Column Dictionary

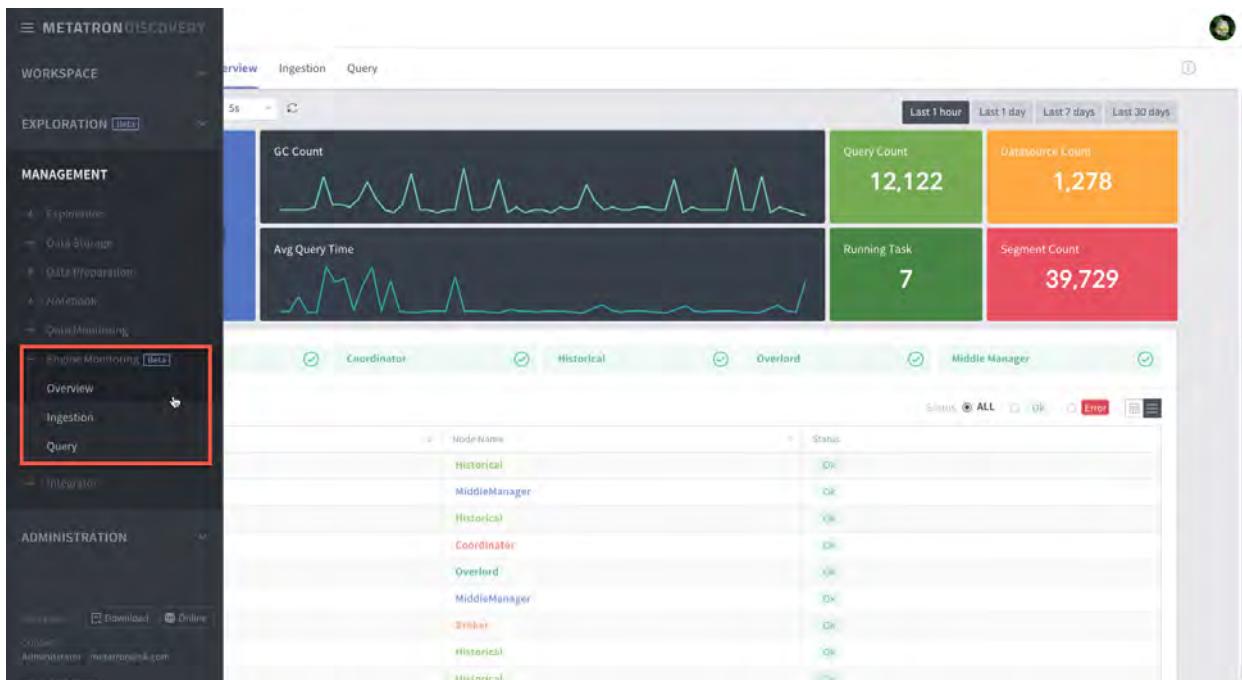
The screenshot shows the Metatron Discovery interface. On the left, the 'Dictionary' tab is selected, displaying 'Dictionary Information' and 'Format Information'. A modal window titled 'Choose a code table' is open, showing a table named 'Ages_10_Code' with a description '나이를 10살 단위로 표현'. The table contains several rows of data, each with a 'Preview' link. On the right, a large red button at the top right of the modal says 'Delete this Column Dictionary'.

10.4 Code Table

The screenshot shows the Metatron Discovery interface with the 'Code Table' tab selected. The table lists various code tables, each with a 'Table Name', 'Description', and 'Updated' timestamp. The table includes entries such as 'eltriny-test-board-hideadasdasdadds...', '반송상태코드 0_0_0_0_0_0_0_0_0...', 'Ages_10_Code', 'iii일', '일이상사오목칠판구십일이삼사오목칠판구십일이삼사오목칠판구십일이삼사오목칠판구십일이삼...', '일이삼사오목칠판구십일이삼사오목칠판구십일이삼사오목칠판구십일이삼사오목칠판구십일이삼...', 'iiihibj', 'iiiini', 'rrrr', 'codecodecodecodecodecodecode...', '페이지레이지레이지레이지레이지레이...', '시도코드to시도별', '돈', and 'qwe'. The 'Updated' column shows dates ranging from 2018-12-10 to 2019-11-07.

ENGINE MONITORING

Engine Monitoring is a feature to monitor the Metatron Engine. Metatron Engine is a time series-based engine using Druid. Engine Monitoring displays Ingestion, Query status monitoring and log details.

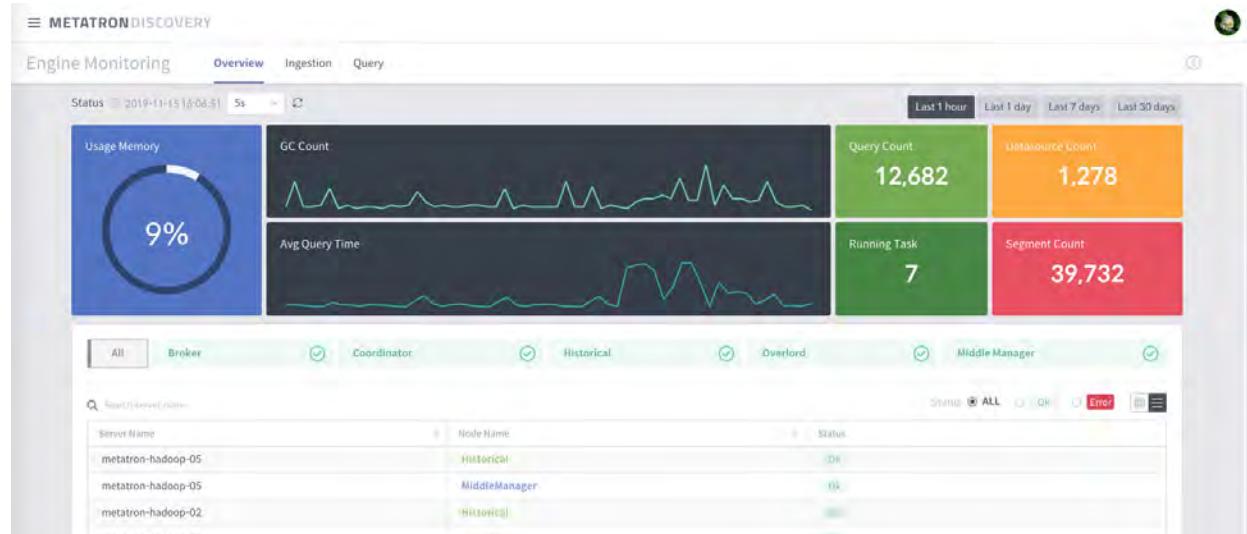


This feature is supported for Metatron Discovery 3.4.0 and above.

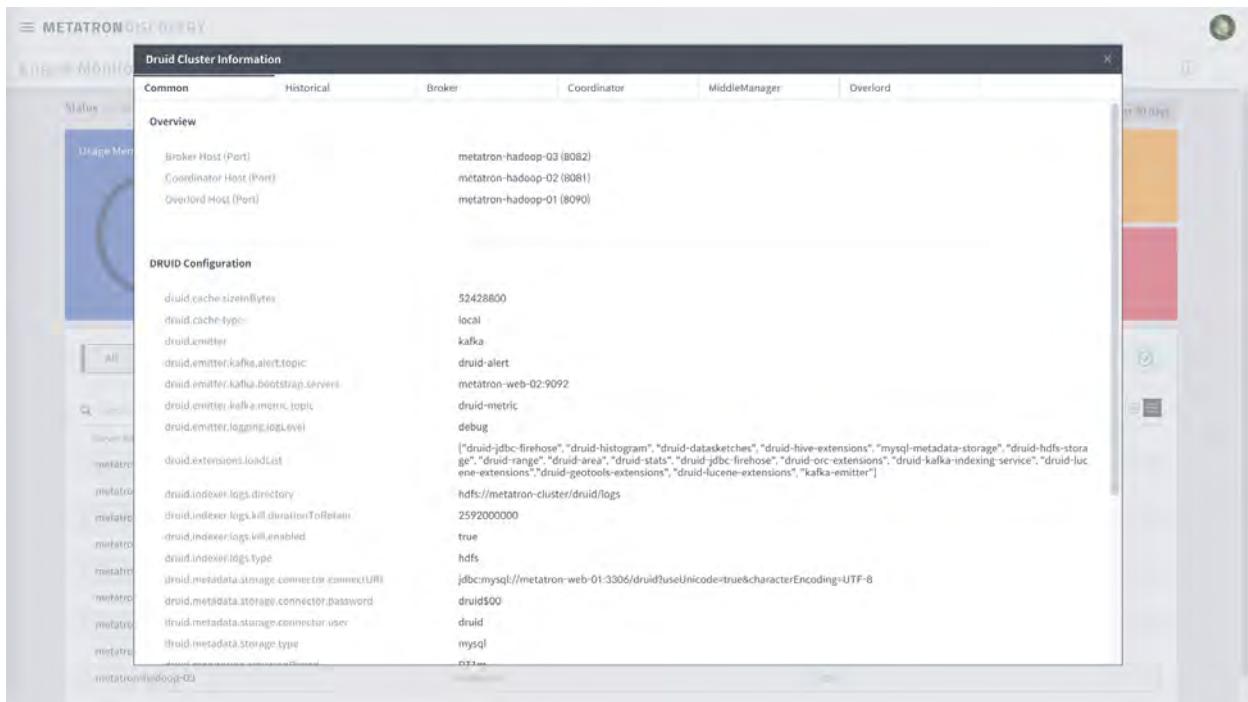
11.1 Overview

11.1.1 Druid Setting Configuration

You can check setting information of Druid here. On the right top side there is a information button(ⓘ). Click the button to check the details.



Below is the view that appears when you click the button. It shows the overall Druid setting information by common detail and 5 nodes section (Historical, Broker, Coordinator, Middle Manager, Overlord).



11.1.2 Historical Usage

Displays the usage of each historical node. Each server entries are acquired from the servers list of the Coordinator.

11.1.3 Cluster Total Usage

Provides Druid historical monitoring feature.

Cluster usage information identifies the following:

- Total usage of cluster
- Usage of each historical

Below is the KPI made by using the servers list of the Coordinator.

Field	Description	Example
Node Count	numbers of historical nodes	
MaxSize		
currSize		
Used		
FreeSize		

11.1.4 Historical Usage

Displays the usage of each historical node. Each server entries are acquired from the servers list of the Coordinator.

11.2 Ingestion

Ingestion is the monitoring of Druid Indexing Service. It provides performance status of the indexing tasks and related information.

Provided information identifies the following:

- **MiddleManager Status**
 - Capacity of each worker and current usage amount
- **Supervisor Status**
 - Status of each supervisor
 - provided feature: terminate (suspend, reset)
- **Task Status**
 - runningTasks, pendingTasks, waitingTasks, completedTasks
 - provided feature: log, kill
- **Lockbox Status**

Ingestion section displays details of both supervisor and middle manager.

11.2.1 Tasks

Tasks can be classified into 4 types of status:

- pending: task waiting to be assigned to a worker
- running: task currently running
- waiting: task waiting on lock
- completed: classified into two states - SUCCESS, FAIL

Task details and menu are as follows.

Field	Description	Example
id	taskId	
type		
dataSource		
createdTime		
queueInsertionTime		
status		
runnerStatusCode		
duration		
locationhost		
locationport		
payload		
status	status	
log		
log last 8k		
kill		
ingestion		

It is displayed as shown below.

Task ID	Status	Created time	Duration	Data source	Type
index_kafka_dacoe_flink_geo_f13596c212c22ed_bnamidian	RUNNING	2019-11-15 13:36:10.897	00:00:00	dacoe_flink_geo	kafka
index_kafka_dacoe_flink_1_1_0651d87a6709f50_idoamibh	RUNNING	2019-11-15 13:36:10.799	00:00:00	dacoe_flink_1_1	kafka
index_kafka_systemshockrealtimetest20190827_12_0420df52b114173_idfbimlk	RUNNING	2019-11-15 13:36:09.555	00:00:00	systemshockrealtimetest20190827_12	kafka
index_kafka_realtime_server_load_json_01_aad601ac12bb553_ngplhnbd	RUNNING	2019-11-15 12:54:57.041	00:00:00	realtime_server_load_json_01	kafka
index_kafka_stream_test_3_20583fdb514c5b_cmfmijf	RUNNING	2019-11-15 12:54:56.977	00:00:00	stream_test_3	kafka
index_kafka_systemshockrealtimetest20190827_12_0420df52b114173_nnnglcpm	RUNNING	2019-11-15 12:36:02.378	00:00:00	systemshockrealtimetest20190827_12	kafka
index_kafka_dacoe_flink_geo_f13596c212c22ed_hjhbmlbf	RUNNING	2019-11-15 12:36:02.378	00:00:00	dacoe_flink_geo	kafka
index_kafka_dacoe_flink_1_1_0651d87a6709f50_gfbelclff	RUNNING	2019-11-15 12:36:02.378	00:00:00	dacoe_flink_1_1	kafka
index_kafka_druid-metric_63bf28627d38b06_alkamjh	RUNNING	2019-11-15 05:16:47.928	00:00:00	druid-metric	kafka
index_kafka_druid-metric-topic_d70e2fb20fcfd77_goohalon	RUNNING	2019-11-14 17:47:02.250	00:00:00	druid-metric-topic	kafka
index_oivws_2019-11-15T04:30:05.0962	SUCCESS	2019-11-15 13:30:05.096	00:01:26	_oivws	index
index_batch_test_2019-11-15T04:30:04.118Z	SUCCESS	2019-11-15 13:30:04.118	00:00:08	batch_test	index
index_oivws_2019-11-15T04:20:04.839Z	SUCCESS	2019-11-15 13:20:04.839	00:01:50	_oivws	index
index_batch_test_2019-11-15T04:20:04.056Z	SUCCESS	2019-11-15 13:20:04.056	00:00:08	batch_test	index
index_oivws_2019-11-15T04:10:06.084Z	SUCCESS	2019-11-15 13:10:06.084	00:01:23	_oivws	index

Following image is the detail view. (a case using Kafka)

information

Queue (afterJoin) Time	2019-11-15T04:55:04.937Z
Created Time	2019-11-15T04:55:04.924Z
Pool	metatrond-hadoop-05
Label(s)	metatrond-hadoop-05-8105
Address(es)	stream.test.3
Type	kafka
Precision	0
Uptime (sec)	0
ThrownAway	0

Status (Log BK)

RUNNING

```

2019-11-15T04:55:13.201 INFO [main] com.sun.jersey.guice.spi.container.GuiceComponentProviderFactory - Binding io.druid.segment.firehouse.CharHandlerResource to GuiceInstantiatedComponentProvider
2019-11-15T04:55:13.201 INFO [main] com.sun.jersey.guice.spi.container.GuiceComponentProviderFactory - Binding io.druid.segment.firehouse.StatusResource to GuiceInstantiatedComponentProvider
2019-11-15T04:55:13.234 INFO [main] com.sun.jersey.guice.spi.container.GuiceComponentProviderFactory - Binding io.druid.segment.firehouse.LookupAnnotationsResource to GuiceInstantiatedComponentProvider
2019-11-15T04:55:13.237 INFO [main] com.sun.jersey.guice.spi.container.GuiceComponentProviderFactory - Binding io.druid.segment.firehouse.CharHandlerResource to GuiceInstantiatedComponentProvider
2019-11-15T04:55:13.242 INFO [main] com.sun.jersey.guice.spi.container.GuiceComponentProviderFactory - Binding io.druid.segment.lookup.LookupAnnotationsResource to GuiceInstantiatedComponentProvider
2019-11-15T04:55:13.245 INFO [main] com.sun.jersey.guice.spi.container.GuiceComponentProviderFactory - Binding io.druid.segment.lookup.LookupAnnotationsResource to GuiceInstantiatedComponentProvider
2019-11-15T04:55:13.251 INFO [main] com.sun.jersey.guice.spi.container.GuiceComponentProviderFactory - Binding io.druid.segment.lookup.LookupAnnotationsResource to GuiceInstantiatedComponentProvider
2019-11-15T04:55:13.251 INFO [main] com.sun.jersey.guice.spi.container.GuiceComponentProviderFactory - Binding io.druid.server.StatusResource to GuiceManagedComponentProvider with the scope: "undefined"
2019-11-15T04:55:13.280 WARN [main] com.sun.jersey.spi.inject.Errors - The following warnings have been detected with resource and/or provider classes:
WARNING: A HTTP GET method, public void io.druid.server.http.SegmentListerResource.getSegments(long,long,java.util.List), throws java.io.IOException, MUST return a non-void type.
2019-11-15T04:55:13.324 INFO [main] org.eclipse.jetty.server.handler.ContextHandler - Started o.e.j.s.ServletContextHandler@beec0f5/mAVAILABLE
2019-11-15T04:55:13.334 INFO [main] org.eclipse.jetty.server.Server - Started o.e.j.s.Server@dd359m
2019-11-15T04:55:13.316 INFO [main] org.eclipse.jetty.server.Server - Started o.e.j.s.ServerConnector@310b64f(HTTP/1.1)(http://1.1.0.0:8105)
2019-11-15T04:55:13.317 INFO [main] com.metamx.common.lifecycle.LifecycleAnnotationBasedHandler - Invoking start method/public void io.druid.query.lookup.LookupReferencesManager.start() on object@druid_query_lookup_LookupReferencesManager@6668ae27
2019-11-15T04:55:13.317 INFO [main] io.druid.query.lookup.LookupReferencesManager - Internal lookup factory references manager
2019-11-15T04:55:13.317 INFO [main] io.druid.query.lookup.LookupReferencesManager - Invoking start method/public void io.druid.server.listener.ListenerResourceAnnouncer.start() on object@druid_query_lookup_LookupReferencesManager@6668ae27
2019-11-15T04:55:13.324 [INFO] [main] io.druid.server.listener.ListenerResourceAnnouncer - Announcing start time on /druid/listeners/lookup/_default/metatrond-hadoop-05:8105

```

ingestion RUNNING



And below is a case of general Task, not using Kafka.

Information

- Queue insertion Time: 1970-01-01T00:00:00.000Z
- Created Time: 2019-11-15T05:30:06.027Z
- Host: oivws
- Type: index

Status (Log 8K)

SUCCESS

```

at io.druid.emitter.KafkaEmitter$1.run(KafkaEmitter.java:161) [kafka-emitter-0.9.1-SNAPSHOT]
at io.druid.emitter.Kafka.KafkaEmitter$emitMetricToKafka(KafkaEmitter.java:165) [kafka-emitter-0.9.1-SNAPSHOT]
at io.druid.emitter.Kafka.KafkaEmitter.access$500(KafkaEmitter.java:51) [kafka-emitter-0.9.1-SNAPSHOT]
at io.druid.emitter.Kafka.KafkaEmitter$2.run(KafkaEmitter.java:136) [kafka-emitter-0.9.1-SNAPSHOT]
at io.druid.emitter.Kafka.KafkaEmitter$2.run(KafkaEmitter.java:136) [kafka-emitter-0.9.1-SNAPSHOT]
at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) [?:1.8.0_171]
at java.util.concurrent.FutureTask.run(FutureTask.java:266) [?:1.8.0_171]
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:630) [ScheduledThreadPoolExecutor.java:180] [?:1.8.0_171]
at java.util.concurrent.ThreadPoolExecutor$Worker$Worker.run(ThreadPoolExecutor.java:294) [?:1.8.0_171]
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:149) [?:1.8.0_171]
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) [?:1.8.0_171]
at java.lang.Thread.run(Thread.java:748) [?:1.8.0_171]
2019-11-15T05:31:33+09:00 INFO [main] org.apache.kafka.clients.producer.KafkaProducer - Invoking stop method/public void void io.druid.emitter.kafka.KafkaEmitter.close() on object(io.druid.emitter.kafka.KafkaEmitter@11d2714a).
2019-11-15T05:31:33+09:00 INFO [main] org.apache.kafka.clients.producer.KafkaProducer - Closing the Kafka producer with timeoutMillis = 9223372036854775807 ms.
2019-11-15T05:31:33+09:00 INFO [main] io.druid.cli.CliPhoen - Finished pean task
Heap
garbage-first heap total 1030192K used 521158K (0x0000007000000000, 0x0000000070010f80, 0x000000007c000000)
region size 1024K
last compaction point 522240K, 22 survivors (2240K)
max compacted 42543K, capacity 67962K, committed 64128K, reserved 1775920K,
class space used 7715K, capacity 8059K, committed 8044K, reserved 1048576K.

```

Ingestion SUCCESS

11.2.2 Supervisors

You can monitor the running Supervisors. Details and menu available for monitoring is as follows:

Field	Description	Exam- ple
Status	All of the supervisors provided by ‘get supervisorIDs’ are at running state	
Datasource		
Detailed Status	Details provided by status API	
Lag	Lag details of kafka, acquired using emitter	
Spec		
Shutdown	Terminates supervisor. Kills related tasks as well.	

It is displayed as shown below.

The screenshot shows the METATRON DISCOVERY interface with the 'Supervisor' tab selected. A table lists various supervisor IDs along with their corresponding topics, data sources, and other metadata. The table includes columns for Supervisor ID, Topic, Datasource, and several other fields. A search bar and a 'Show up to 20' button are visible at the bottom right of the table area.

Supervisor ID	Topic	Datasource
realtime_server_load_json_01	realtime_server_load_json	realtime_server_load_json_01
systemshockrealmtest20190827_12_d4b1541cf684fc78a75d99a7e87cb0a	realtime_sample_12	systemshockrealmtest20190827_12
stream_test_3	druid-alert-testbed	stream_test_3
systemshockrealmtest20190926_1_b3586bb3da924f4b8004a5e0c61fe6a5	realtime_sample_4	systemshockrealmtest20190926_1
druid-metric_9c6a65cf396446a597ba35767770e7df	druid-metric	druid-metric
dacoe_flink_1_1_ba83f75e45f43cf947fd94928007afdf	dacoe_flink_1	dacoe_flink_1_1
druid-metric_topic_e150b3f1d0c641dabc5b71abaceee90b	droid-metric-topic	druid-metric-topic
systemshockrealmtest20190926_2_58da1b76127f49b38185b8bd88e71435	realtime_sample_20190926_02	systemshockrealmtest20190926_2
dacoe_flink_geo_7962ed01dac46b89978fbfce8af49c9	dacoe_flink_1	dacoe_flink_geo

The screenshot shows the METATRON DISCOVERY interface with the 'Supervisor' tab selected. It displays detailed information for a specific supervisor and a 'LAG' chart. The 'Information' section shows the topic and datasource. The 'LAG' chart displays the lag over time, with a prominent peak labeled '2019-11-15 15:45' and 'LAG : 2823'. The 'Active Tasks' section shows a task ID and a 'Details' button.

Information

- Topic: realtime_sample_12
- Datasource: systemshockrealmtest20190827_12

LAG

2019-11-15 15:45
LAG : 2823

Active Tasks

Task ID: Index_kafka_systemshockrealmtest20190827_12_0420df52b114173_idxname | Details

11.2.3 MiddleManagers

List of workers.

The screenshot shows the METATRON DISCOVERY interface with the 'Ingestion' tab selected. Under the 'Middle Manager' sub-tab, there is a table listing two workers:

Worker Host	Worker IP	Version	Capacity(Used/Total)	Availability Groups	Running Tasks	Completed Time
metatron-hadoop-04:8091	localhost	0	4/10	4	4	2019-11-15 13:56:05,217
metatron-hadoop-05:8091	localhost	0	6/10	6	6	2019-11-15 13:51:55,992

Below the table, there is a search bar and a button to 'Show up to 20' results.

The screenshot shows the METATRON DISCOVERY interface with the path 'metatron-hadoop-04:8091'. Under the 'information' tab, it displays the following details for the worker:

- Host: metatron-hadoop-04:8091
- IP: localhost
- Capacity: 4/10
- Version: 0
- Availability Groups: 4
- Running Tasks: 4

Under 'Running Tasks', there is a list of tasks:

- index_kafka_realtime_server_load_json_01_ba4ad60c3804abb_bdmldbk
- index_kafka_dacoe_flink_1_1_0651d87a6709f50
- index_kafka_stream_test_3_db5459136c28e81_gffifgih
- index_kafka_druid-metric_63bff28627d3806_gmckohgp

At the bottom, it shows the last completed task time: 2019-11-15T07:01:32.648Z.

11.3 Query

The screenshot shows two views of the Metatron Discovery Engine Monitoring interface.

Top View (List of Queries):

ID	Result	Service	Host	Type	DataSource	Started time	Duration (ms)
43a0b41f-a226-491c-8e3a-c5cd7397262f	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	ettest	2019-11-15 14:31:59.360	1
ec81be45-d6e2-4094-9135-9d6750186003	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	stcaz	2019-11-15 14:31:59.142	1
13d75e7b-5965-41e6-845f-5769dc54208a	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	JustATestName	2019-11-15 14:31:58.928	0
bdcdb5d-3541-4040-a2f2-c662569169f4	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	dd11	2019-11-15 14:31:58.721	0
94d9b85a-3250-4c95-89cf-a01d2c92e177	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	hive_preset_engine_dri...	2019-11-15 14:31:58.504	3
68185af4-7884-49f7-90fc-6f2beb795936	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	stage_part_test2	2019-11-15 14:31:58.274	0
394da03a-b9e6-4aed-81d8-598b52d01d78	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	dss_parete1	2019-11-15 14:31:58.060	0
77004272-dad1-4106-b0cc-1898ecd8929a	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	order	2019-11-15 14:31:57.828	0
32f9b133-bb91-45ad-9283-45c87b34c0f4	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	gprgg	2019-11-15 14:31:57.606	0
49df052b-b4ef-42ad-9df4-8624e1d42bd3	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	네한민국 시군구별 multip...	2019-11-15 14:31:57.389	0
33b63a6f-3849-47d5-a8ac-26b4054858a8	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	hhhh_vhyad	2019-11-15 14:31:57.168	0
ff000db3-96db-4c55-ac89-8798e1853b8a	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	Query Parsed(eventtime)	2019-11-15 14:31:56.952	1
6a2b0d2d-2461-4192-acca-5c7426f1ae90	Success	druid/prod/broker	metatron-hadoop-03:8082	segmentMetadata	correlation matrix for m...	2019-11-15 14:31:56.732	0

Bottom View (Detailed Query Information):

Query ID	43a0b41f-a226-491c-8e3a-c5cd7397262f
Result	Success
Service	druid/prod/broker
Host	metatron-hadoop-03:8082
Type	segmentMetadata
DataSource	ettest
Started Time	2019-11-15 14:31:59.360
Duration	1 ms

Part II

EX-pack for Workflow Integrator

INTRODUCTION OF INTEGRATOR EXPANSION PACK

The Integrator Expansion Pack provides a GUI for easier control over Apache Oozie, the workflow scheduling system for Hadoop jobs. It is a module that processes data in the workflow for use in Metatron Discovery. Users can easily design and set up a routine to repeatedly perform Hadoop jobs, thereby obtaining data required for Metatron Discovery tasks on a regular basis.

The key features of the Integrator Expansion Pack are as follows:

Editing and scheduling a workflow simultaneously

The intuitive chart editor can be used to easily create workflows and schedule runs.

Managing multiple clusters at once

The source of raw data and the destination table can be freely designated for each node in the workflow, by which multiple clusters can be managed at once.

Workflow sharing

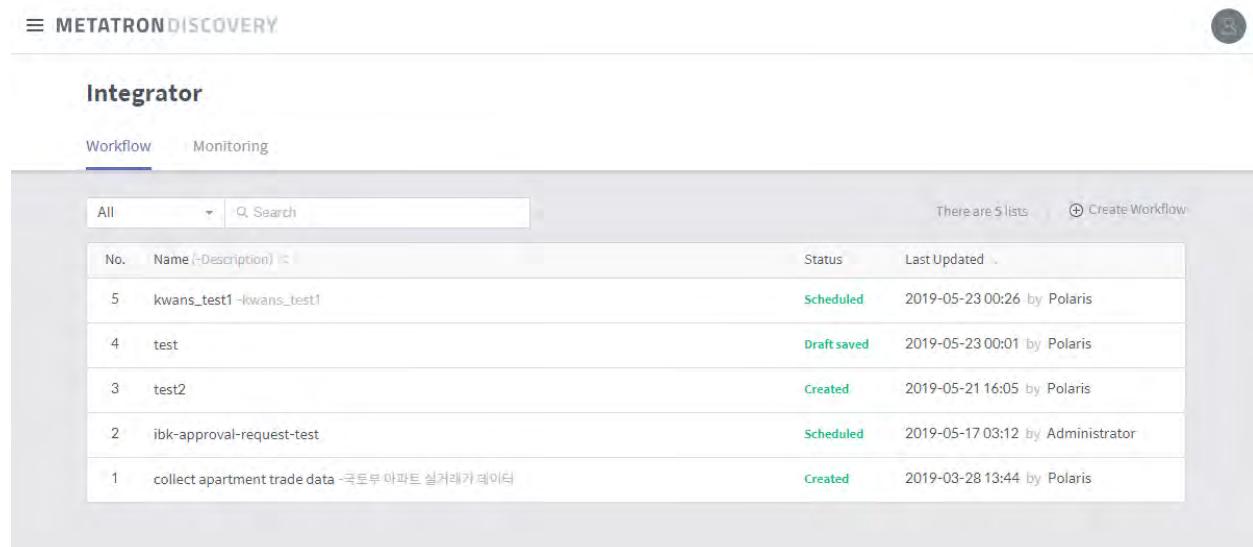
Established workflows can be shared and managed by multiple users within your organization.

Alarms and reports

The result of executing a reserved workflow is reported through various channels such as SMS, e-mail, and messenger.

WORKFLOW LIST

The **Workflow** tab on the main page of Integrator lists registered workflows as shown below. The **Status** column gives the brief progress of each workflow.



The screenshot shows the 'Integrator' section of the Metatron Discovery interface. The 'Workflow' tab is selected. At the top, there is a search bar labeled 'Q, Search' and a button for creating a new workflow labeled '+ Create Workflow'. Below the search bar, there is a table with columns for 'No.', 'Name (-Description)', 'Status', and 'Last Updated by'. The table contains six rows of workflow data:

No.	Name (-Description)	Status	Last Updated by
5	kwans_test1 -kwans_test1	Scheduled	2019-05-23 00:26 by Polaris
4	test	Draft saved	2019-05-23 00:01 by Polaris
3	test2	Created	2019-05-21 16:05 by Polaris
2	ibk-approval-request-test	Scheduled	2019-05-17 03:12 by Administrator
1	collect apartment trade data -국토부 아파트 실거래가 데이터	Created	2019-03-28 13:44 by Polaris

Click on one of the workflows in the list to enter the workflow editor. See [Workflow editor](#) for details on the workflow editor.

Click **+ Create Workflow** on the upper right to open a dialog box to create a new workflow. Enter the name and description of the workflow, and click **Done** to create the new workflow.

The screenshot shows the Metatron Discovery Integrator interface. At the top, there is a navigation bar with the text "METATRON DISCOVERY" and a user icon. Below the navigation bar, the main title "Integrator" is displayed, followed by two tabs: "Workflow" (which is selected) and "Monitoring".

The main content area shows a list of workflows. A search bar at the top of the list allows filtering by "Name / Description". The list includes the following items:

No.	Name / Description	Status	Last Updated
5	kwans_test1 - kwans test	Scheduled	2019-05-23 00:26 by Polaris
4			
3			
2			
1			

A modal dialog box titled "Create Workflow" is open in the center. It contains fields for "Name" and "Description", both of which have placeholder text "Please enter a name" and "Please enter a description" respectively. At the bottom of the dialog are two buttons: "Cancel" and "Done".

WORKFLOW EDITOR

Through the GUI of the workflow editor, you can conveniently edit the selected Hadoop workflow and schedule runs. Click one of the workflows listed in [Workflow list](#) to enter the workflow editor. The following is displayed.

1 Workflow node selection area: Choose nodes to add to the workflow. Click to expand the panel and view the names of all nodes. The nodes are categorized into two types.

2 Please enter a description

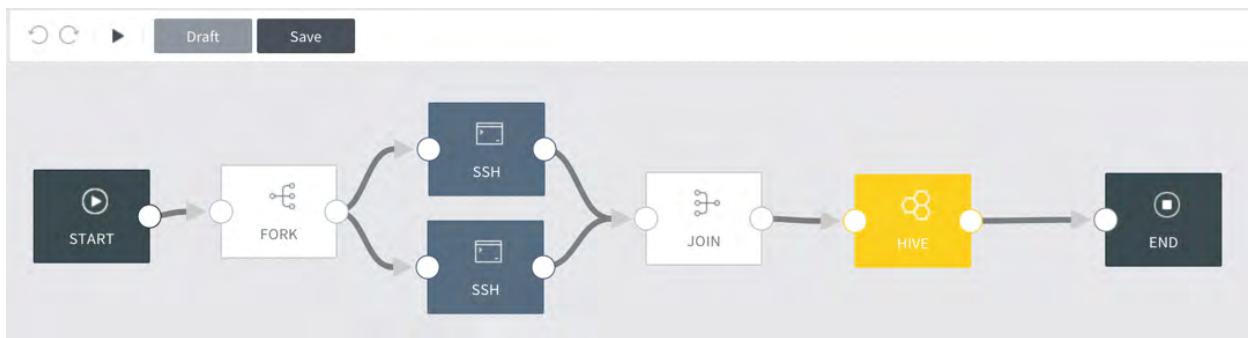
3 Workflow settings

4 Manual run

Job ID	Start Time	Elapsed Time	Status
0000116-190416235342967-oozie-oozi-W	2019-04-19 11:37:26	25 sec	SUCCEEDED
0000115-190416235342967-oozie-oozi-W	2019-04-19 11:37:05	0 sec	FAILED
0000114-190416235342967-oozie-oozi-W	2019-04-19 11:36:10	0 sec	FAILED
0000002-190305012223848-oozie-oozi-W	2019-03-05 11:41:27	27 sec	SUCCEEDED
0000001-190305012223848-oozie-oozi-W	2019-03-05 11:25:54		RUNNING
0000000-190305012223848-oozie-oozi-W	2019-03-05 10:28:16		KILLED
0000001-190305004451829-oozie-oozi-W	2019-03-05 10:08:05	28m 14s	KILLED
0000000-190305004451829-oozie-oozi-W	2019-03-05 10:02:00		SUSPENDED

1. **Workflow node selection area:** Choose nodes to add to the workflow. Click to expand the panel and view the names of all nodes. The nodes are categorized into two types.

- **Action nodes (categorized as “Task” in editor):** Define tasks involved in collecting, processing, and ingesting raw data in the Hadoop cluster. See [Action nodes](#) for details.
 - **Control flow nodes (categorized as “General” in editor):** Define the start and end of a workflow and determine the flow path of action nodes. See [Control flow nodes](#) for details.
2. **Workflow chart canvas:** The sequence between added nodes is defined. As shown in the figure below, drag the desired nodes to the canvas, and connect the nodes according to the desired sequence to complete the workflow chart.



Undo or redo actions using the buttons on the top, and click to run the current workflow. And click the **Draft** button to save the current workflow, and the **Save** button to save it as the actual workflow.

3. **Workflow settings area:** Set up the task details of individual nodes selected in the workflow chart canvas. See relevant node items in [Action nodes](#) and [Control flow nodes](#) for details.
4. **Workflow run details area:** View the run details of the defined workflow.

- **Manual run tab:** Click on the top left of the editor to view the details of manual runs.
- **Scheduled run tab:** Schedule workflow runs at desired times using the UI, and view the details of scheduled runs. See [Schedule a workflow run](#) for details.

Below is a comprehensive list of topics on using the workflow editor.

14.1 Action nodes

Action nodes in Integrator define tasks involved in collecting, processing, and ingesting raw data in the Hadoop cluster. The supported Hadoop jobs and individual system tasks (Java, Shell, etc.) are as follows:

- [Sqoop](#)
- [MR](#)
- [EXEC](#)
- [Java](#)
- [HIVE Query](#)
- [SSH](#)
- [Spark](#)
- [Sub-Workflow](#)
- [DistCp](#)
- [HDFS](#)
- [Done](#)
- [Druid](#)

14.1.1 Sqoop

Retrieves data from RDP or runs a simple query.

14.1.2 MR

Runs JAR files in a local directory.

14.1.3 EXEC

Runs local files such as Python and shell.

14.1.4 Java

Runs a Java class. (Note that the main function must be defined.)

14.1.5 HIVE Query

Runs a HIVE query.

14.1.6 SSH

Runs a command remotely. Note that SSH passwordless login must be set up for the remote server.

14.1.7 Spark

Runs SPARK.

14.1.8 Sub-Workflow

Used for association with existing workflows. When running an association of multiple workflows, it defines each workflow as a task.

14.1.9 DistCp

Copies files from the source Hadoop cluster to the target Hadoop cluster.

14.1.10 HDFS

Used to manage Hadoop files.

14.1.11 Done

Creates a Done file upon completion.

14.1.12 Druid

Used for incremental ingestion of data into the Druid engine.

14.2 Control flow nodes

The control flow nodes of Integrator define the start and end of a workflow and determine the flow path of action nodes. The supported nodes are as follows:

- Start
- End
- Decision
- Fork
- Join

14.2.1 Start

The start point of all workflows. Required to run a workflow.

14.2.2 End

The end point of all workflows. Required to end a workflow.

14.2.3 Decision

Branches the workflow based on conditions. It uses as many switch case statements as the number of branches.

14.2.4 Fork

Branches the workflow without conditions for concurrent, parallel execution.

14.2.5 Join

Joins several nodes.

14.3 Schedule a workflow run

Workflow runs can be scheduled to repeatedly run a workflow at certain intervals. The results of scheduled runs can be reported through SMS, messenger, and e-mail.

14.3.1 List of scheduled runs

Click the **Scheduled run** tab in the run details area on the bottom right of the workflow editor, and a list of scheduled runs will be displayed as follows. The list displays the run status of each scheduled run. Click ► to execute the scheduled run, and ✎ to delete.

Name(-Description)	Status	Updated
kwans_test1_sche1_C05231136_C05231140	CREATED	2019-05-23 23:26:46 by Polaris
kwans_test1_sche1_C05231136	CREATED	2019-05-23 20:36:34 by Administrator
kwans_test1_sche1	CREATED	2019-05-23 01:04:10 by Polaris

14.3.2 Add a scheduled run

Click + **Create execution schedule** in the scheduled run area. A dialog box to create a new scheduled run is displayed as follows. Fill out each field as instructed below, and click **Create**.

Create a New Execution Schedule

Name	<input type="text" value="Please enter a name"/>
Description	<input type="text" value="Please enter a description"/>
Tags	# <input type="text" value="Please enter a tag"/>
Workflow	<input type="text" value="kwans_test1"/>

Period	From <input type="text" value="2019-06-10 00:00"/>	To <input type="text" value="2020-06-10 23:59"/>
Frequency	<input type="text" value="Daily"/>	<input type="text" value="00:00"/>
Concurrency	<input type="text" value="1"/>	
Timeout(min)	<input type="text" value="Please enter a timeout unit (by minute)"/>	

Datasets

⊕ Add

Configuration (Move to Metabase)

<input type="checkbox"/>	<input type="text" value="Key"/>	<input type="text" value="Value"/>
--------------------------	----------------------------------	------------------------------------

⊕ Add

Variables (Move to Configuration)

<input type="checkbox"/>	<input type="text" value="Key"/>	<input type="text" value="Value"/>
--------------------------	----------------------------------	------------------------------------

⊕ Add

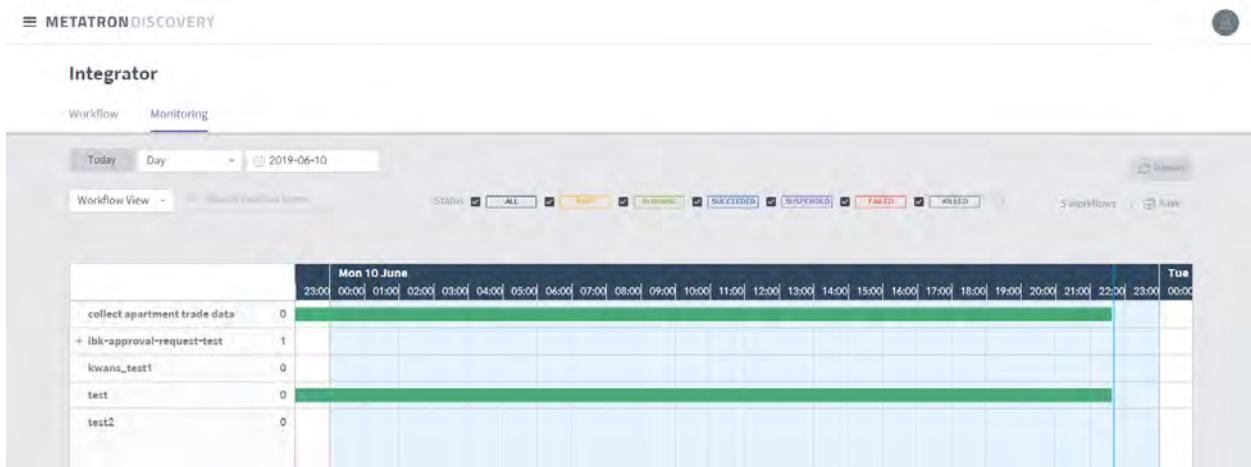
Alert OFF (Move to Configuration)

- **Name:** Enter a name for the scheduled run.
- **Description:** Describe the scheduled run.
- **Tags:**

- **Workflow:** Select a workflow to schedule to run.
- **Period:** Set the start and end times of the scheduled run.
- **Frequency:** Set the frequency of the scheduled run.
- **Concurrency:**
- **Timeout (min):**
- **Datasets:**
- **Configuration:**
- **Variables:**
- **Alert:**

MONITORING

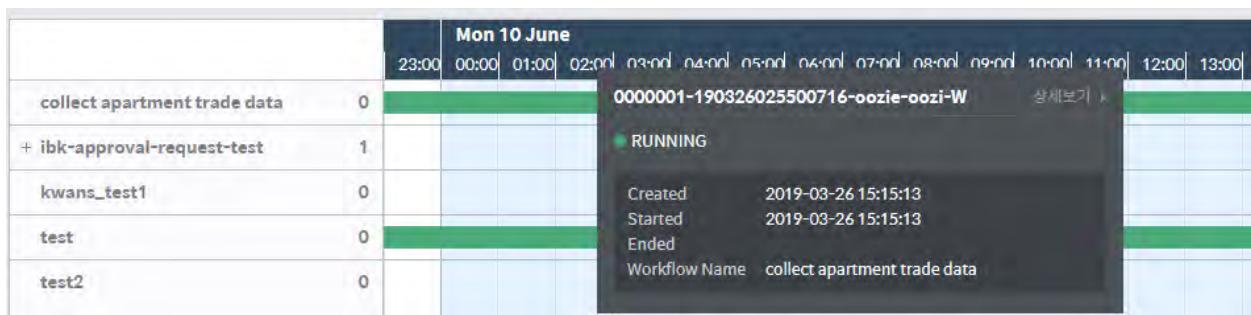
The **Monitoring** tab on the main page of Integrator displays runs and schedule information in graph form for each workflow.



The status bars of the graph represent scheduled or manual runs, and related information is presented as follows:

- Position and length: The status bar spans the duration of the run represented on the timeline.
- Color: The status bar is displayed in the same color as the color of the **Status** item in the top legend. For example, a status bar in green indicates that the run is ongoing.

Hovering the cursor over the status bar displays the run details as shown below. Click **View details** on the top right of the dialog box to view more detailed information.



16.1 Hand over data source ingestion

- Background processing to prevent system overload when ingesting huge amounts of data

16.2 Linkage with workbench

- Repeated execution of specific queries
- Handing over the execution of time-consuming queries

16.3 Linkage with data preparation

- Repeated use of wrangled datasets

Part III

EX-pack for Anomaly Detection

INTRODUCTION OF METATRON ANOMALY

The Anomaly Expansion Pack is a tool that detects abnormal data flow and immediately alerts users. For this detection, it uses prediction models built based on machine learning.

17.1 Basic principles

As shown below, Anomaly predicts an aggregate of the target data source in real time and monitors the actual value.



Here, the value marked as **Predict** is the data aggregate predicted through machine learning, and the value marked as **Actual** is the actual monitored value. As shown below, the **total abnormal score** increases with the difference between the two values. That is, the data aggregate is considered as deviating from the normal range if the actual value is significantly different from the predicted value.



In this example, it is set to generate a low level alarm when the abnormal score reaches 20 points, a moderate if it exceeds 40 points, a major alarm if it exceeds 60 points, and a critical level alarm if it exceeds 80 points.” According to the training data, It can be predicted that a critical class alarm was generated on April 6th at 3pm.

The alarms are reported through various channels to the user, so that immediate action can be taken in response to anomalies.

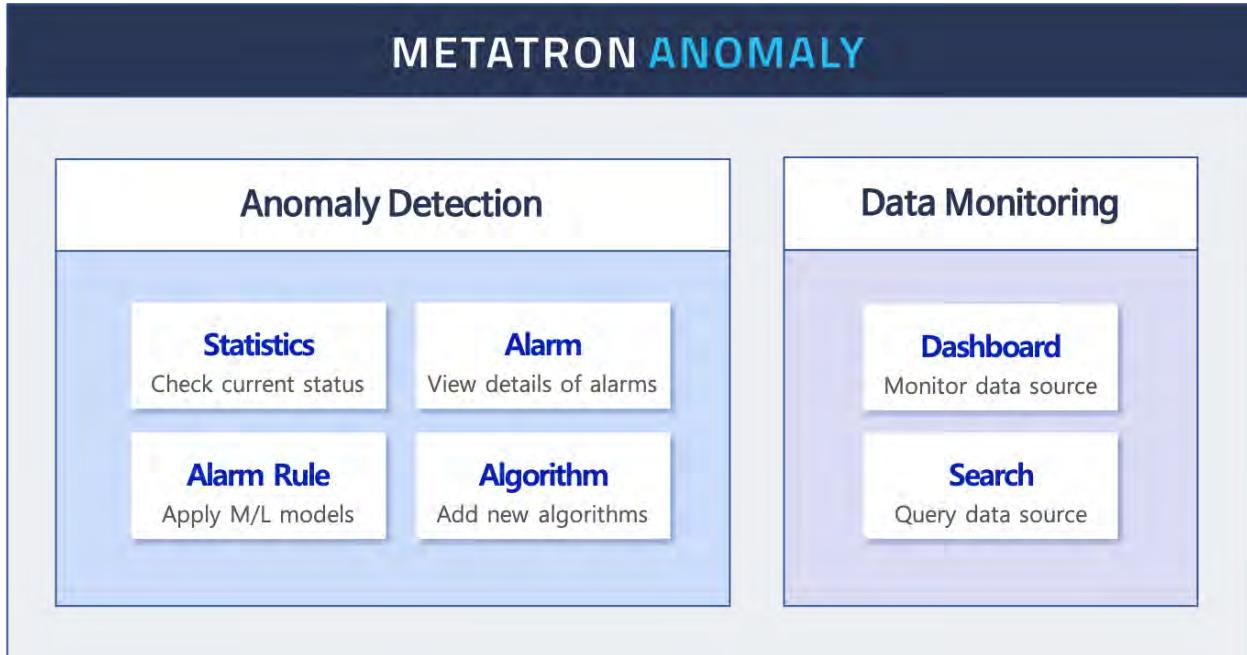
17.2 Key functions

The key functions of Anomaly are as follows:

- User convenience enhanced with automatic recommendation of a prediction model based on machine learning
- Immediate alarm triggering and report generation in case of anomaly
- Support real-time dashboard and real-time search function to analyze data source
- Support 3rd-party system linkage to apply new algorithm model

17.3 Structure

Anomaly's menu is divided into two categories: Anomaly Detection and Data Management.



Under **Anomaly Detection** menu, features support overall anomaly detection statistics, alarm information, alarm rule setting, and new algorithm addition.

Under **Data Monitoring** menu, features provide a real-time dashboard and a search function that allows you to query the data source.

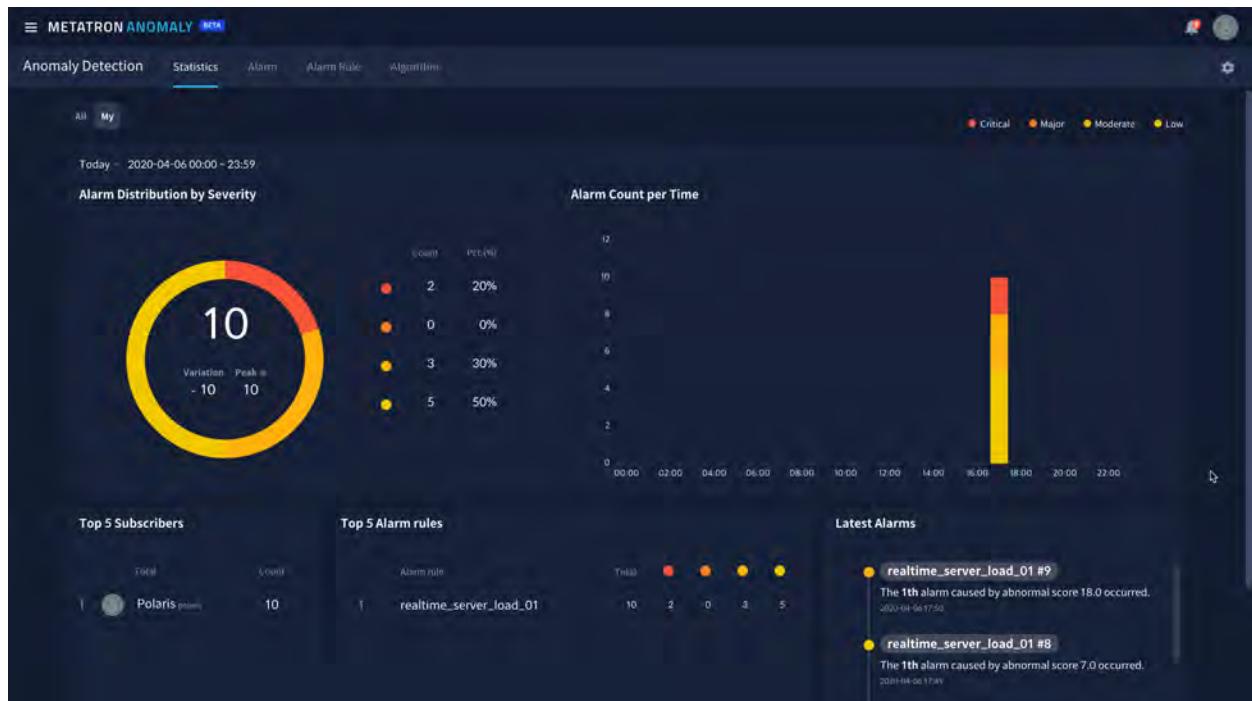
Users can easily navigate across menus, use references to detailed items, and gain organic understanding of alarms including their rule settings, past occurrences, and overall statistics.

EIGHTEEN

STATISTICS

The **Statistics** tab menu shows the overall statistics of the alarms that have occurred. This page allows statistics by various criteria such as importance, when the alarm occurred, and alarm rules so that the user can grasp the current status of the alarm from various angles. Calculate and present.

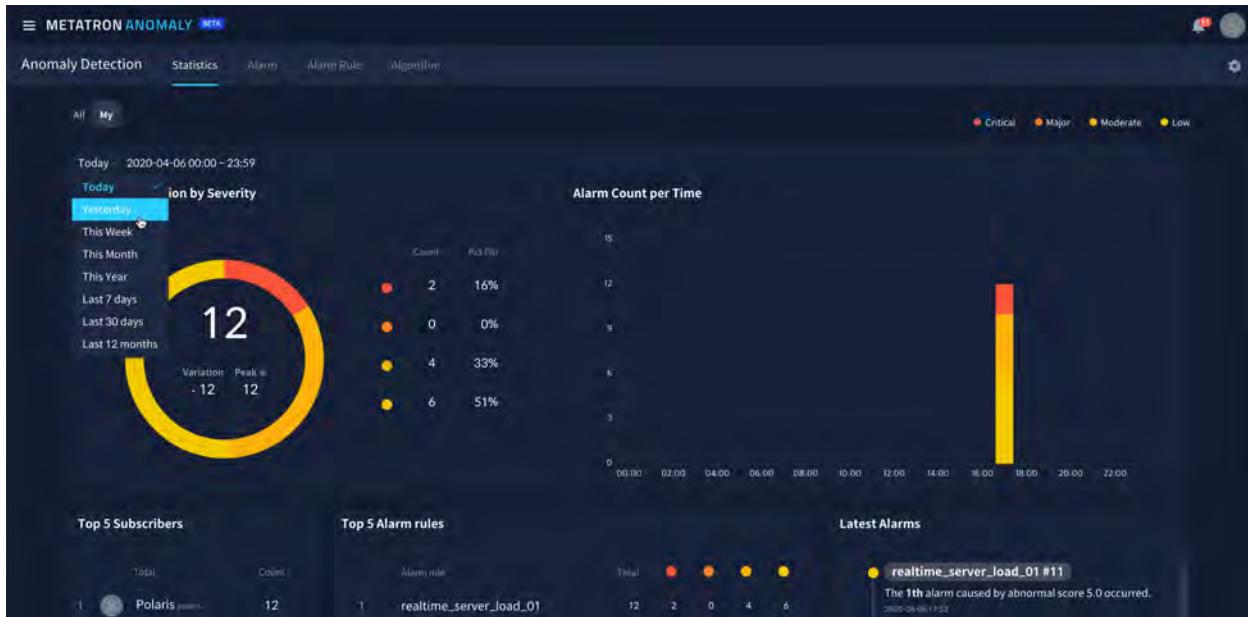
The basic structure of the page is as follows.



- **Alarm Distribution by Severity:** It shows the proportion of alarm occurrence by severity.
- **Alarm Count per Time:** Shows alarm frequency per time zone.
- **Top 5 Subscribers:** 가장 많은 알람을 통보받은 사용자 5명을 보여줍니다.
- **Top 5 Subscribers:** Shows 5 users who are notified of the most alarms.

- **Latest Alarms:** Shows the most recent alarms.

You can change the standard period for calculating statistics using the period setting menu at the top of the page.



ALARM

In the **Alarm** tab menu, you can check the alarm history that has occurred so far. Unlike the: ref: Statistics page, which shows the overall status of the alarm, this menu provides an optimized UI for viewing and browsing more individual alarms.

This tab consists of the following two pages.

- [Alarm List](#)
- [Alarm Details](#)

19.1 Alarm List

When entering the **Alarm** tab, the alarms that have occurred so far are listed and displayed. Using the **Alarm rule / Timeline** selection box at the top of the screen, you can sort the alarm list by alarm rule or by the time that occurred.

- [Alarm rule \(sort by alarm rule\)](#)

The screenshot shows the METATRON ANOMALY interface. At the top, there's a navigation bar with tabs for Anomaly Detection, Statistics, Alarm, Alarm Rule, and Algorithm. Below the navigation bar, there are filters for My, Alarm rule, Severity (All), Occurrence Date (All days), Show Unchecked Only, and a search bar.

Latest Alarms:

- realtime_server_load_01 #1... (Low, 61 sec)
- realtime_server_load_01 #1... (Low, 60 sec)
- realtime_server_load_01 #1... (Low, 60 sec)
- realtime_server_load_01 #1... (Moderate, 60 sec)
- realtime_server_load_01 #1... (Moderate, 60 sec)
- realtime_server_load_01 #1... (Low, 60 sec)

Score type check rule:

- Score type check rule #5 (Low, 1 min)
- Score type check rule #4 (Moderate, 2 min)
- Score type check rule #3 (Low, 2 min)
- Score type check rule #2 (Critical, 2 min)
- Score type check rule #1 (Critical, 2 min)

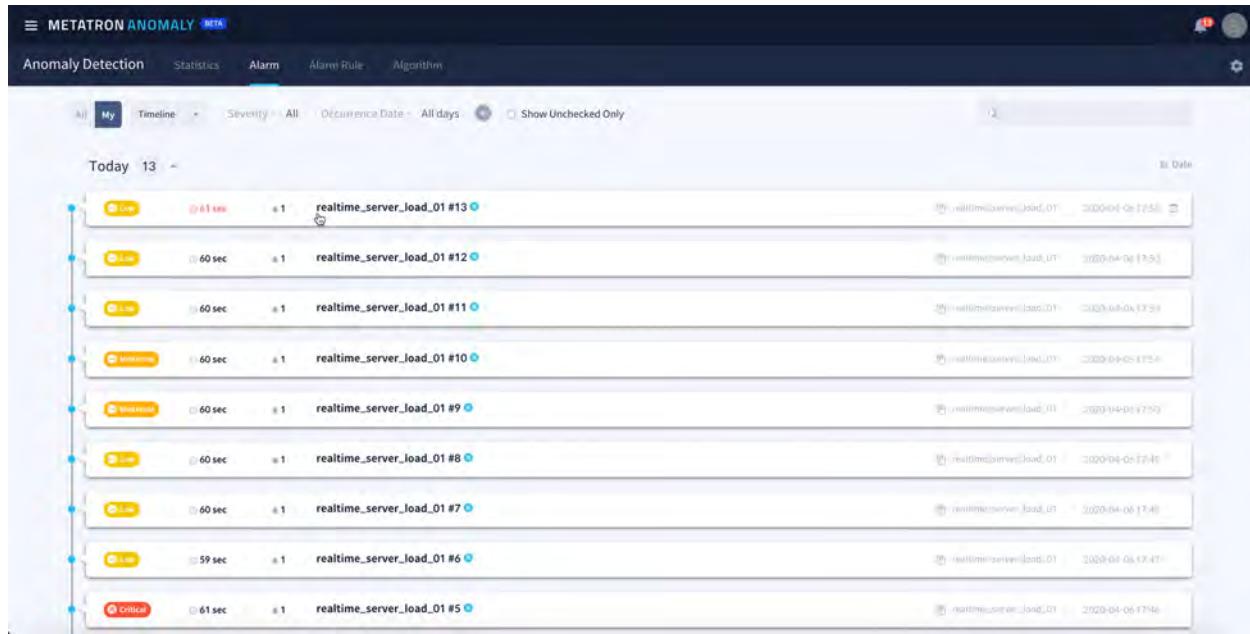
- **Timeline (sort by occurrence time)**

This screenshot shows the same METATRON ANOMALY interface as the previous one, but the data is sorted by occurrence time. The latest alarms are now:

- realtime_server_load_01 #1... (Low, 60 sec)
- realtime_server_load_01 #1... (Critical, 60 sec)
- realtime_server_load_01 #1... (Critical, 59 sec)
- realtime_server_load_01 #1... (Low, 61 sec)
- realtime_server_load_01 #1... (Low, 60 sec)
- realtime_server_load_01 #1... (Low, 60 sec)

The score type check rules remain the same as in the first screenshot.

Click **+ Load more** at the end of a category to show more alarm entries in that category.



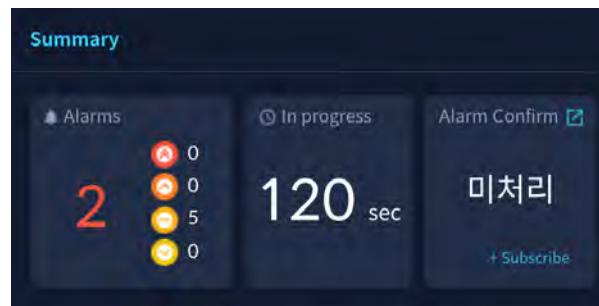
19.2 Alarm Details

Select one of the items listed in the alarm list to view detailed information about the alarm. Below is the description of each area of the alarm detail page.

19.2.1 Summary

이 영역에서는 해당 알람의 발생 현황을 보여줍니다. 정해진 주기에 따라 알람이 연속적으로 발생하면 1개의 알람 항목으로 계속 유지되며, 알람의 심각도 (severity) 기준을 넘은 데이터 포인트 수가 함께 표기됩니다. 또한 알람을 확인한 후 처리 결과를 기록할 수 있도록 링크를 제공합니다. 내가 생성한 알람 룰이 아닌 경우 **Subscribe**을 눌러 해당 알람 룰로 추후 발생한 알람들에 대해 알림을 받을 수 있습니다.

The example below shows that the alarm occurred twice in a row (**Alarms**), and because the alarm check interval is 1 minute, two alarms lasted for a total of 120 seconds. (**Elapsed Time**).



19.2.2 Alarm History

이 영역에서는 해당 알람에 적용된 알람 룰에 의해 발생한 알람의 이력을 보여줍니다. 우측 링크 아이콘을 누르면 해당 알람으로 이동합니다.

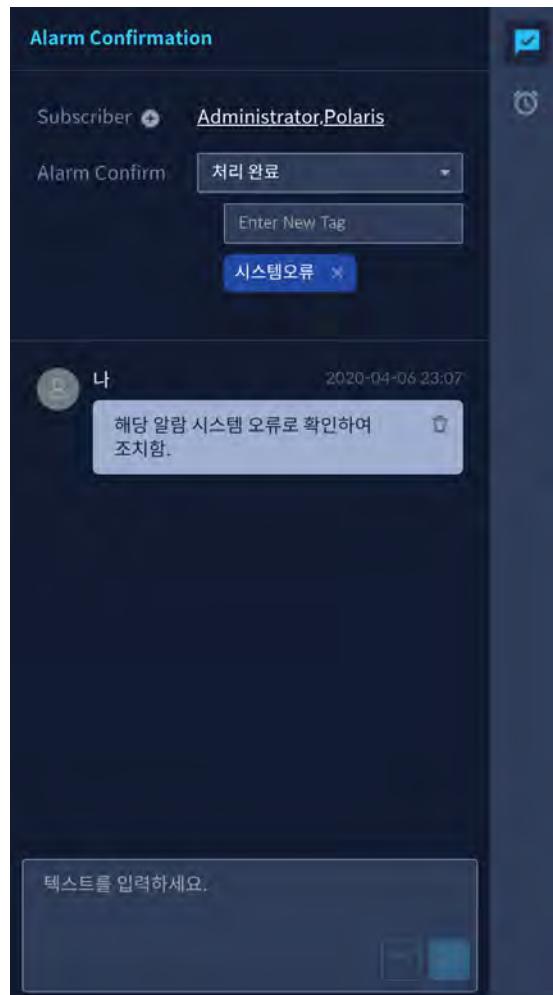
Severity	모든타입	There are 15 items			
NO	Occurrence time	Alarm Interval	Alarm		
211	2020-04-06 23:26 60 sec	1	1	<input checked="" type="checkbox"/>	Link
210	2020-04-06 23:25 60 sec	1	1	<input checked="" type="checkbox"/>	Link
209	2020-04-06 23:24 60 sec	1	1	<input checked="" type="checkbox"/>	Link
208	2020-04-06 23:23 120 sec	1	2	<input checked="" type="checkbox"/>	Link
207	2020-04-06 23:21 60 sec	1	1	<input checked="" type="checkbox"/>	Link
206	2020-04-06 23:20 60 sec	1	1	<input checked="" type="checkbox"/>	Link
205	2020-04-06 23:19 60 sec	1	1	<input checked="" type="checkbox"/>	Link

19.2.3 Alarm Confirmation

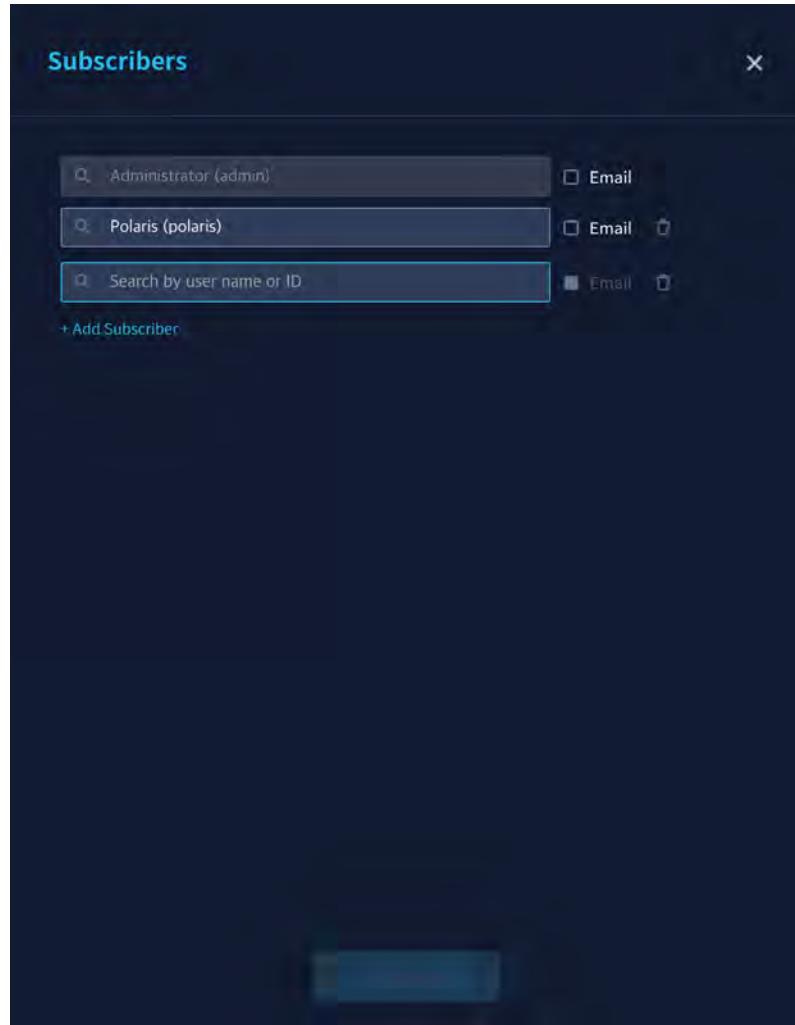
우측 탭 첫번째 메뉴에서는 알람 확인 후 해당 알람 구독자 리스트를 확인하고 (Subscriber) 알람을 확인하여 상태를 기록하고 (Alarm Confirm) 작업자가 기록을 남길 수 있는 커뮤니케이션 기능을 제공합니다.

There are four types of alarm confirmation items.

- **미처리:** 알람 최초 발생 시 기본값. 해당 알람에 대해 어떠한 조치도 취하지 않은 상태
- **알림 중지:** 해당 알람을 사용자가 확인하여 더이상 알림 (notification) 을 받지 않는 상태
- **처리 완료:** 해당 알람을 확인하고 조치를 취한 상태로, 해당 알람에 관련된 tag 기록 가능
- **오탐:** 이상 상태가 아닌데 발생한 알람



구독자 (Subscriber) 는 해당 알람에 관계된 유저를 아이디로 검색하여 추가할 수 있으며, E-mail에 체크하면 해당 유저 정보에 기록된 이메일로 알람을 발송합니다.



19.2.4 Alarm Rule

이 영역에서는 해당 알람의 심각도와 알람 발생 시각, 그리고 이 알람을 발생시킨 룰과 데이터 소스에 관련된 정보를 보여줍니다.

- **Severity:** 현재 발생한 알람의 심각도
- **Occurrence Time:** 알람 발생 시각
- **Alarm Rule:** 알람을 발생시킨 임계치와 임계치 초과 건수/알람 발생 검사 주기. 우측 링크 버튼 클릭 시 해당 알람 룰로 이동
- **Alarm Interval:** 알람 발생 검사 주기. 1분일 경우 1분마다 Abnormal score가 임계치를 넘었는지 검사

- **Data Source:** 데이터 소스 정보
- **Granularity:** 데이터 소스가 적재되는 시간 단위
- **Training Interval:** 모델 학습을 위해 사용한 데이터 기간
- **Scoring Method:** 여러 개의 측정값 (Measure) 을 사용할 경우 Abnormal Score를 계산하는 방식



19.2.5 View by Chart 탭

이 탭 영역에서는 해당 알람 구간에서 모니터링한 데이터의 Abnormal Score 를 그래프로 보여줍니다. 각 조건별 점수 임계치 (Threshold) 에 상응하는 알람 (Critical, Major, Moderate, Low) 별로 발생된 알람의 건수를 확인할 수 있습니다. 차트 산출 방식에 관해서는 [Basic principles](#) 항목을 참조하십시오.



- **Total abnormal score:** 알람 룰에 포함된 모든 측정값 컬럼에 대한 Abnormal Score를 보여줍니다.
- **Chart by measures:** Shows the trend between the predicted value and the actual value of each individual measure column data included in the alarm rule.

19.2.6 View by Table 탭

이 탭 영역에서는 각 알람 발생 건별로 데이터 실제치와 예측치, 그리고 Abnormal Score를 표 형식으로 나열합니다.

The table displays a list of three alarm events. Each event is identified by a number (1, 2, 3) and a color-coded status indicator (red for 1, yellow for 2, yellow for 3). The columns provide detailed information for each event, including occurrence time, total abnormal score, and specific scores for SUM_cpu and SUM_memory.

	Occurrence time	Total Abnormal Score	SUM_cpu (Weight Value: 73%)			SUM_memory (Weight Value: 132%)		
			Actual	Predict	Score	Actual	Predict	Score
1	● 2020-04-06 23:05:00	35	2,163	1,156	64	1,981	2,239	6
2	● 2020-04-06 23:05:53	5	2,124	2,160	2	4,049	4,425	8
3	● 2020-04-06 23:05:57	6	2,159	2,149	1	2,691	3,196	11

+ Load More

ALARM RULE

Metatron Anomaly makes it easy for users to easily create and manage rules that trigger alarms in their time-series data. Metatron Anomaly's alarm rules have the following characteristics:

- Machine learning based on unsupervised learning for all real-time data without error history
- Easy and fast alarm rule creation in 3-step
- Built-in statistical prediction model
- Automatic model learning and optimal model recommendation
- Supports model re-learning when applied model accuracy decreases

The structure of this unit is as follows.

20.1 Creating an Alarm Rule

Anomaly guides users through the following procedures in order to help users easily create the desired alarm rules.

- Select Data Source
- Select metrics to monitor
- Setting the training data
- Choosing a Model
- Setting alarm rule conditions
- Complete the Rule

20.1.1 Select Data Source

To create an alarm rule, you must first set up a data source to monitor.

1. Click the **Create Alarm Rule** button at the top right of the Alarm Rule page.

2. Select the data source you want to monitor.

No.	Datasource	Type	Created	Updated
19	realtime_omron_udp_03_01 Open data	Ingested data	All Workspaces	2019.11.15
18	realtime_server_load_json_01 Open data	Ingested data	1 Workspaces	2020.04.03
17	sales - Sales data [2011~2014] Open data	Ingested data	All Workspaces	2019.09.26
16	sales dataset with forcasting Open data	Ingested data	All Workspaces	2019.10.11
15	sensordata_sample	Ingested data	2 Workspaces	2019.10.14
14	sensor_Test	Ingested data	1 Workspaces	2019.10.11
13	sensor_test	Ingested data	1 Workspaces	2019.10.11
12	south korea multipolygon Open data	Ingested data	All Workspaces	2019.06.17
11	southkorea_apartment_trade_dataset_2019 - From 201903 to ... Open data	Ingested data	1 Workspaces	2019.06.22
10	systemshock-realitimetest-20190926_01 Open data	Ingested data	All Workspaces	2019.10.25

realtime_server_load_json_01
 Metadata name: realtime_server_load_json_01
 Description: Ingested data
 Visibility: Private
 Created: 2020-04-03
 Size: 115.89 MB
 Rows: 14,671,250

Cancel
Next

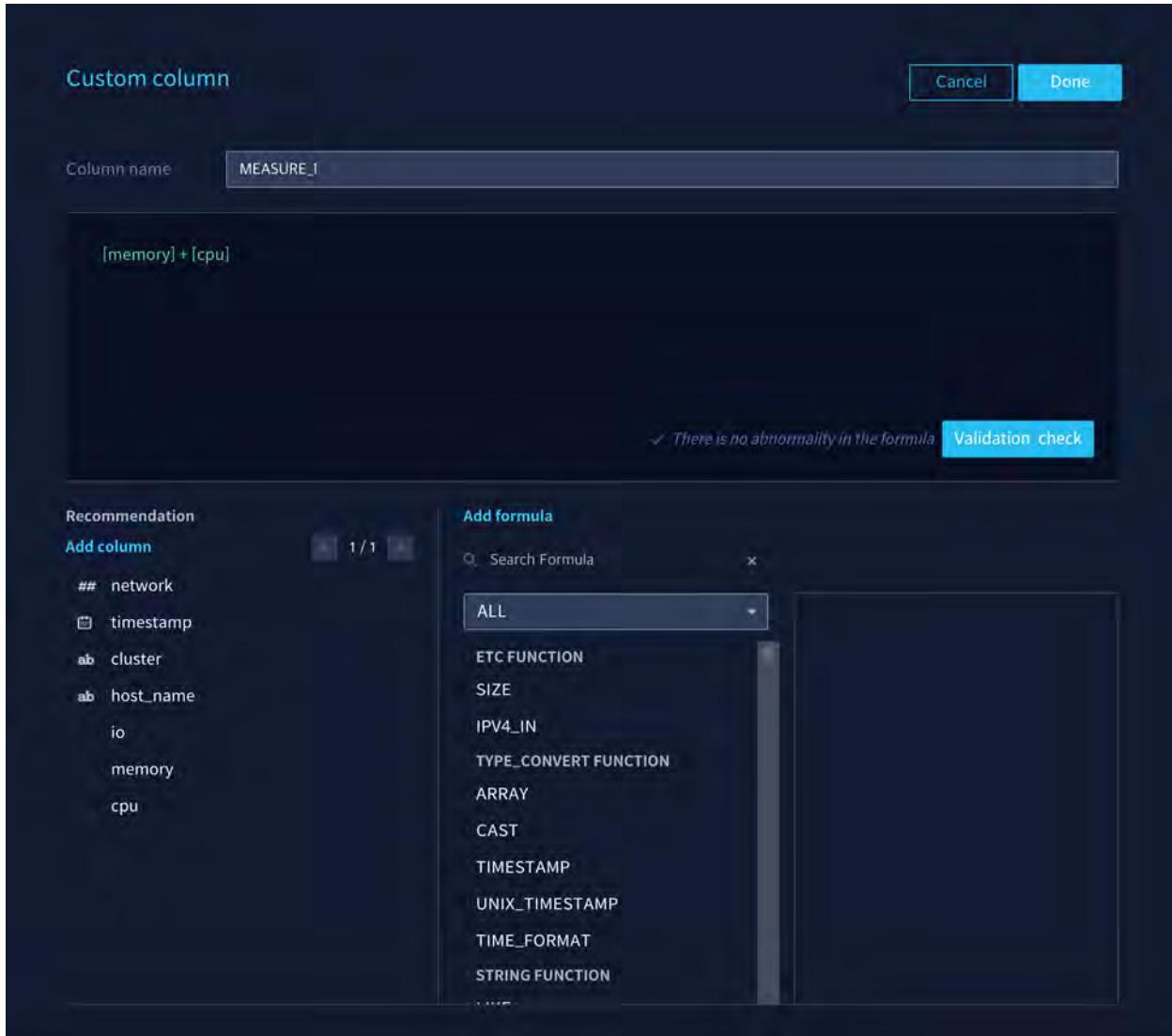
20.1.2 Select metrics to monitor

Selecting a data source will take you to the next screen and the **Data** panel on the left will open. Use this panel to select metrics to monitor as shown below.

1. **Select Measures:** In the **Measure** tab area, select the column you want to monitor. The clicked measure column is automatically moved to the aggregation shelf.



2. **Add User-defined column:** If necessary, you can create a new user column by applying a formula to an existing column. In the upper right corner of the **Measure** area, Click the button to open a dialog box and set up a custom column.



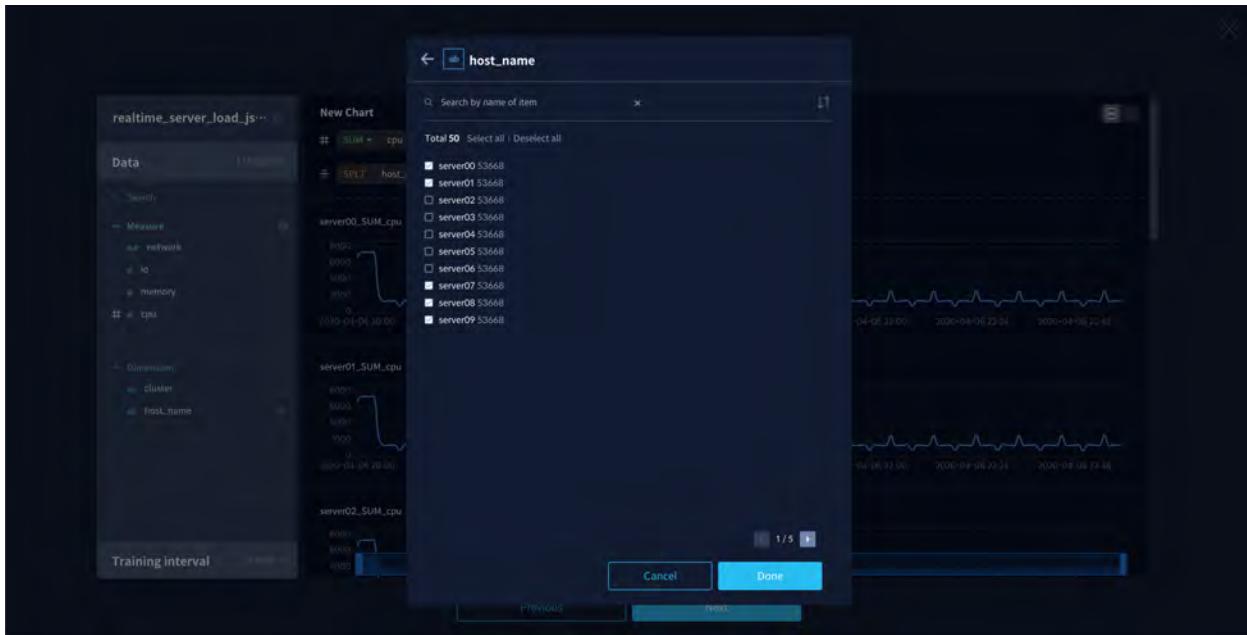
3. **Change measure aggregation method:** Select the desired aggregation method by clicking on each column placed on the Aggregate shelf. The default is SUM.



4. **Split:** You can split aggregated data based on dimension value columns. In the Dimension area, move the mouse cursor over the measure column to be used as the basis for division, and then click the button. The maximum number of splits is 10, and if the dimension value is 10 or more, 10 random values are selected.



5. **Filtering by Dimension value:** You can filter aggregate data based on dimension value columns. In the **Dimension** area, move the mouse cursor over the measure column to set the filter, and click the  button. Then select the specific category you need to monitor as shown below.



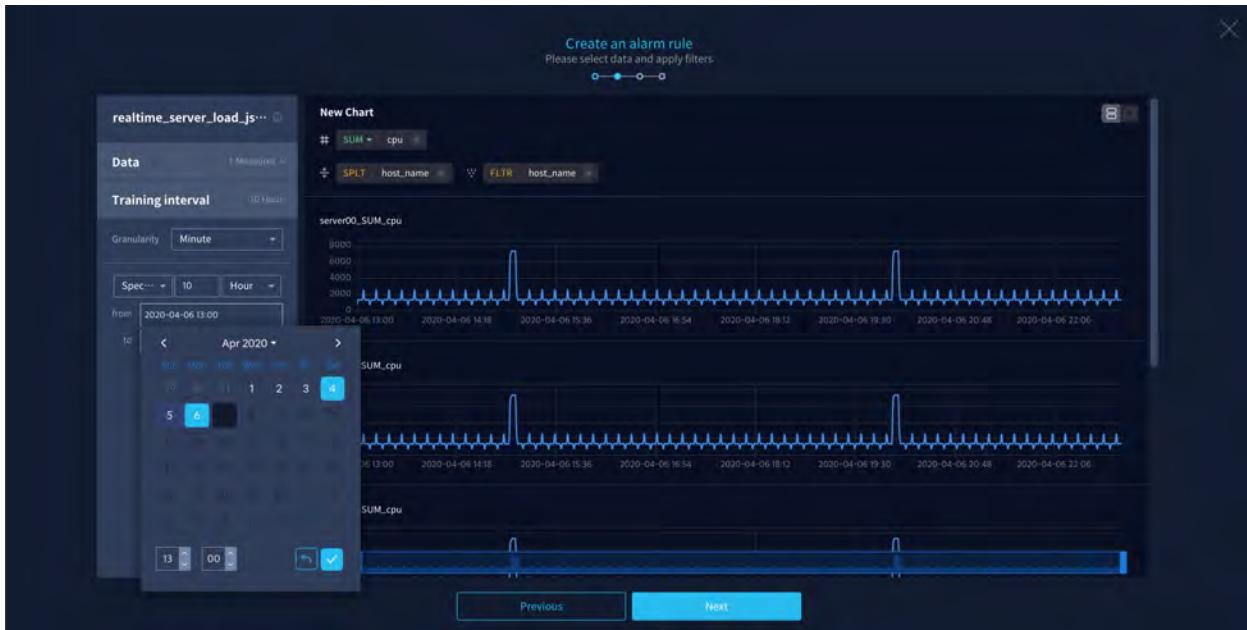
20.1.3 Setting the training data

When you finished selecting metrics to monitor, now you can select the data range to use for training the predictive model in the **Training Interval** panel.

1. **Granularity** can determine the default unit of time for data to be used for model training. While looking at the graph, choose the unit that best shows the pattern of the data.



- Set the range of data to use for training the model. You can enter a range of data to train in units equal to or greater than the default granularity set earlier.

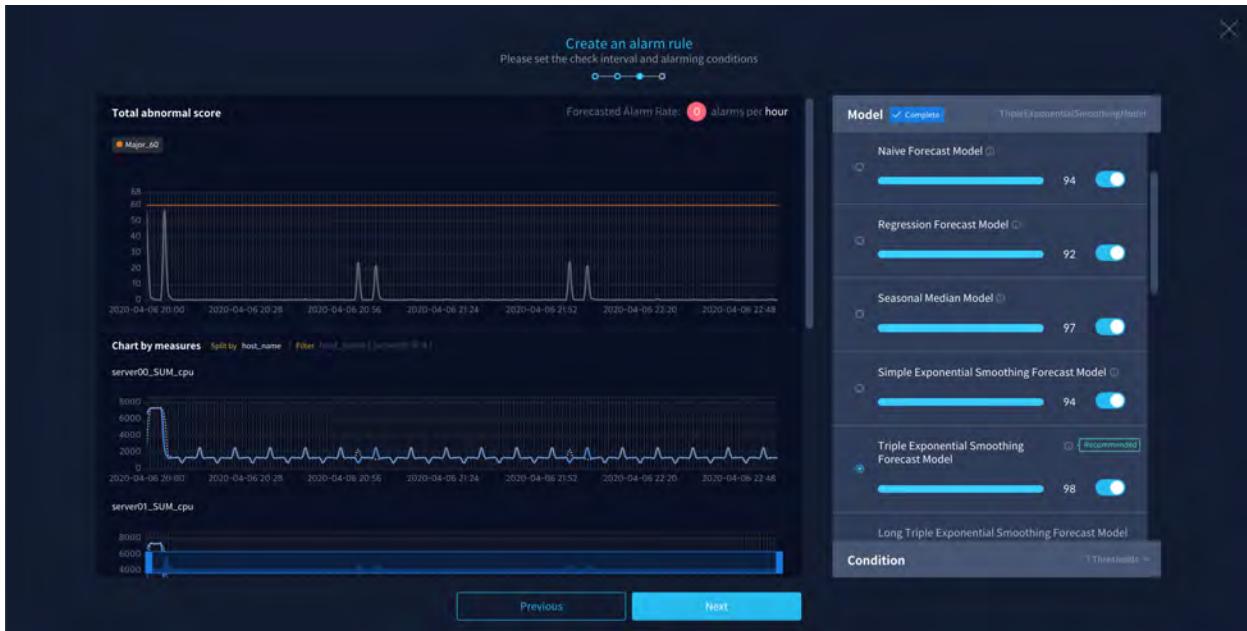


- When all settings are complete, click **Next**.

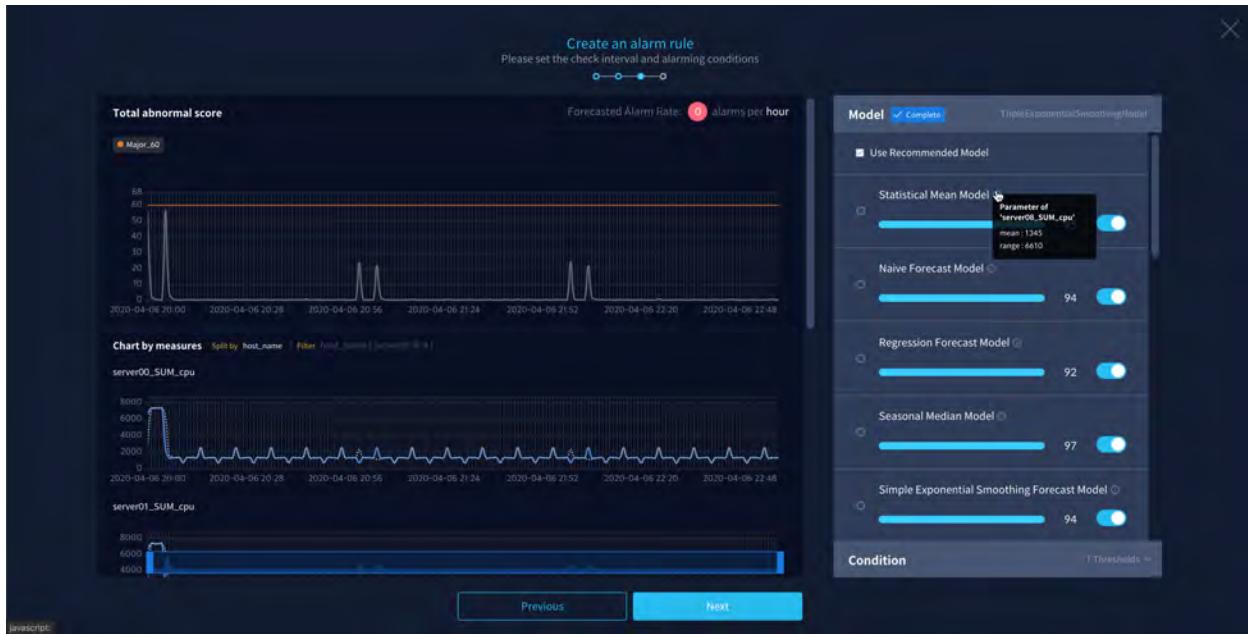
20.1.4 Choosing a Model

Now go to the **Model** panel and choose which prediction model to use. Metatron Anomaly trains each model using a given set of training data and produces the results. Choose a suitable prediction model through one of the two methods below.

- **Use recommended model:** By default, the model with the highest accuracy score (out of 100) displayed on the right is automatically selected with a **Recommend** mark.



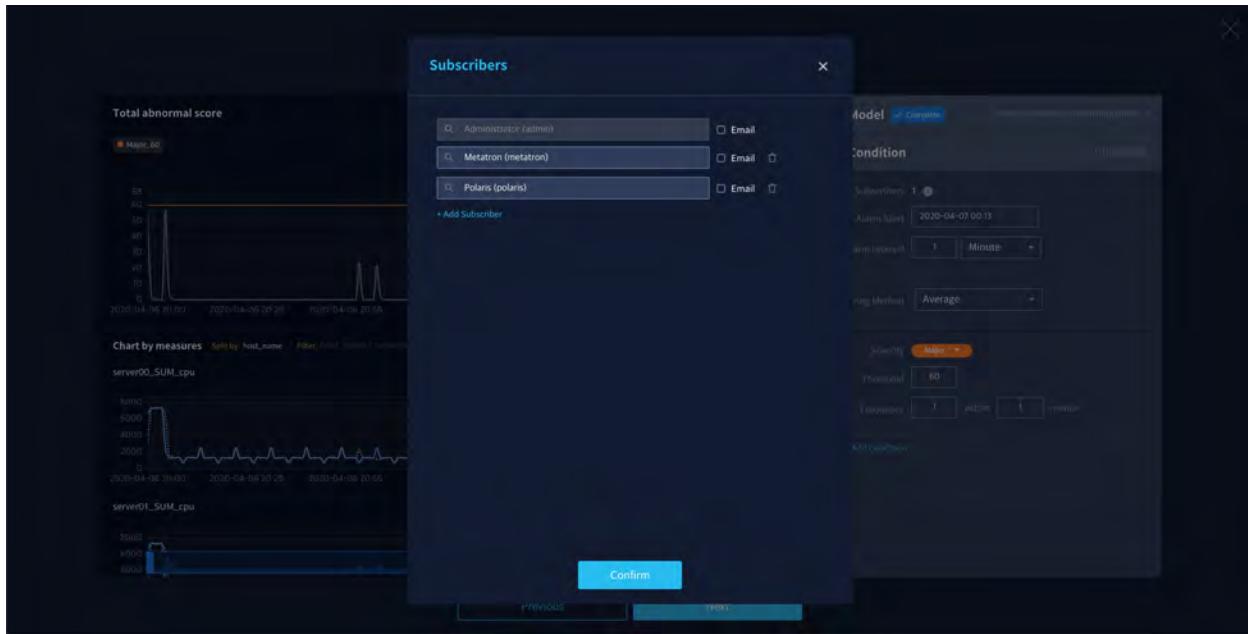
- **Select yourself after comparison:** If you select each model, you can see the predicted value and Abnormal Score in the graph. You can select the model that you think is most suitable. When you mouse-hover the icon to the right of the model name, you can see the detail learning values.



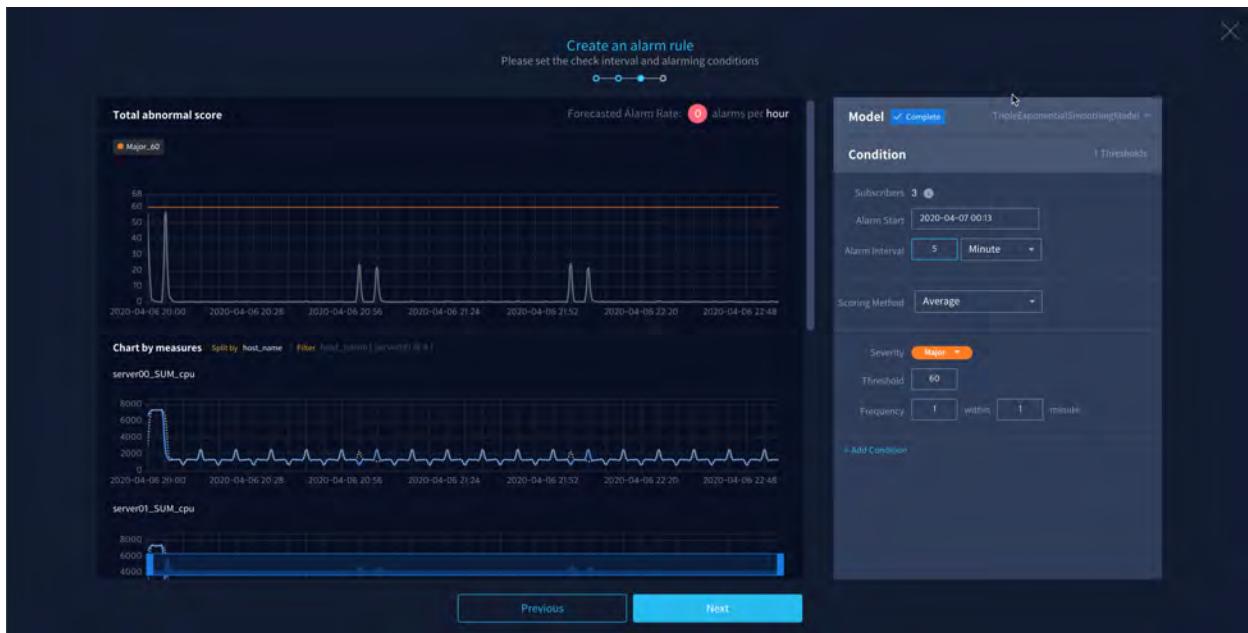
20.1.5 Setting alarm rule conditions

After selecting the predictive model to use, now you need to set the conditions for the alarm to occur in the **Condition** panel.

1. Click to the right of the **Subscribers** to open a dialog box, and set the target and method to be notified when an alarm occurs.

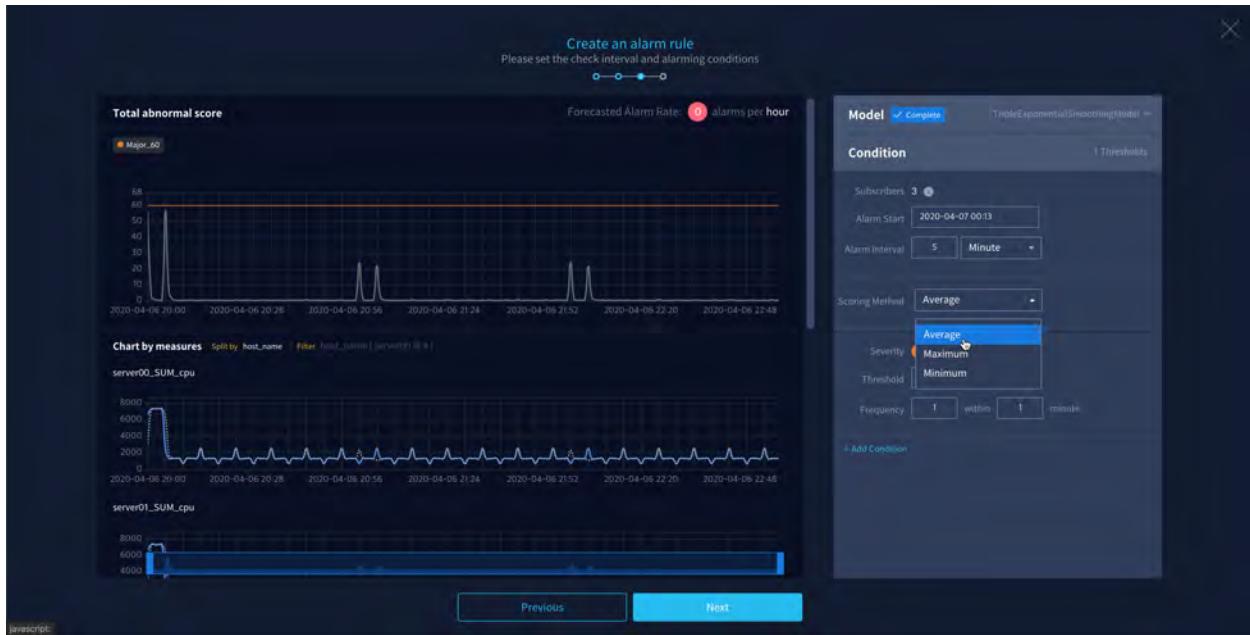


2. Set the time when the alarm is triggered by referring to the description of each item below.



- **Alarm Start:** Set when to start an alarm. The alarm starts after the time corresponding to this setting value.
- **Alarm Interval:** Set the interval to generate an alarm when the condition of the alarm is met.

3. Scoring Methoddetermines how abnormal socres are calculated from multiple measure values split by dimension. The default value is calculated as the average of the abnormal scores of all measures, and can be changed to the maximum or the minimum.



4. Set the alarm trigger conditions according to the abnormal score of monitored data with reference to the description of the following items. By default, one Major level condition is given and you can set more conditions with + Add Condition button.



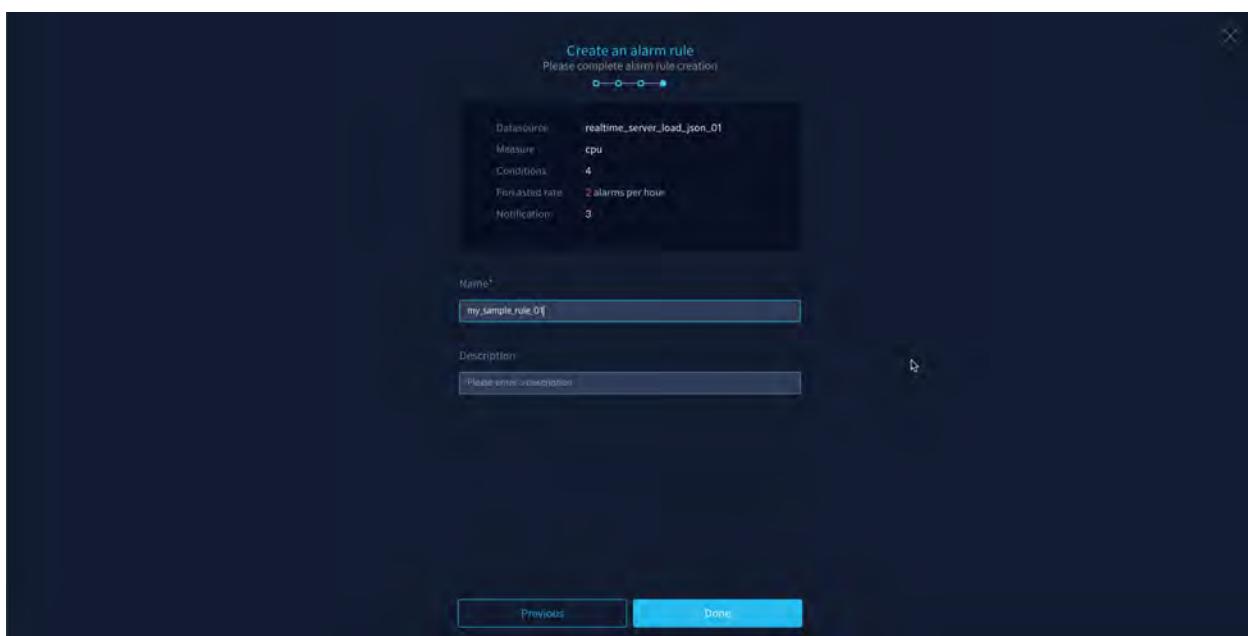
- **Severity:** Set the severity of the alarm for a given condition.
- **Threshold:** If the abnormal score exceeds this setting, the data is considered abnormal.
- **Frequency:** Determines how often an alarm is triggered when the frequency of abnormal scores exceeds the threshold. For example, if it is set to “3 within 5 minute”, an alarm is generated if the abnormal score exceeds the limit value more than 3 times within 5 minutes.

5. When all settings are complete, click **Next**.

20.1.6 Complete the Rule

After all the settings you've done, finish the process of creating the alarm rule as shown below.

1. Enter the name and description of the alarm rule and click the **Done** button.



2. The created alarm rule is exposed at the top of the alarm rule list, and is immediately changed to the running state.

Alarm Rule Name	DataSource	Measure	Alarm Interval	Condition	Alarm	Running State	Updated	Owner
my_sample_rule_01	realtime_server_load_json...	cpu	5 Minute	4	0	Running	2020-04-07 00:28	admin
realtime_server_load_json_001	realtime_server_load_json...	cpu_memory	1 Minute	3	272	Running	2020-04-07 00:28	admin
realtime_server_load_json_01	realtime_server_load_json...	cpu	1 Minute	4	0	Running	2020-04-06 17:37	admin

20.2 Viewing and modifying alarm rule details

Alarm Rule tab menu allows you to view and modify registered alarm rules. In addition, in this menu, you can easily grasp the status of data abnormal scores calculated according to the selected prediction model.

The alarm rule menu consists of the following two pages.

- [Alarm Rule List](#)
- [Alarm Rule Details](#)

20.2.1 Alarm Rule List

When entering the **Alarm Rule** tab, the currently registered alarm rules are listed and displayed.

Alarm Rule Name	DataSource	Measure	Alarm Interval	Condition	Alarm	Running State	Updated	Owner
my_sample_rule_01	realtime_server_load_json...	cpu	5 Minute	4	0	Running	2020-04-07 00:28	admin
realtime_server_load_json_001	realtime_server_load_json...	cpu_memory	1 Minute	3	272	Running	2020-04-07 00:28	admin
realtime_server_load_json_01	realtime_server_load_json...	cpu	1 Minute	4	0	Running	2020-04-06 17:37	admin

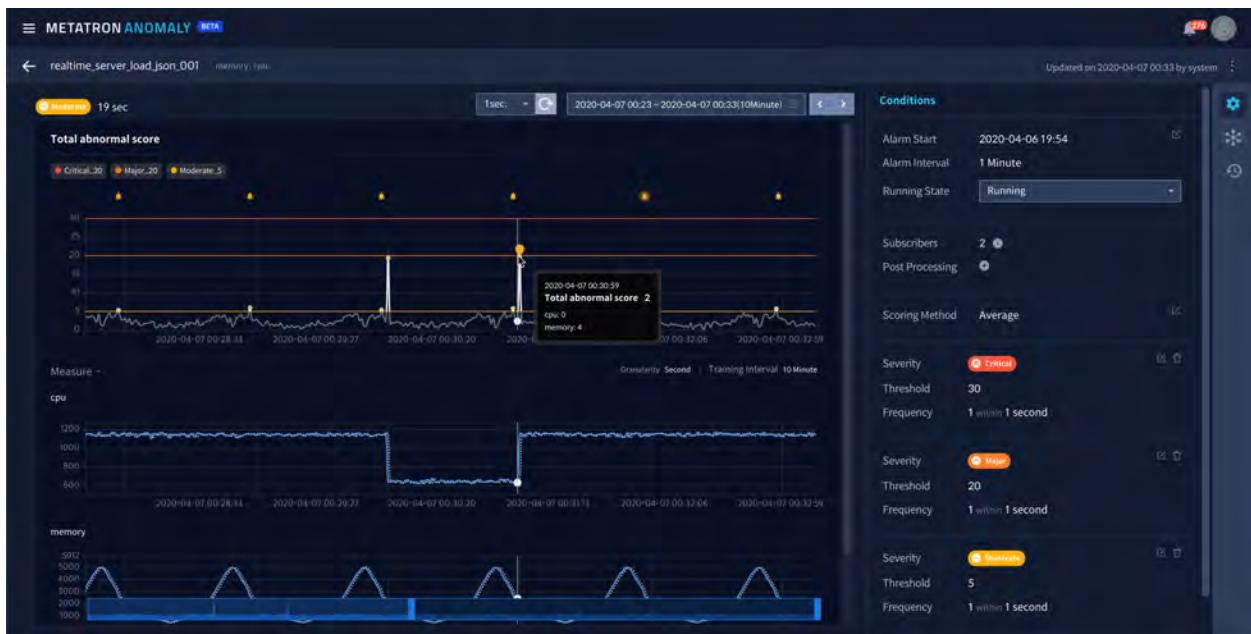
The information displayed in the list is as follows, and you can filter or search the rules to list based on this.

- **Current Status:** Monitoring result status according to the rule
- **Alarm Rule Name:** Rule names

- **DataSource:** Data sources being monitored
- **Measure:** Measure columns being monitored
- **Alarm Interval:** Alarm generation time interval
- **Condition:** The number of alarm occurrence conditions applied to the rule
- **Alarm:** Number of alarms triggered by the rule
- **Running:** Whether the rule is running or not
- **Updated:** Time and user who last updated the rule
- **Owner:** User who created the rule

20.2.2 Alarm Rule Details

If you select one of the alarm rule list items, you can view detailed information about the alarm rule and modify some settings. On the left side of the screen, the monitoring status is visualized and the alarm rule condition setting value is displayed on the right side.

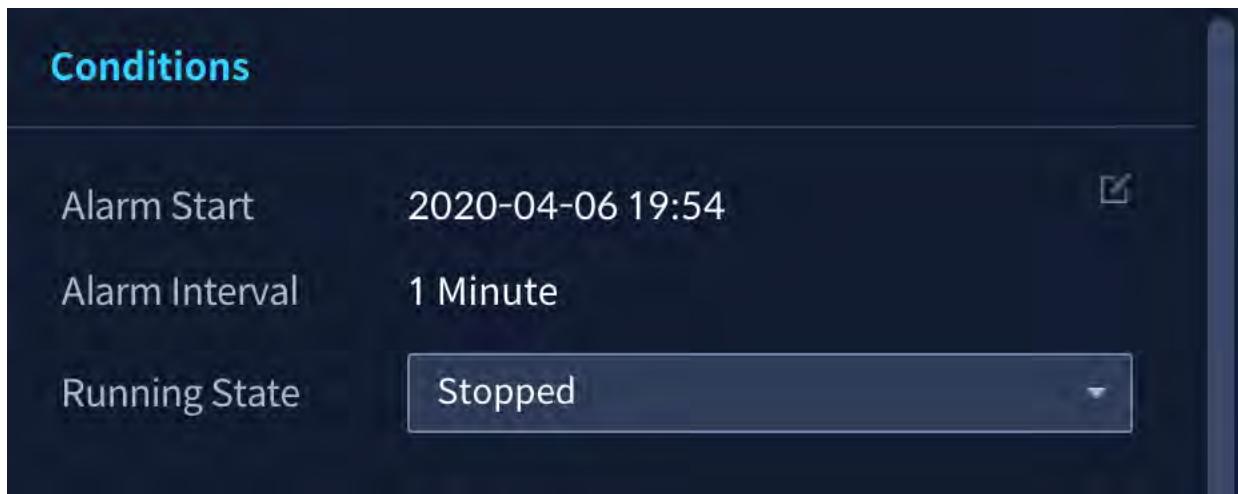


The monitoring period setting value displayed on the screen is displayed at the top of the monitoring status area. You can change the period setting value by clicking the icon.

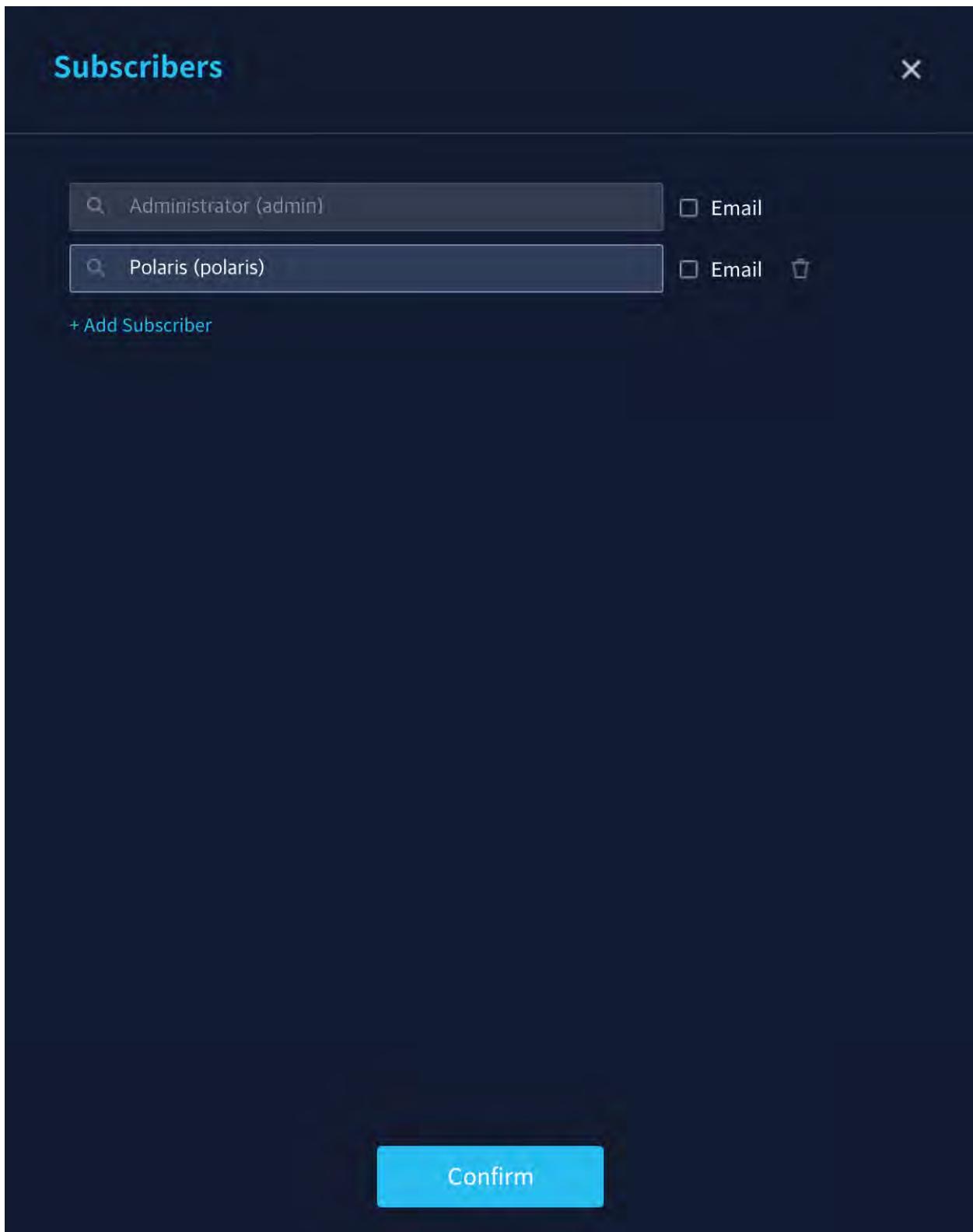


In the condition area on the right, you can adjust the overall settings of the rule.

- **Alarm Start:** Time to start checking for alarms
- **Alarm Interval:** Period to check alarm occurrence conditions
- **Running State:** Whether the alarm rule condition is being checked (running) or not (stopped)

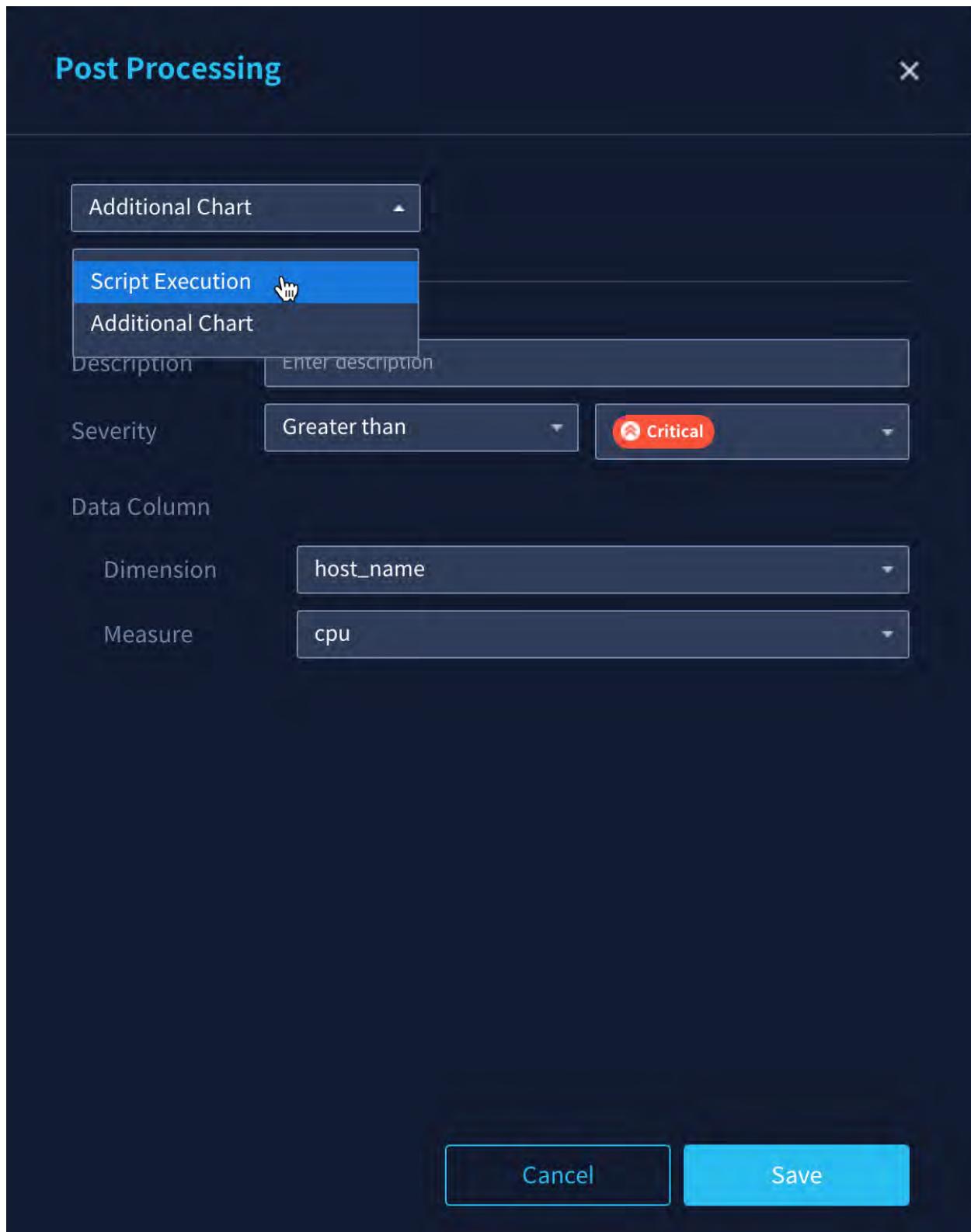


If you click the icon to the right of Subscribers, you can add / change subscribers of the corresponding alarm rule.



Metatron Anomaly provides **Post Processing** that can be configured to take additional action when an alarm occurs due to the rule. Post processing currently provides two functions.

- **Script Execution:** Register and run a shell script
- **Additional Chart:** Expose table chart to alarm details



In addition, the existing alarm occurrence condition can be modified. See: [ref: alarm_rule_settings](#) for more information.



If you click the button at the right end, the **Conditions** panel will switch to the **Alarm History** panel to show the alarm history that has occurred so far (again, press the button to go back to the **Conditions** panel).



20.3 Model Manager

When we apply machine learning models to time series data, we usually have a problem that data patterns change over time and the accuracy of the model gradually decreases. In this case, data scientists ask the data manager to get the new data and then build the model. They have to go through re-learning to get a certain level of accuracy and redeploy to the system. This can sometimes take up to several months.

Metatron Anomaly supports the **Model Manager**, which allows users who are not a data scientists nor a data manager to easily retrain the model.

The model manager consists of the following functions.

- Model accuracy fluctuation
- Model re-training and learning history
- Comparison of models and application of new models

Click  in the right menu of the created alarm rule detail page to enter the model manager.



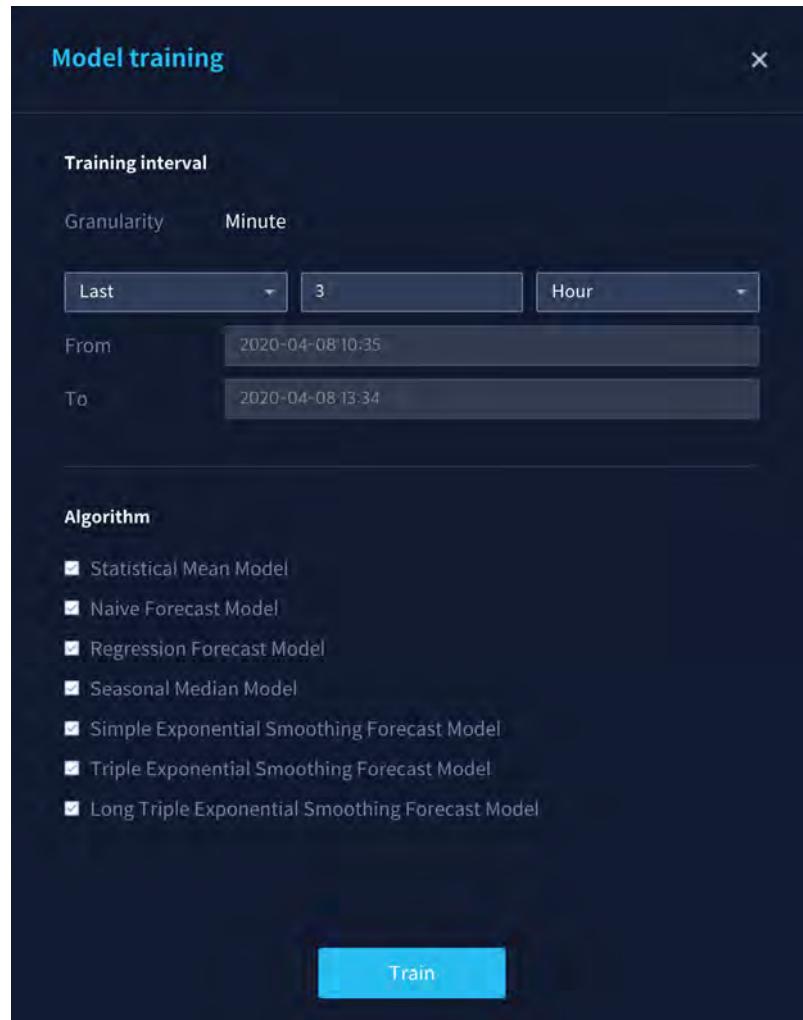
20.3.1 Model accuracy fluctuation

The upper part shows how much the model accuracy has increased or decreased compared to the most recent learning, and the numerical value shows that the accuracy score changes over time when the mouse is over the graph. At the bottom, the information of the currently applied model and the timing of application are indicated. It's possible.



20.3.2 Model re-training and learning history

If the accuracy is lower than the desired value, you can re-learn by clicking the **Train** button at the top right. Select the range of training data and algorithm type to be re-trained and press the **Train** button to start the job.



When re-learning starts, you can see the current status of learning in the menu recorded as the current time in Training History. You can also check the history of the past in the list.



20.3.3 Comparison of models and application of new models

Click the icon to the right of the new model to compare the previously applied model with the newly trained model. The previously applied model is marked with a blue line, and the newly selected model is marked with a pink line to show the values predicted by the two models. You can compare abnormal score values at the same time.



To apply the newly trained model to the rule, click **Apply this training model** from the menu on the right. The applied model is tagged with .



ALGORITHM

Metatron Anomaly utilizes machine learning algorithms to generate alarms for abnormal values in time series data. These outlier detection algorithms are divided into two types depending on whether or not an abnormal sample is used for training.

- **Supervised Anomaly Detection:** A supervised learning algorithm that detects outliers using a training data set with normal or abnormal tags. High accuracy, but takes time and money to acquire abnormal samples.
- **Unsupervised Anomaly Detection:** Unsupervised learning algorithm that can detect outliers even if there are no abnormal tags in the data set, assuming that most of the data are normal samples.

Metatron Anomaly provides learning of the Unsupervised algorithm as a standard to detect anomalies in all time series data without normal or abnormal data labels.

Metatron Anomaly provides the **Algorithm Manager** function to manage these algorithms and add new algorithms. The Algorithm Manager consists of the following three pages.

- [Algorithm List](#)
- [Creating New Algorithm](#)
- [Algorithm Details](#)

21.1 Algorithm List

If you enter the **Algorithm** tab of the Anomaly Detection sub-menu, you can see the algorithms available for model training in the list.

No	구분	Algorithm Name	Description	Notebook	Language	Status	Date	Owner
1	기본	Statistical Mean Model	Statistical Mean Model	-	-	Available	2020-02-28	system
2	기본	Naive Forecast Model	Naive Forecast Model	-	-	Available	2020-02-28	system
3	기본	Regression Forecast Model	Regression Forecast Model	-	-	Available	2020-02-28	system
4	기본	Seasonal Median Model	Seasonal Median Model	-	-	Available	2020-02-28	system
5	기본	Simple Exponential Smoothing Forecast Model	Simple Exponential Smoothing Forecast Model	-	-	Available	2020-02-28	system
6	기본	Triple Exponential Smoothing Forecast Model	Triple Exponential Smoothing Forecast Model	-	-	Available	2020-02-28	system
7	기본	Long Triple Exponential Smoothing Forecast Model	Long Triple Exponential Smoothing Forecast Model	-	-	Available	2020-02-28	system

By default, Metatron Anomaly has the following seven statistical algorithms built into the system.

- Seasonal Median Model
- Statistical Mean Model
- Regression Forecast Model
- Naive Forecast Model
- Simple Exponential Smoothing Forecast Model
- Triple Exponential Smoothing Forecast Model
- Long Triple Exponential Smoothing Forecast Model

21.2 Creating New Algorithm

You can add a new algorithm by clicking the **+ Algorithm** button at the top right of the algorithm page.

The screenshot shows the METATRON ANOMALY BETA interface. At the top, there are tabs for Anomaly Detection, Statistics, Alarm, Alarm Rule, and Algorithm. The Algorithm tab is currently selected. Below the tabs, there is a search bar with filters for Notebook (All), Language (All), Available (Available), and N/A (0). A table lists two algorithms: 'Statistical Mean Model' and 'Naive Forecast Model'. The 'Algorithm' button in the top right corner is highlighted with a red box.

No.	구분	Algorithm Name	Description	Notebook	Language	Status	Date	Owner
1	기본	Statistical Mean Model	Statistical Mean Model	-	-	Available	2020-02-28	system
2	기본	Naive Forecast Model	Naive Forecast Model	-	-	Available	2020-02-28	system

Enter the name and description of the algorithm you want to create. The default working environment available is a Jupyter Notebook with Python language.

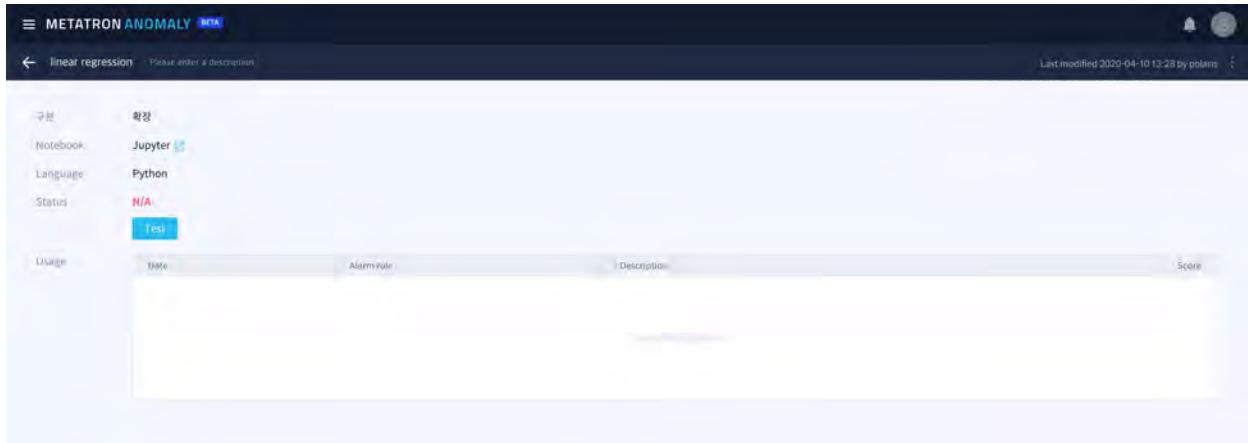
The screenshot shows the 'Add an algorithm' dialog box. It has fields for Name (with placeholder 'Please enter a name'), Description (with placeholder 'Please enter a description'), Notebook (set to 'Jupyter'), and Language (set to 'Python'). At the bottom, there are 'Cancel' and 'Create' buttons.

21.3 Algorithm Details

If you create a new algorithm, you will be moved to the detail page. In the **category**, if it is a user-generated algorithm, it will be displayed as an extension, and if it is a system-implemented algorithm, it will be displayed as a default.

Clicking the next to the notebook takes you to the Jupyter Notebook environment where you can implement new algorithms. A linear regression algorithm is implemented as a basic template, and a new algorithm can be implemented by the user with appropriate modifications.

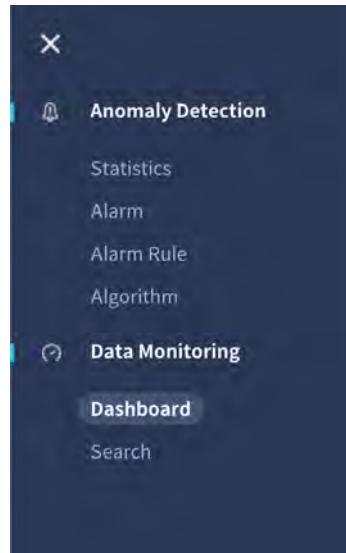
You should test the implemented algorithm to see if it is suitable for your system. If you press the Test button at the bottom, the implemented algorithm will be tested internally for your system. **Status** will show the result. The test results are recorded as “N/A” if never tested, **Fail** if failed, **Available** if successful.



DASHBOARD

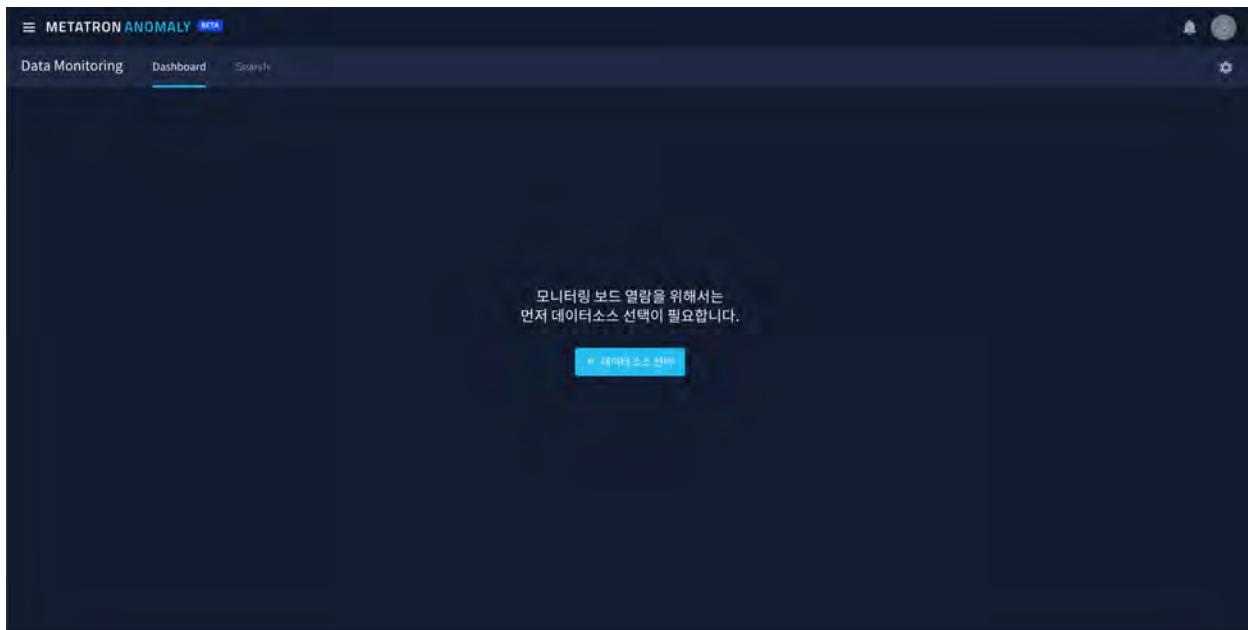
Metatron Anomaly provides monitoring of the data source itself, in addition to anomaly detection using a machine learning model. It can be used to find the cause after an alarm has occurred, or it can be used to check what measure and dimension to create an alarm rule for.

Among them, the dashboard is a sub-menu of Data Monitoring, and it is a function created to quickly grasp the status of the data source with a few of the established charts.

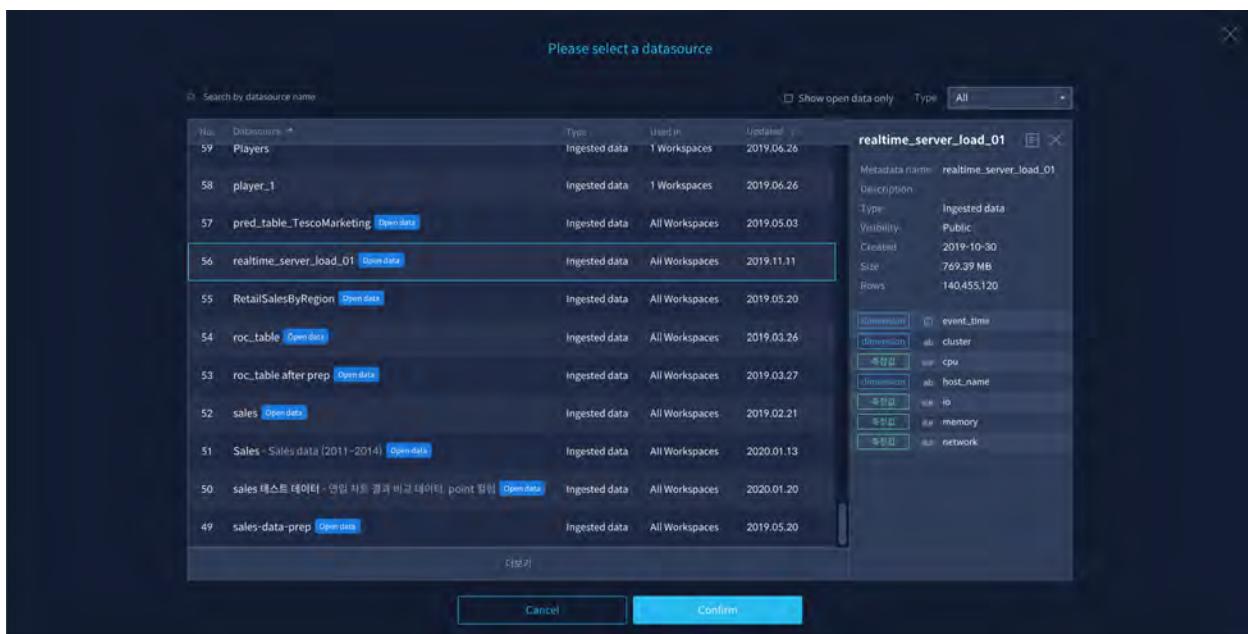


22.1 Select Data Source

First of all, you need to select the data source you want to monitor. You can see the button like below when entering the dashboard menu.



Search and select data source after clicking the button.



22.2 Real-time Dashboard

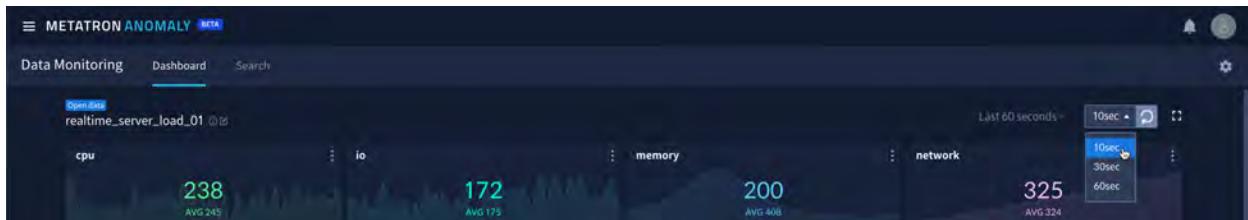
As soon as you select a data source, a dashboard is created with charts for four key measurements. This dashboard is retained even if the user navigates back to another screen and then returns.



1. Also you can see the information of the data source at the top of the dashboard. Click if you want to change the data source to monitor.
2. If no chart is drawn after selecting the data source, check the period to be monitored in the menu on the right-top. This dashboard assumes that you are monitoring data sources that are constantly updated.



3. Click at the top to auto-update the dashboard at a fixed time. By default, it is updated every 10 seconds, and the update cycle can be changed to 3 seconds, 20 seconds, or 30 seconds.



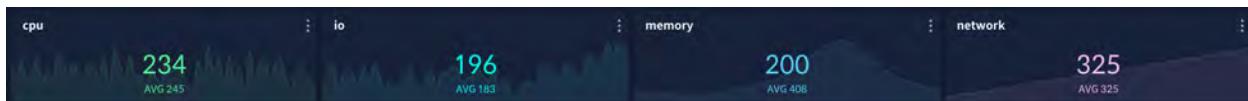
4. Click to switch to full screen mode. Press again in fullscreen mode to return to the normal screen.



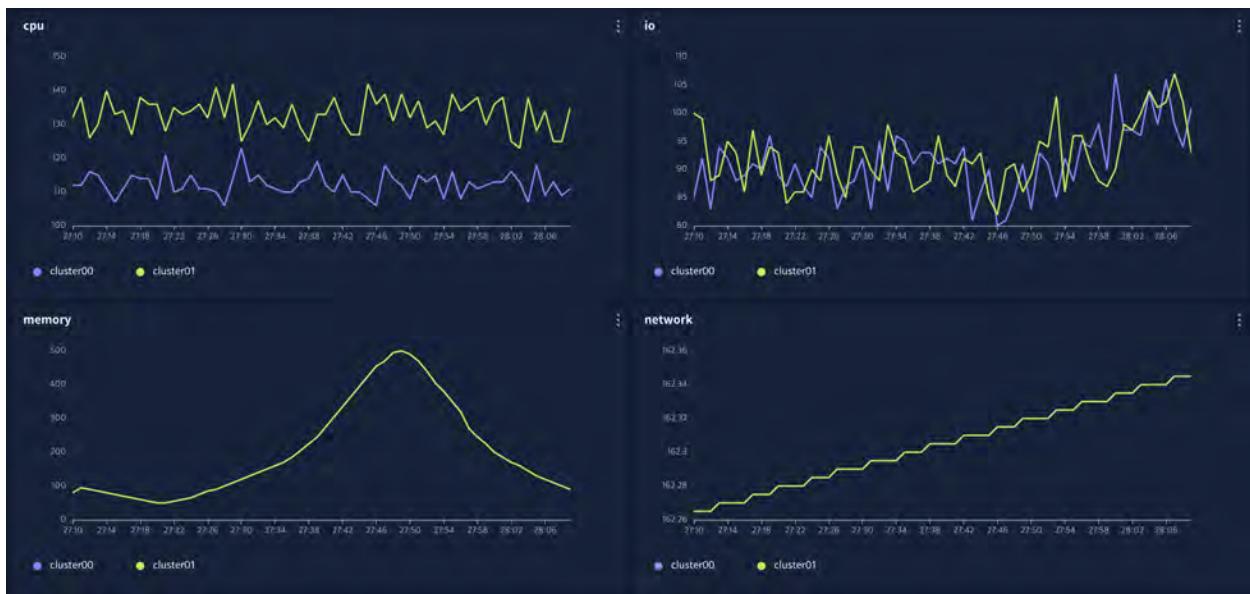
22.3 Chart

The dashboard automatically draws 11 charts for 4 random measures from selected data source.

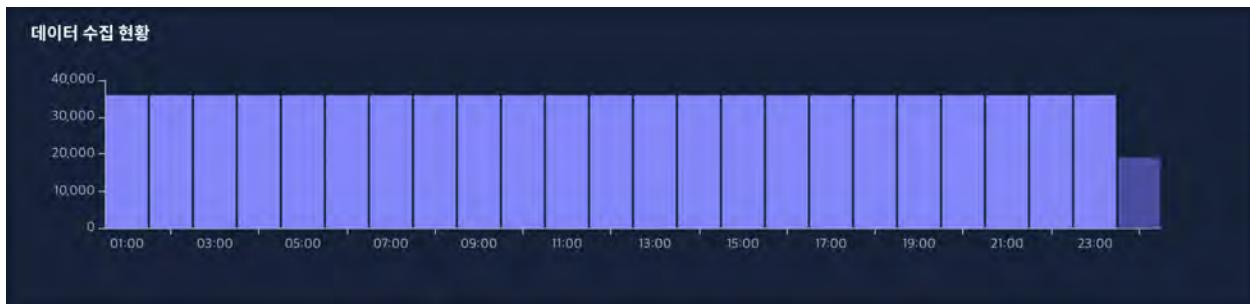
- 4 KPI charts for measures : KPI charts for current and average values for 4 individual measures



- 4 Line charts by 1 demension : Line charts for 4 individual measures for 1 randomly selected dimension value.



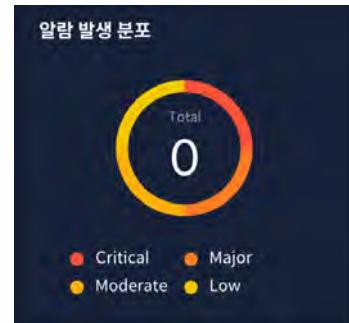
- **Data collection status :** A bar chart that records how many data records were collected over a 24-hour period.



- **Data collection delay time :** A chart showing the collection delay time as the difference between the time when the most recent data was collected and the current time.

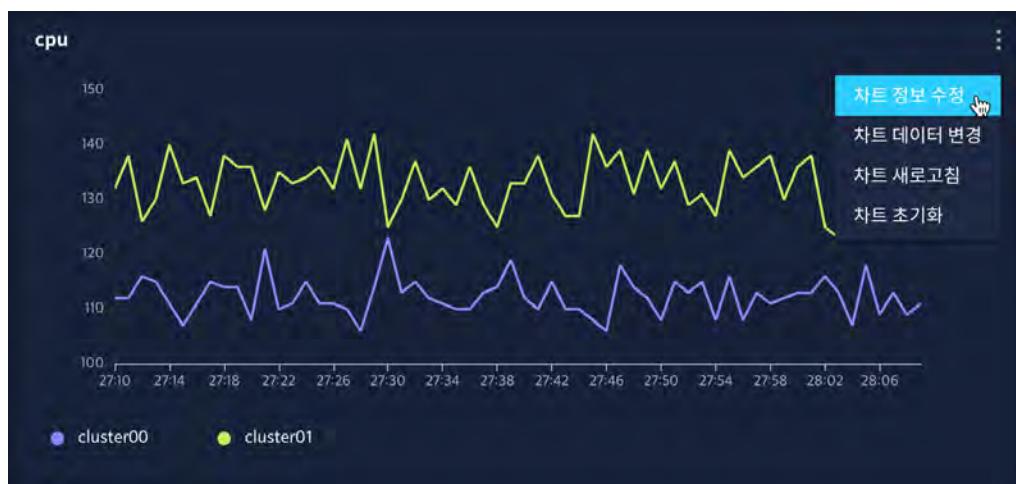


- **Alarm occurrence distribution** : A pie chart showing the alarms generated by the data source by severity.

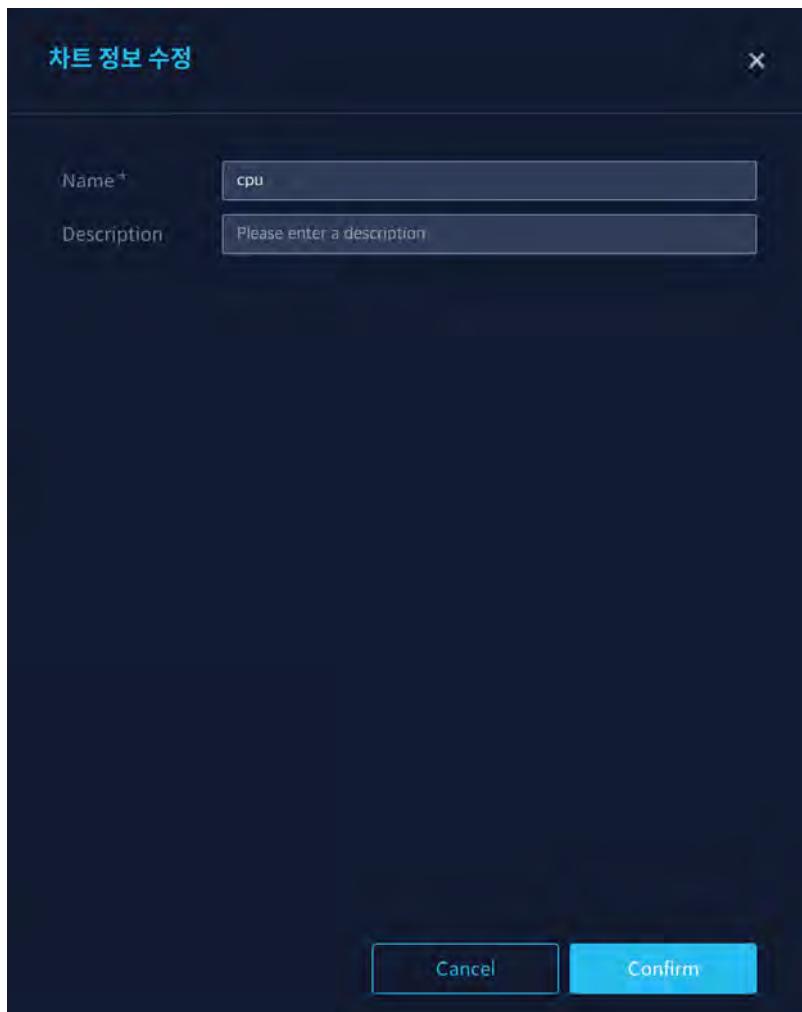


22.3.1 Chart Change

Each chart can be changed by clicking the button on the right.



1. **Modify chart information** : You can rename the chart or add a description.



2. **Change chart data** : You can change the measure or dimension value to be displayed on the chart.

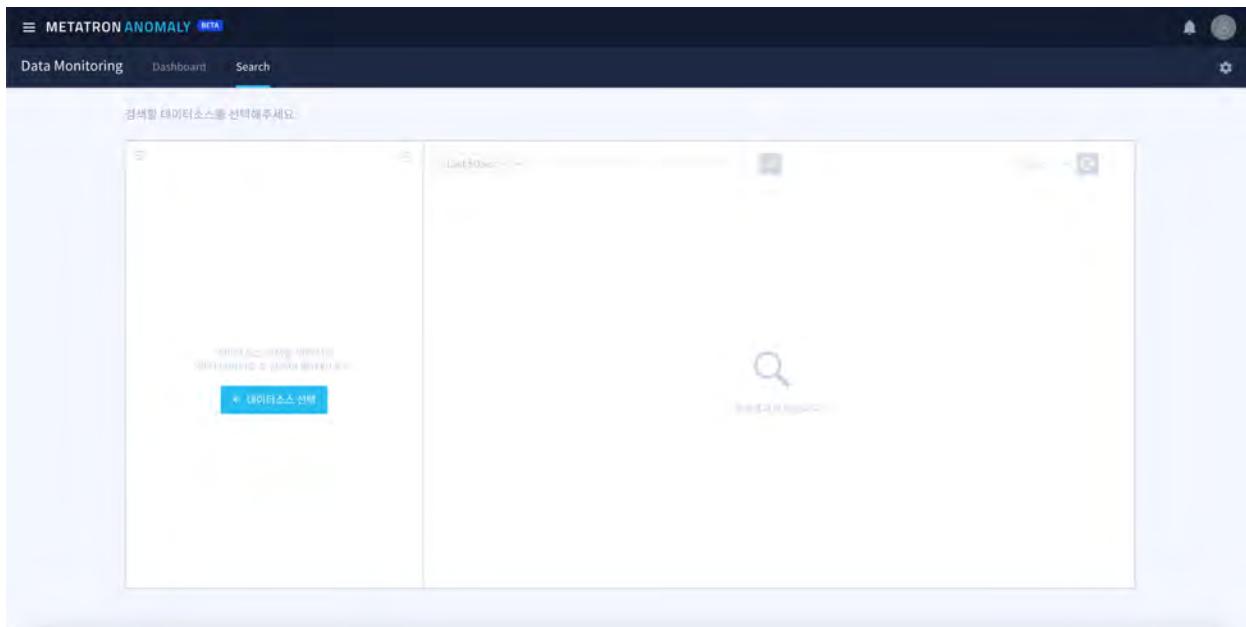


3. **Refresh chart** : Update to the latest data for the individual chart.
4. **Initialize chart** : Initializes the chart drawn with the first set measure and dimension values.

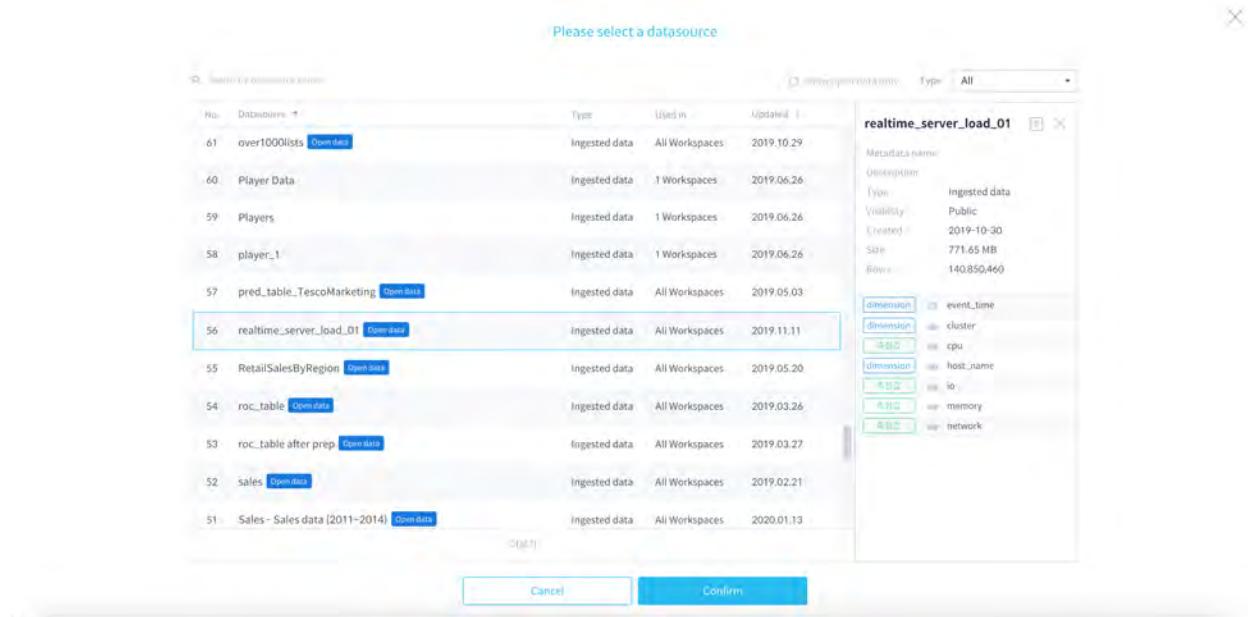
In general, there is a system for detecting outliers and a system for querying data, so you need to access another system to search for data to find the cause immediately after anomaly detection. Metatron Anomaly provides the ability to query various data sources selected by the user within the same system immediately after receiving an anomaly detection alarm.

23.1 Selecting Data Source

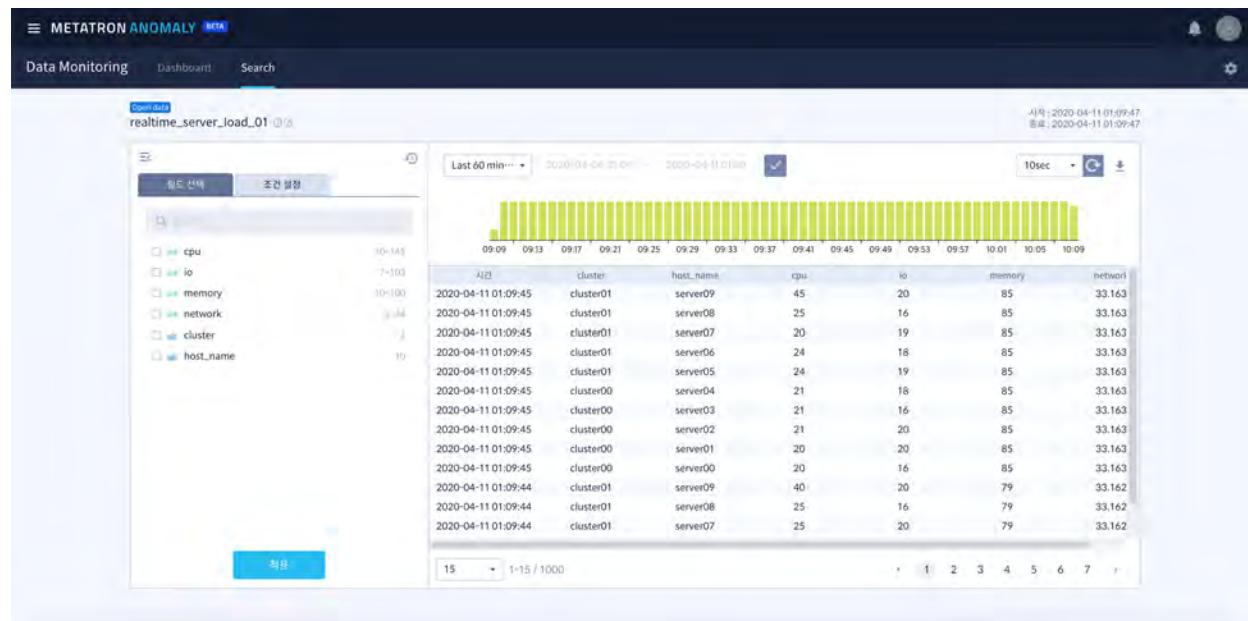
When you first access the search menu, you must first select a data source. The selected data source is retained even if you navigate to a different screen until you change to a different data source.



Click the data source select button at the bottom.



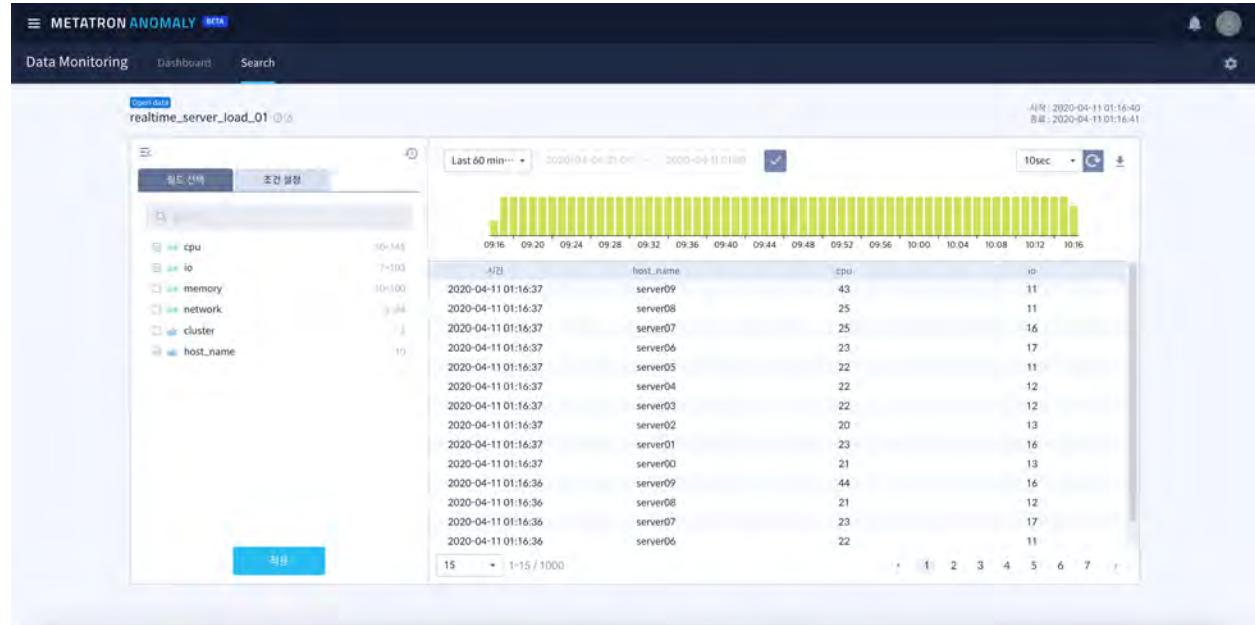
When a data source is selected, the result values for the entire field are queried by default. The query period is set differently depending on the data source collection time unit.



23.2 Choosing Fields & Conditions

23.2.1 Select fields

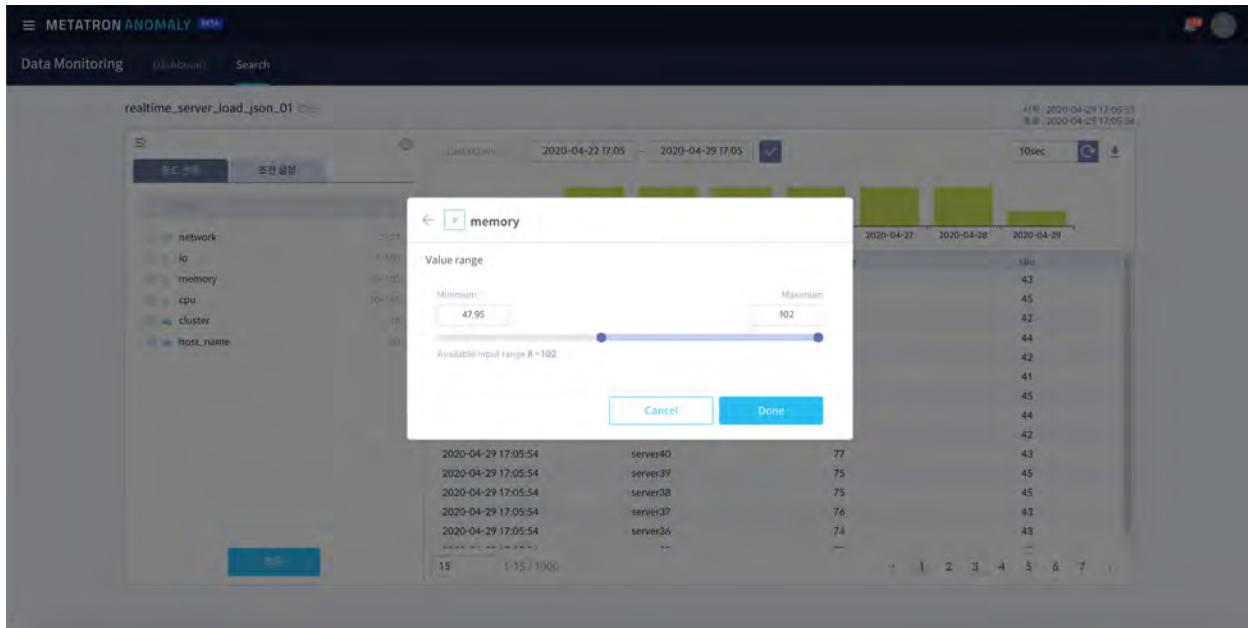
To the right of each field value is the range of values for a measure and the number of values for a dimension. If you select the field you want to search and click Apply, you can search only for the corresponding field value.



The number to the right of the field name means the range of each column value or the number of eigenvalues.

- If the column is **Measure**, the number on the right means the range of the minimum value to the maximum value of the column.
- If the column is **Dimension**, the number on the right means the number of eigenvalues of the column.

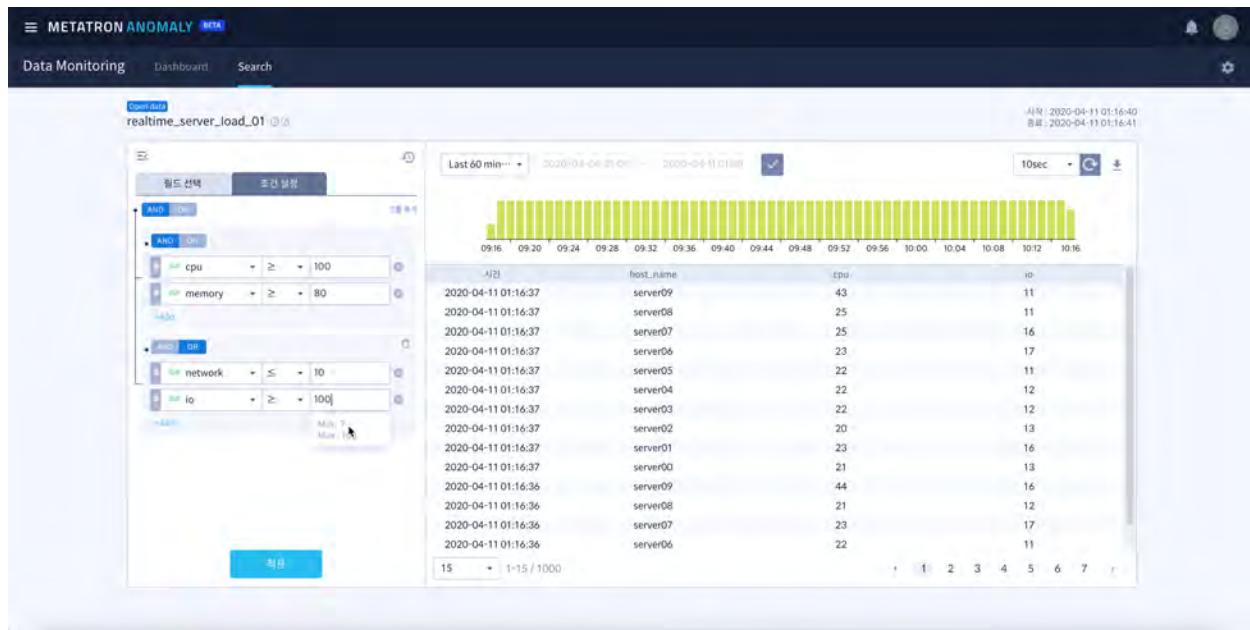
If you click the number to the right of the field name, you can quickly filter for the column, and the added filter can be checked in the **condition setting tab**.



23.2.2 Condition setting

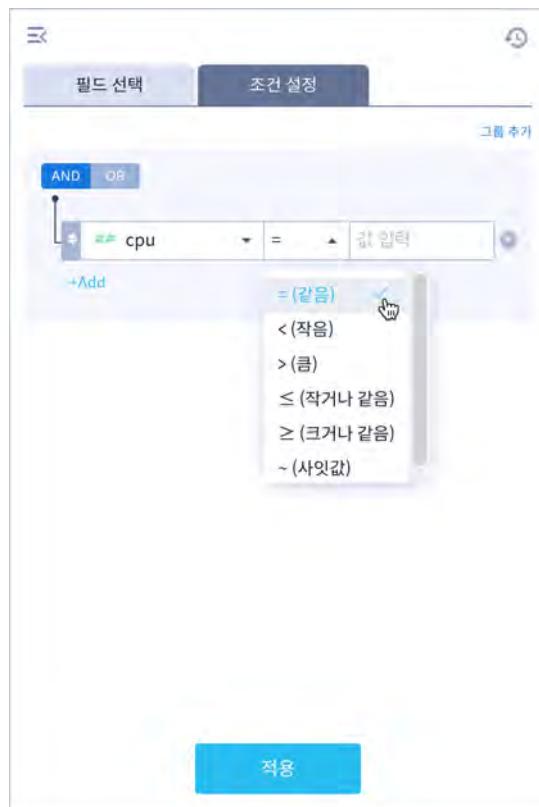
You can set the search condition for each field in detail by using condition setting tab. We provide a easy UI that enables system operators and business users who cannot write data inquiry query to easily set complex conditions to search data.

1. Conditional expressions for individual field values can set up an and / or relationship with each other, and by adding a group at the top, you can also set up an and / or relationship between groups.



2. There are six comparison operators provided for measure field conditional expressions:

- = (equals)
- < (less than)
- ≤ (less than or equal to)
- ≥ (greater than or equal to)
- ~ (between)

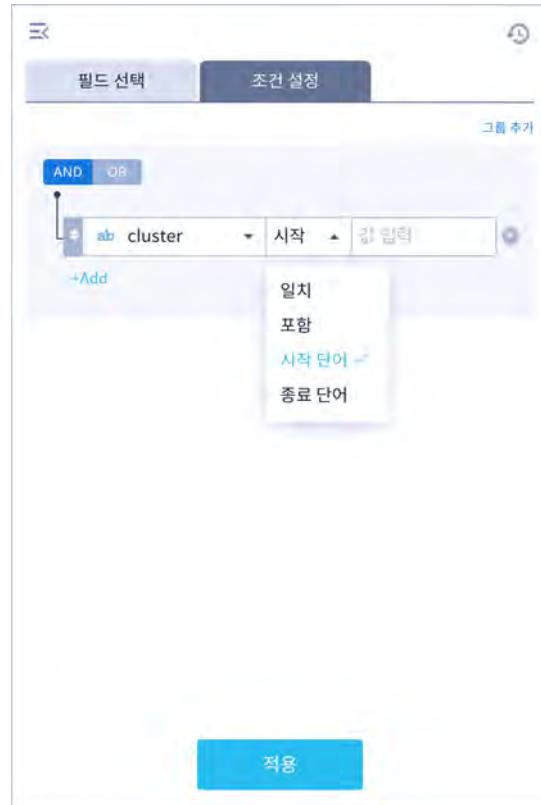


When the cursor is entered in the condition input field for the measure value, the minimum and maximum values are displayed as a tooltip. You can enter the conditional expression by referring to the corresponding value.



3. There are four operators provided for dimension value field conditional expressions:

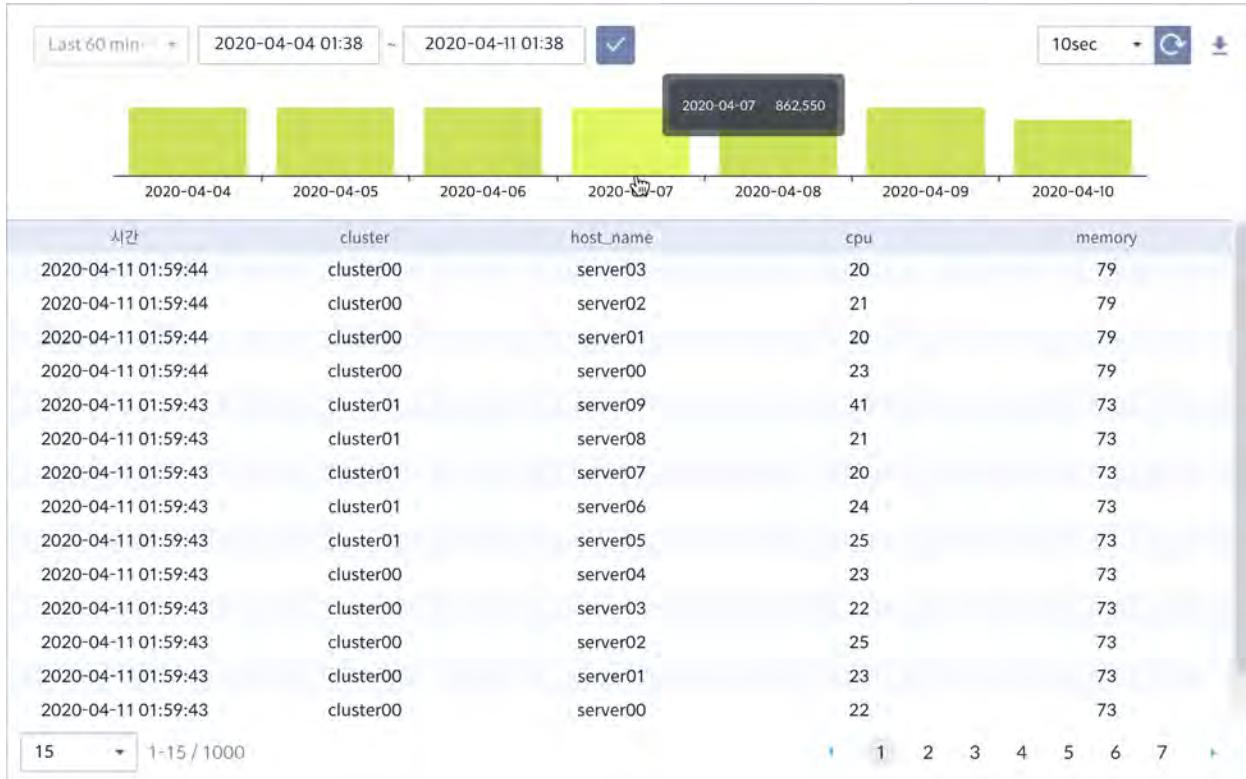
- Start from
- Inclusion
- Start word
- End word



4. All search histories can be searched for the data source by clicking ⓘ . Search time, search condition, etc. can be inquired for each search history, and clicking Apply on the right will inquire data with the same conditions.



23.3 Search Result



- Set inquiry period:** You can set the data range to be searched at the top of the search result window. You can select the time period relative to the current time from the drop-down menu or specify a time range.
- Auto update of search results:** If you click the button on the right, the search results are updated at 10 second intervals to support inquiries for new incoming data. The update cycle can be changed to 3 seconds, 30 seconds or 60 seconds, and clicking again can stop the update.
- Download Excel file:** If you click , Excel type file(.xls) is downloaded on your local PC.
- Histogram :** This is a bar chart that measures the number of data per unit of time the data is stored.
- Change the number of a list item :** The maximum number of data records to be viewed at one time is 1000, and the number of records to be displayed on one page can be changed to 15, 30, or 50 in the drop-down menu at the bottom.