

# Introduction of Metatron Discovery

---

SKT Metatron  
Development Team

---

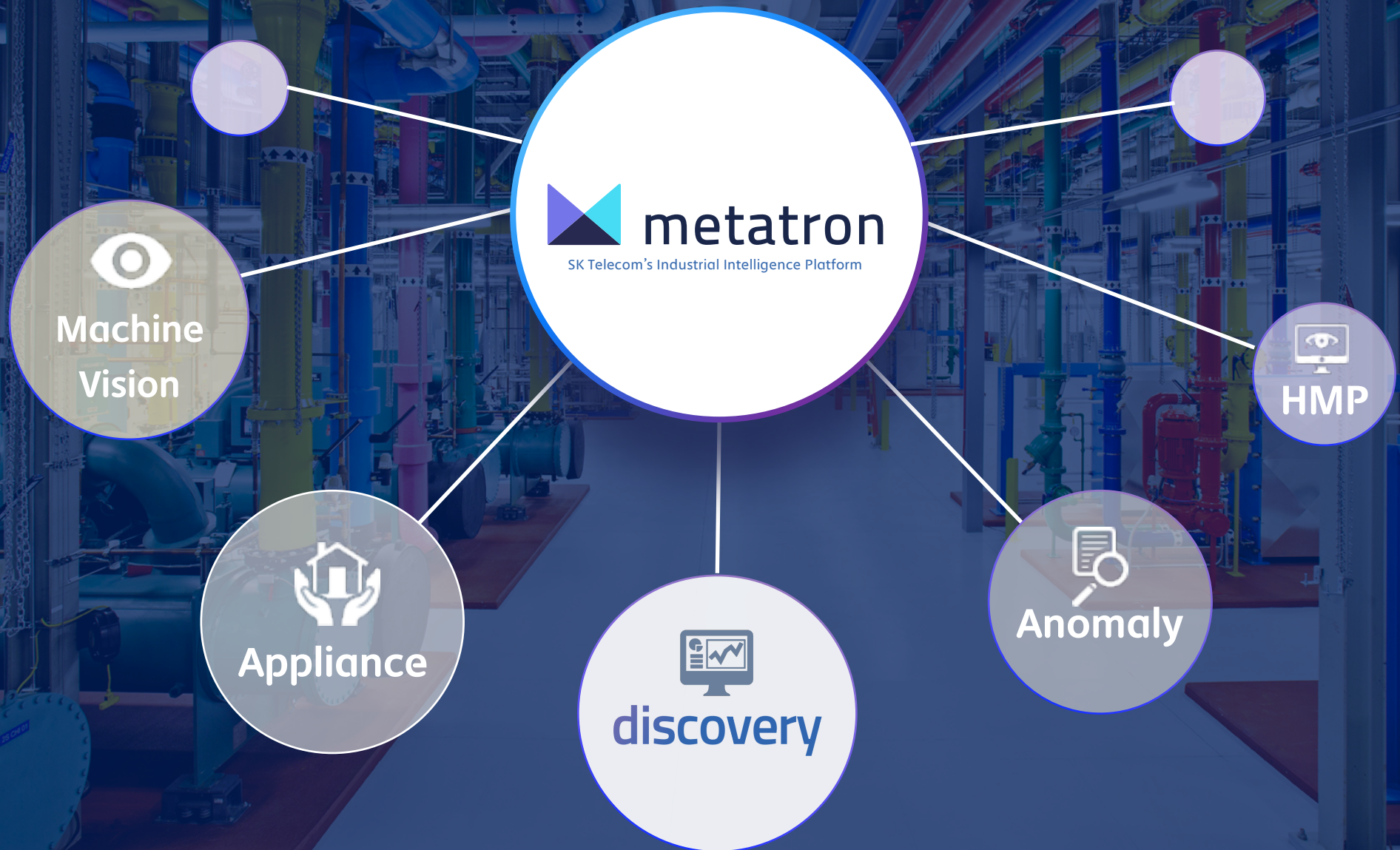
## Contents

1. Overview
2. Use Cases
3. Technology
4. Functions



# *Intelligence Platform for Smart Factories*

Big data- and AI-based technology development and commercialization





Metatron Discovery

# Overview



# metatron discovery



Big data- and AI-based data discovery and analytics



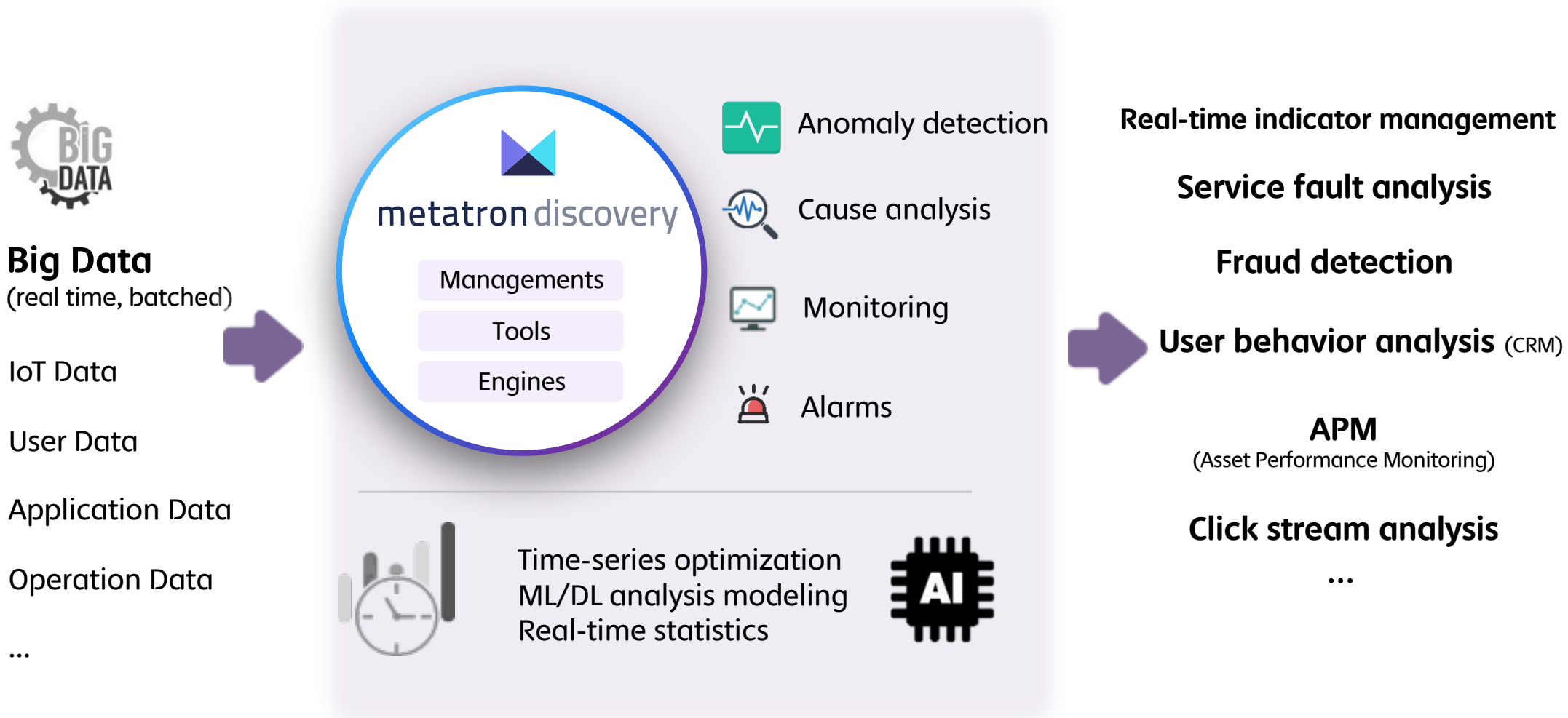
**Performance Efficiency**  
**Business Insights**  
**Customer Understanding**

Machine learning, big data, and visualization technologies are combined so that even non-experts can obtain highly valuable insights from the data easily.



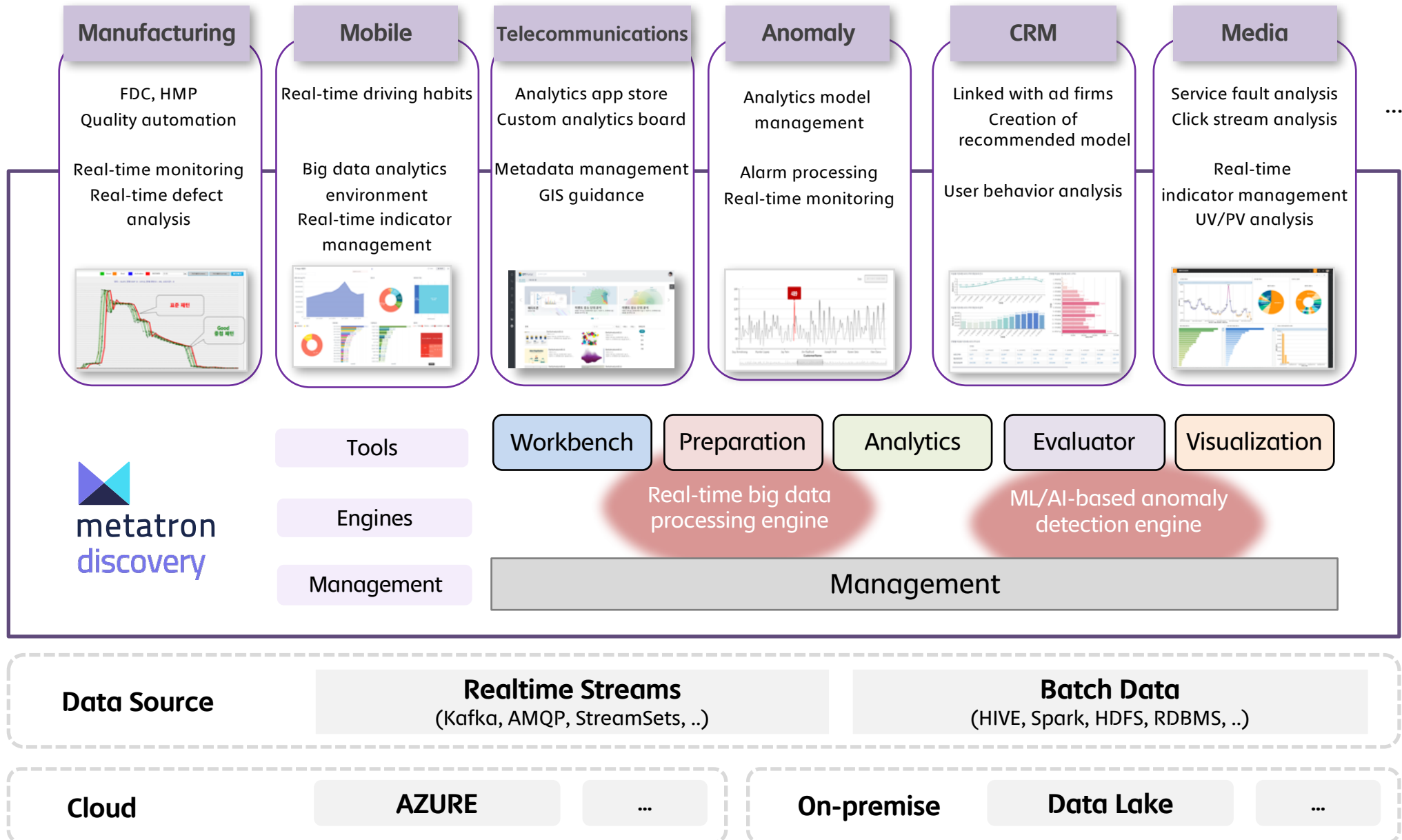
## Metatron Discovery – Overview

“Big data- and AI-based data discovery and analytics”



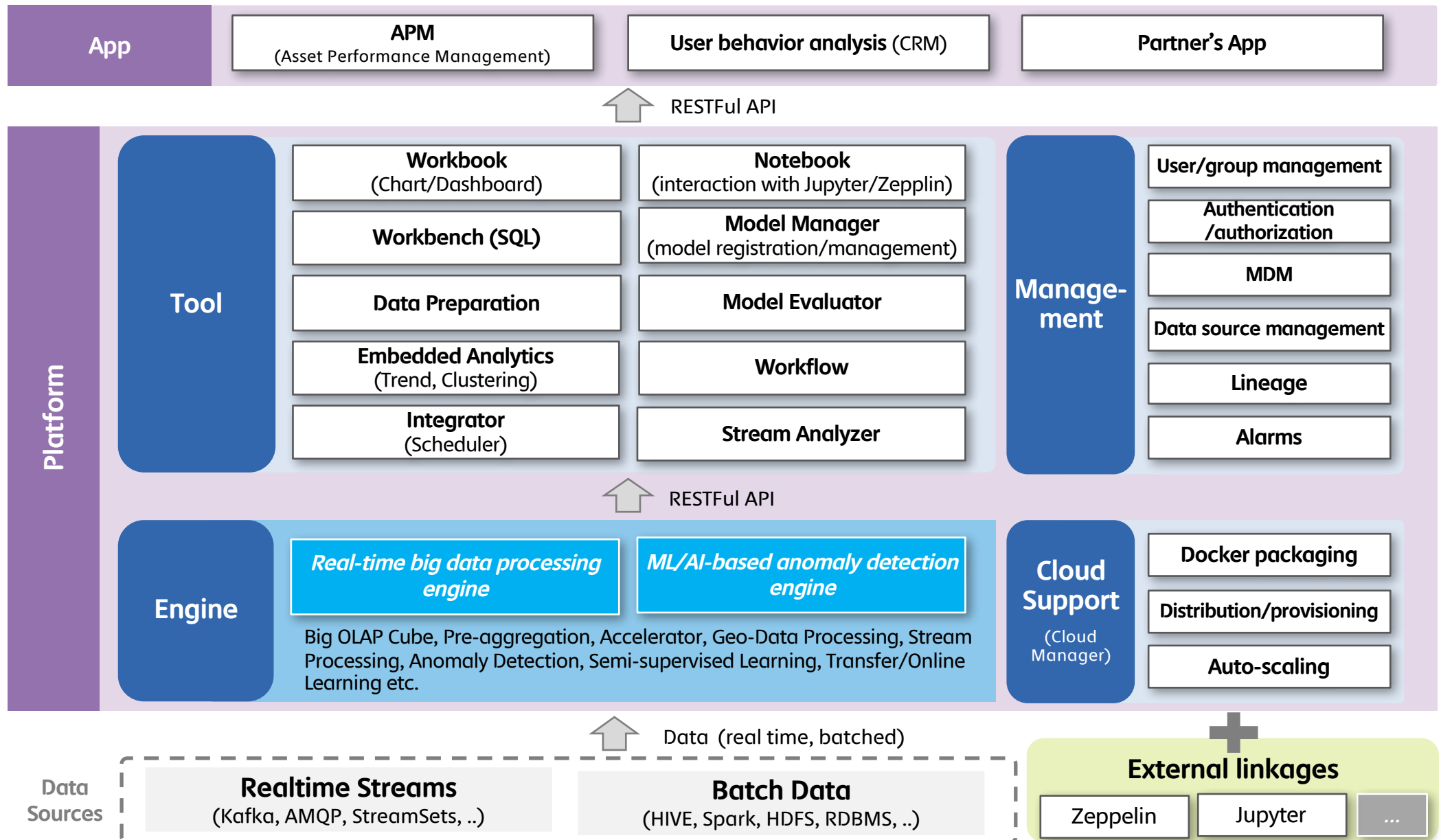


# Metatron Discovery – Use Cases



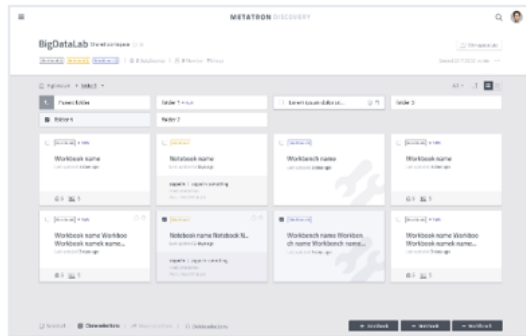


# Metatron Discovery – Architecture

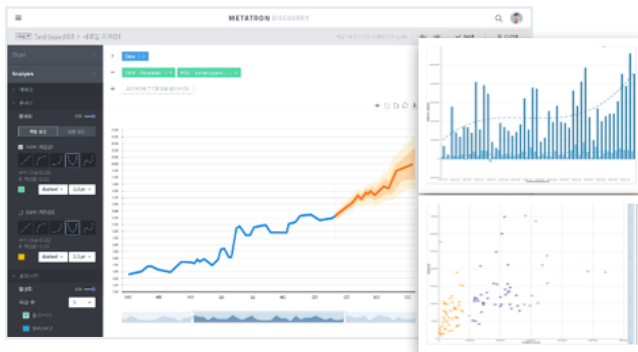




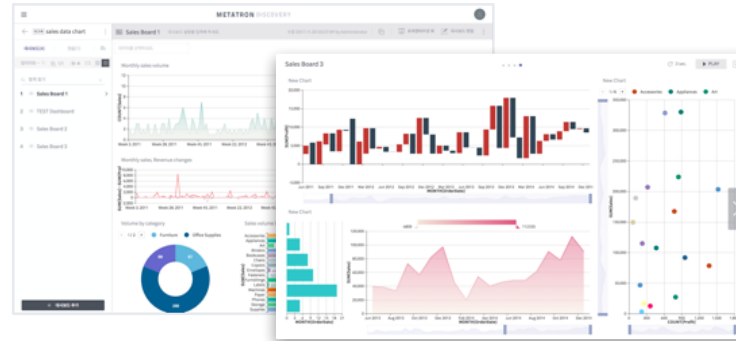
# Metatron Discovery – Tools & Management



Workspace



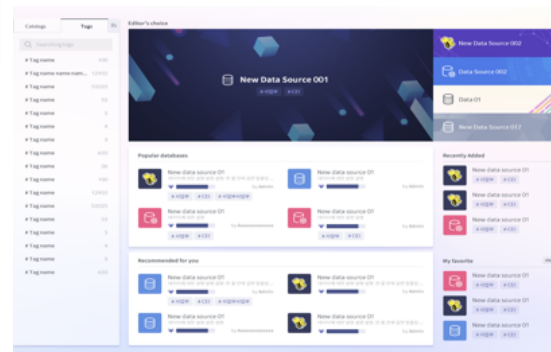
Embedded Analytics



Notebook - Dashboard/Chart



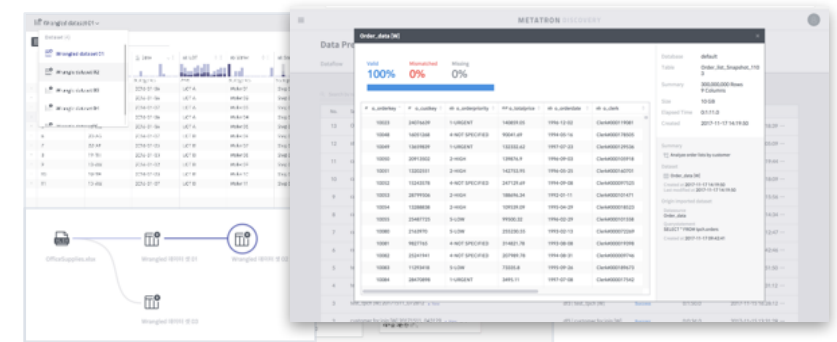
Data Lineage



MDM



Workbench – SQL Explorer



Data Prep. – Data Transformer



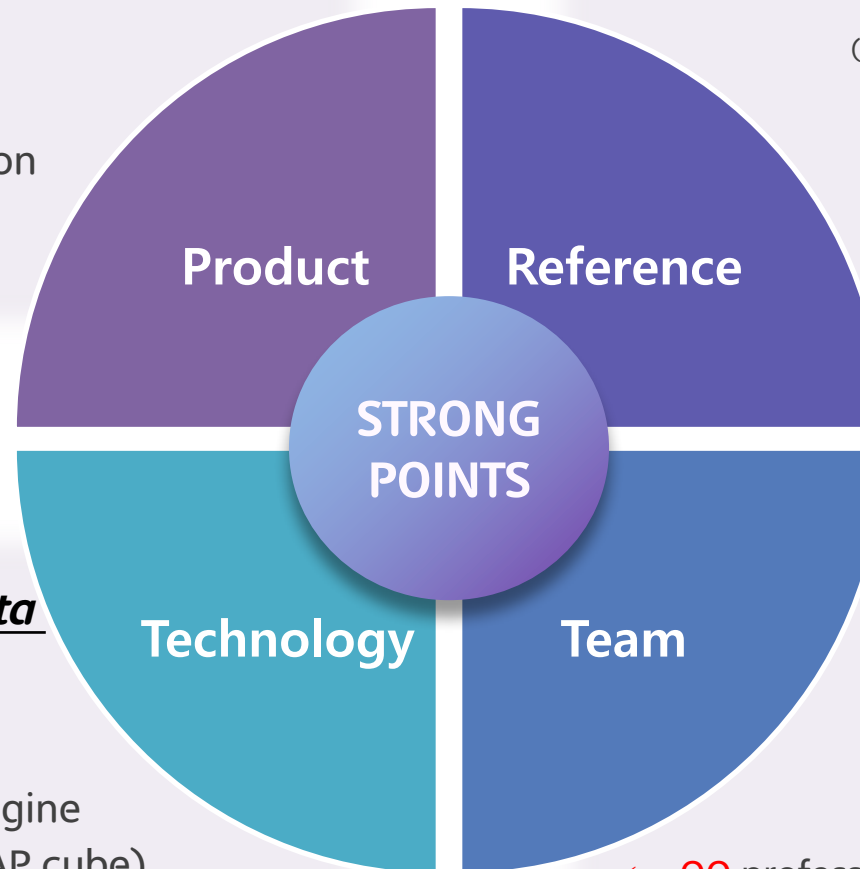
# Metatron Discovery – Features and Strengths

## Unified solution for data discovery

- ✓ Easy preparation
- ✓ Fast & intuitive visualization
- ✓ Powerful management

## Tested with various use cases

- ✓ Applied to 0 fields  
(telecommunications, finance, manufacturing, etc.)
- ✓ Applied to 00 sites  
(site A, site B, site C, etc.)
- ✓ Capable of processing up to 0000 queries/day



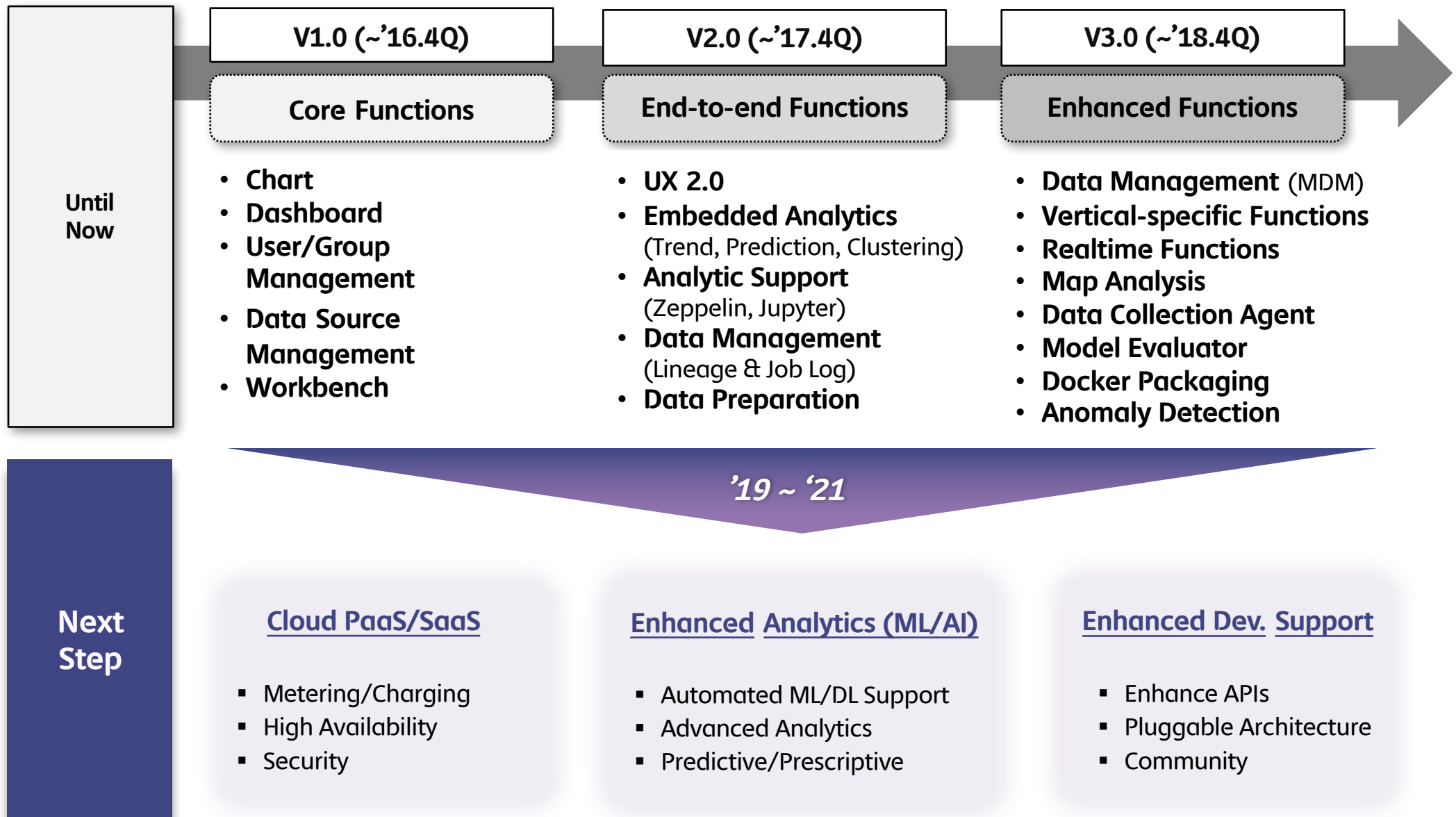
## Interactive time-series data processing and analytics optimization

- ✓ Sub-second processing engine
- ✓ Big data handing (big OLAP cube)
- ✓ Automated ML/DL support

## World-class experts in the fields of big data, analytics and IT

- ✓ 0 PMC members
- ✓ 00 professionals with 10+ years of experience
- ✓ 0 experts with PhDs

# Metatron Discovery – Roadmap

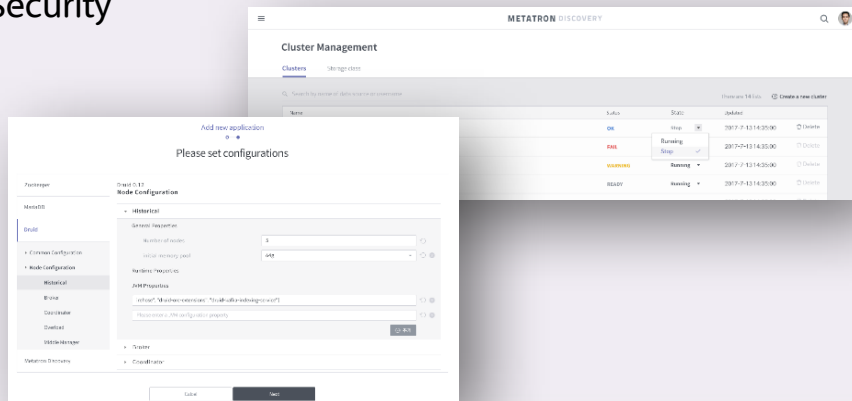




# Metatron Discovery – Next Step

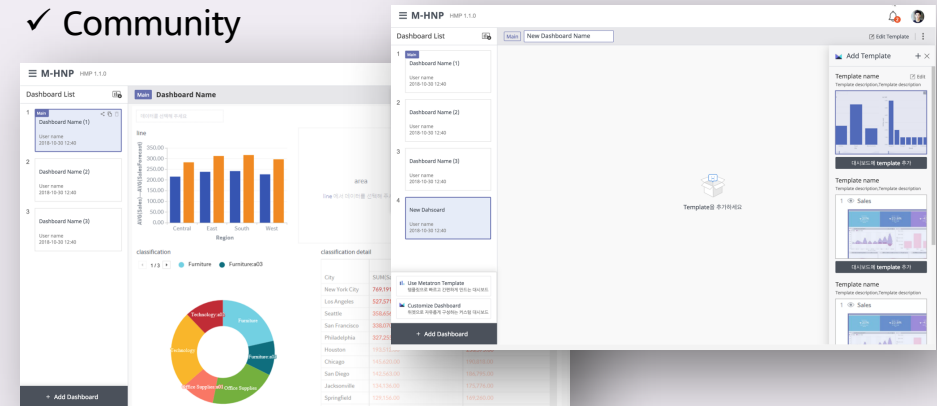
## Cloud PaaS/SaaS

- ✓ Metering/Charging
- ✓ High Availability
- ✓ Security



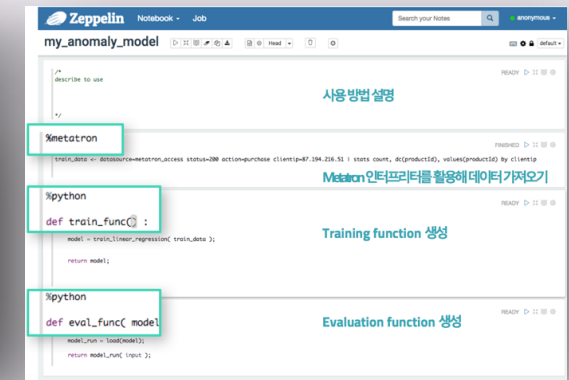
## Enhanced Development Support

- ✓ Enhance APIs
- ✓ Pluggable Architecture
- ✓ Community



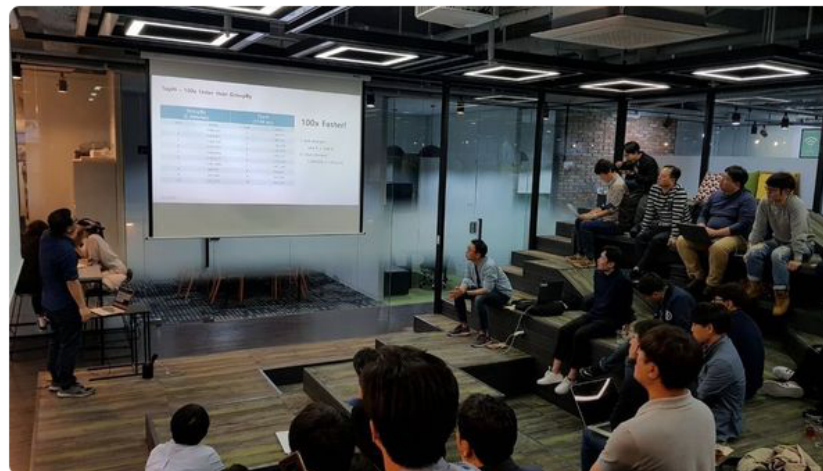
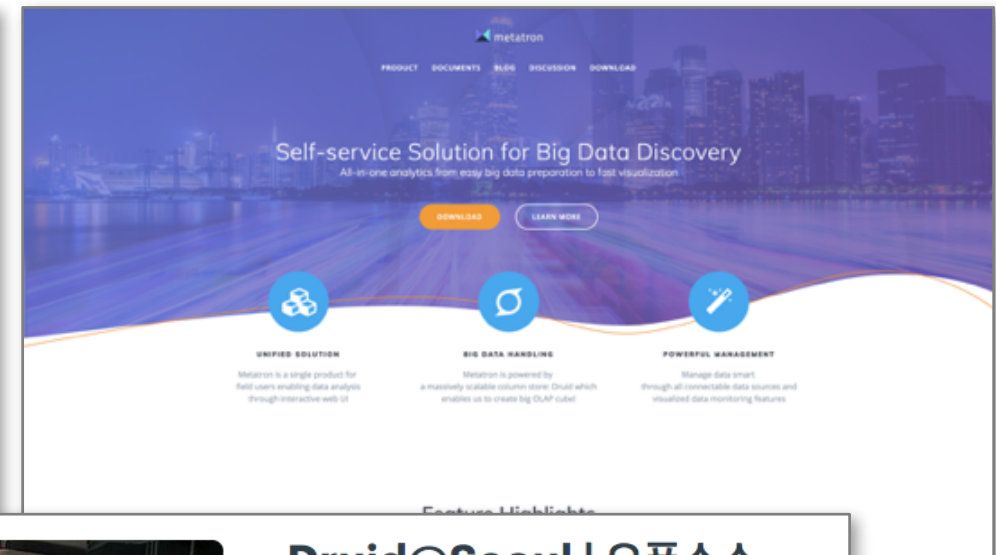
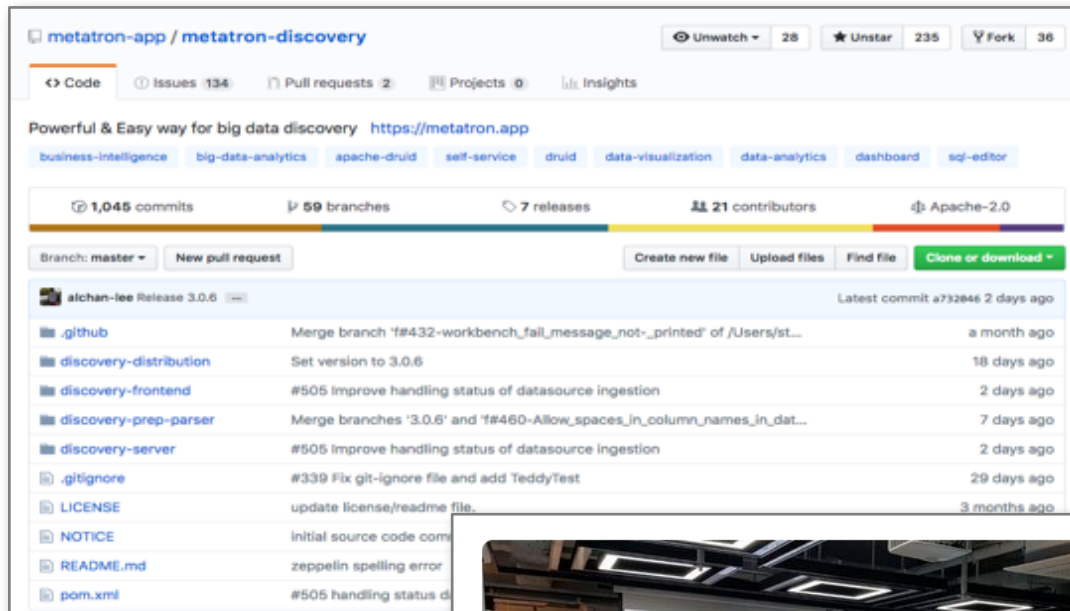
## Enhanced Analytics (ML/AI)

- ✓ Automated ML/DL Support
- ✓ Advanced Analytics
- ✓ Predictive/Prescriptive
- ✓ Co-working with Zeppelin



# Metatron Discovery – Open Source & Activities

- Website: <https://metatron.app/>
- Source code: <https://github.com/metatron-app/>
- Board meetings held offline once a month to share information on version releases, discuss plans for future releases, etc.
- Druid meetups held offline once a quarter, where developers and users share about Druid technology and application experiences, etc.  
<https://www.meetup.com/Druid-Seoul/>



## Druid@Seoul | 오픈소스 Druid 사용자 모임

Seoul, Korea (South) · 272 members · Public group



Organized by  
Navis R. and 5 others

Share: [f](#) [t](#) [in](#)

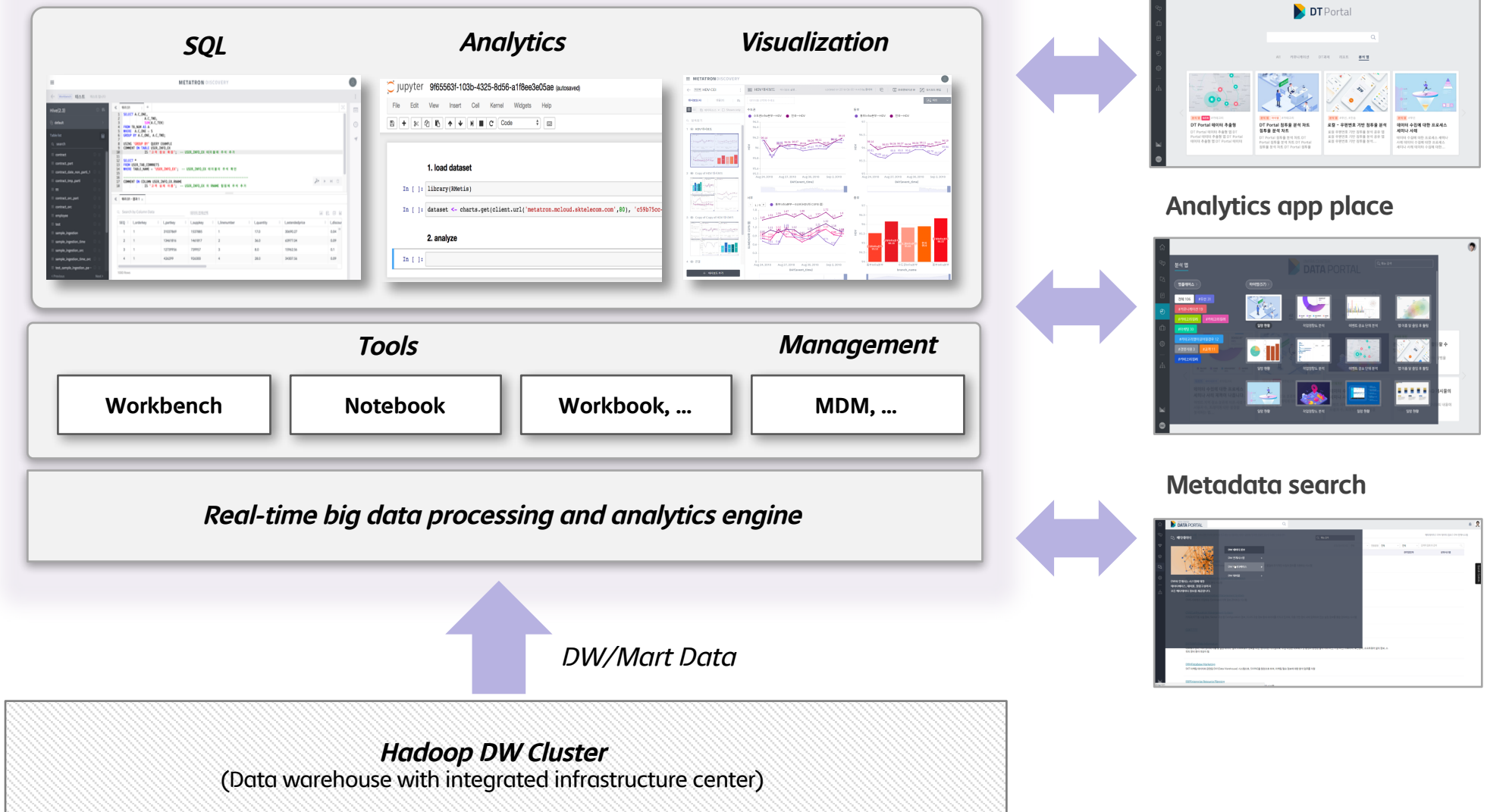




Metatron Discovery

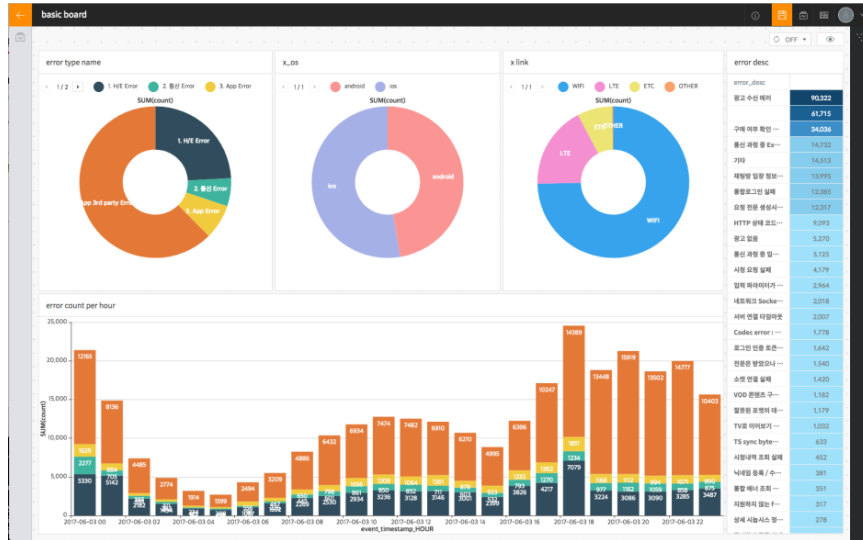
# Use Cases

# Use Case – Building an N/W Data Analytics Portal (Telecommunications)

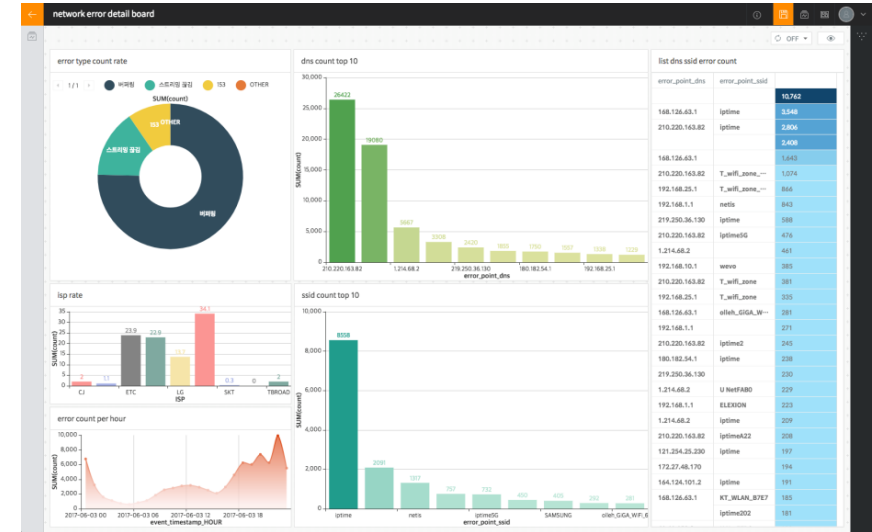


# Use Case – Log Analysis

- Error monitoring



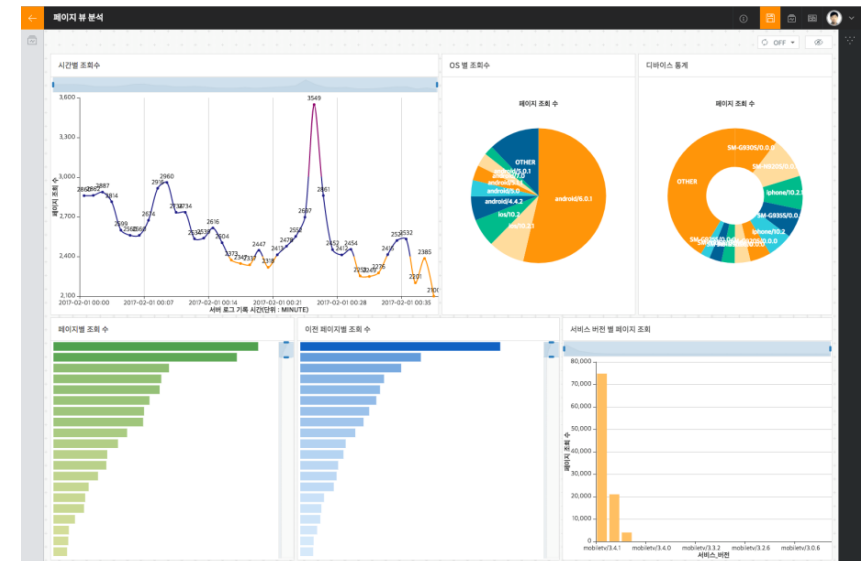
- Network error detail board



- OS device detail board



- Page view analysis



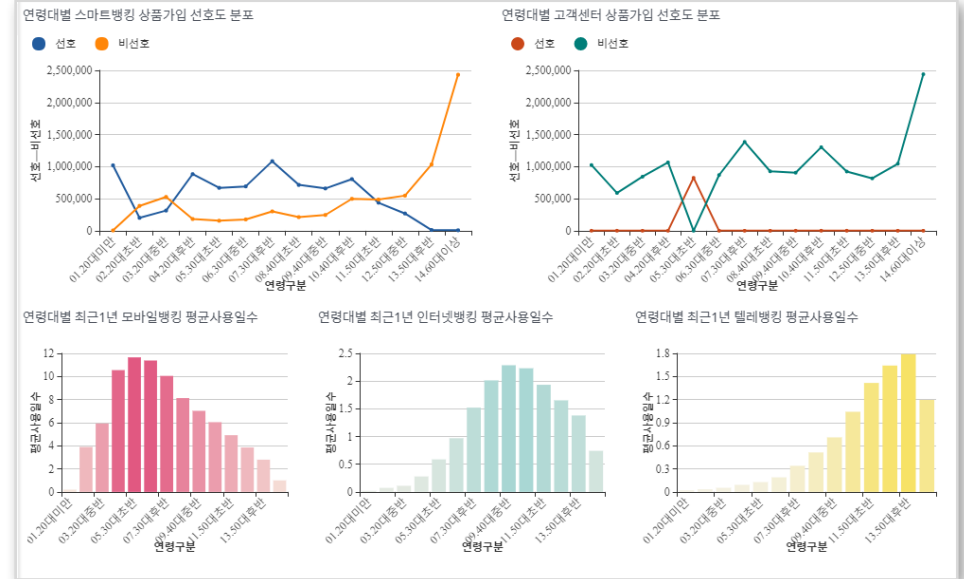


# Use Case – Finance

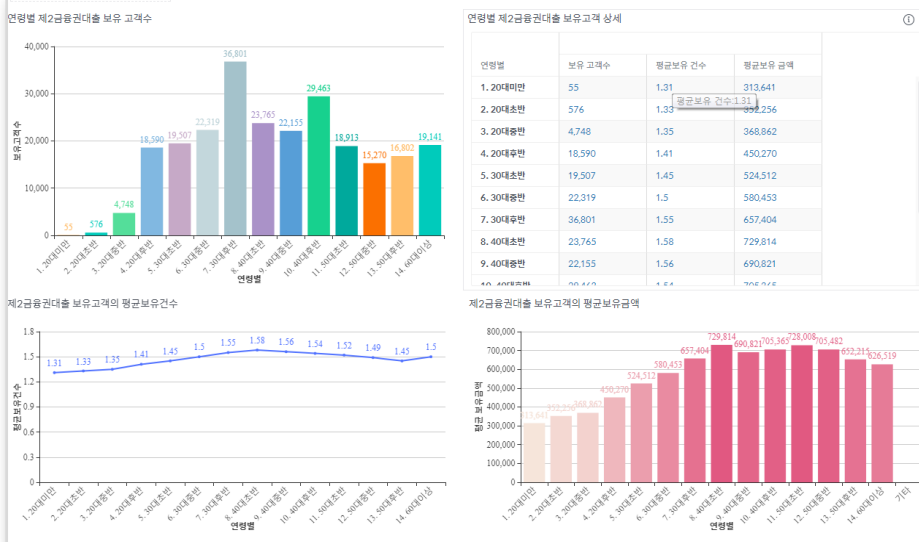
## Analysis 1



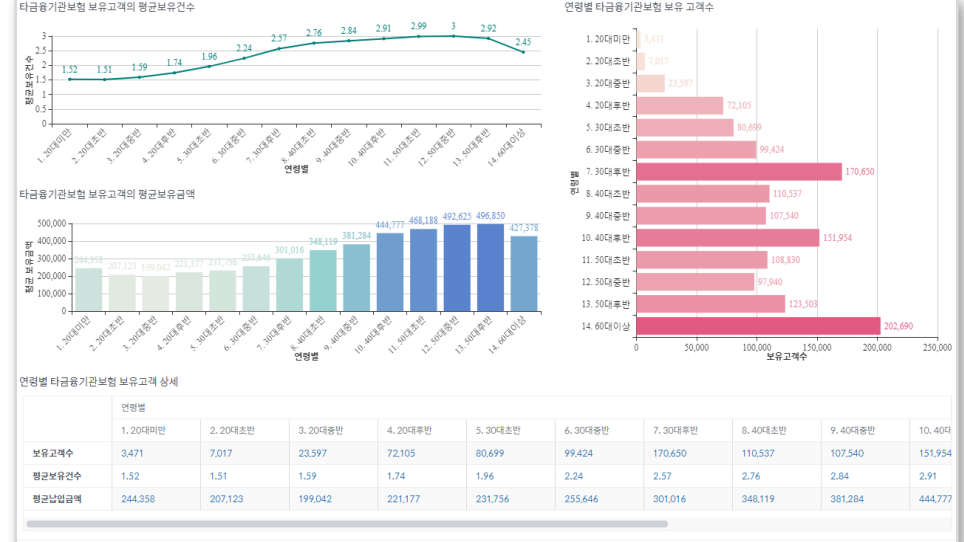
## Analysis 2



## Analysis 3

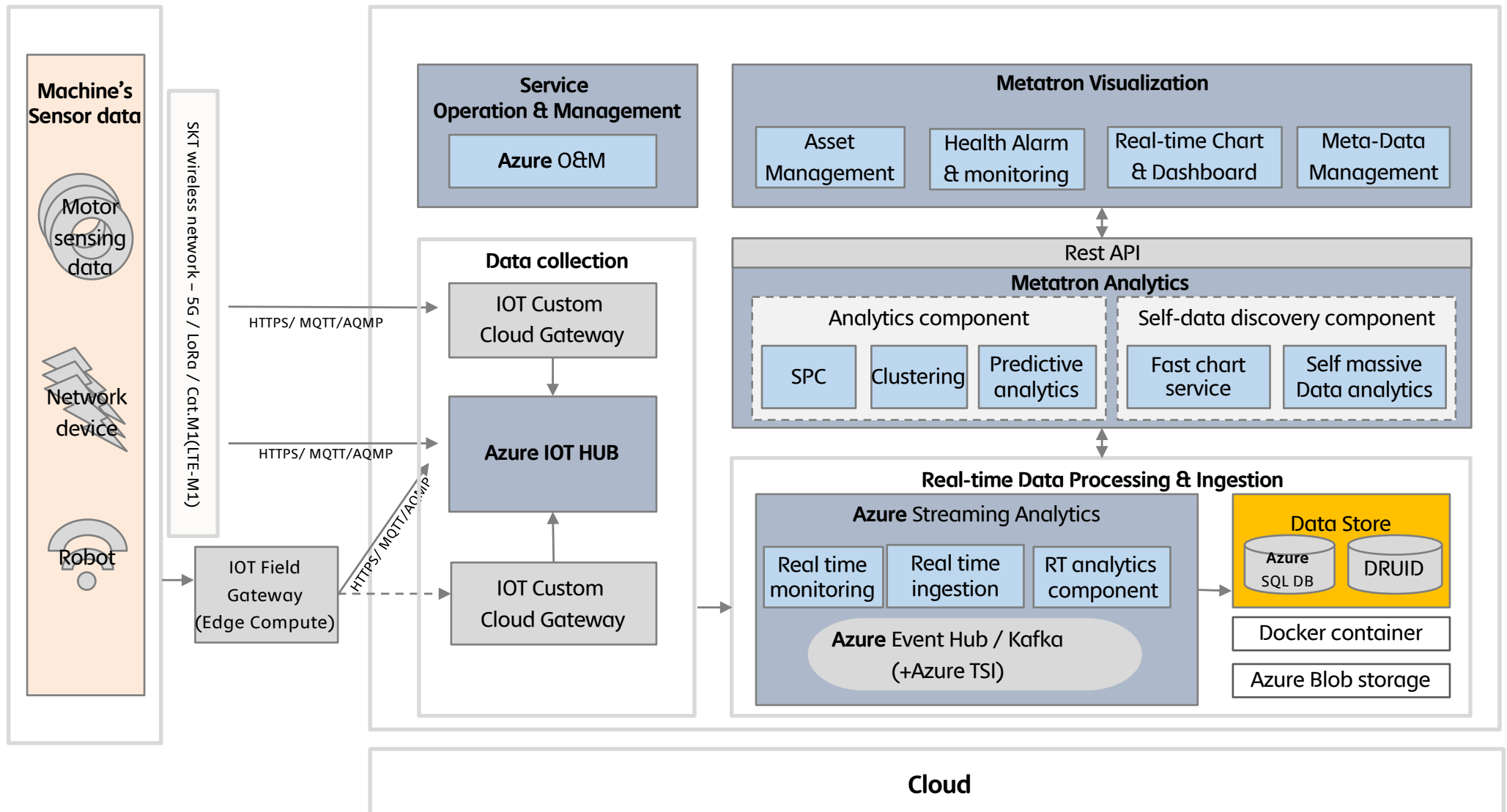


## Analysis 4



# Use Case – Manufacturing

## Machine Health Monitoring & Predictive Maintenance Analysis Service





An aerial photograph of a dense city skyline, likely New York City, taken from a high vantage point. The image is dominated by numerous skyscrapers and high-rise buildings. The sky is a mix of soft pinks, oranges, and blues, indicating the time is either dawn or dusk. A semi-transparent dark blue overlay covers the lower two-thirds of the image, providing a background for the text.

Metatron Discovery

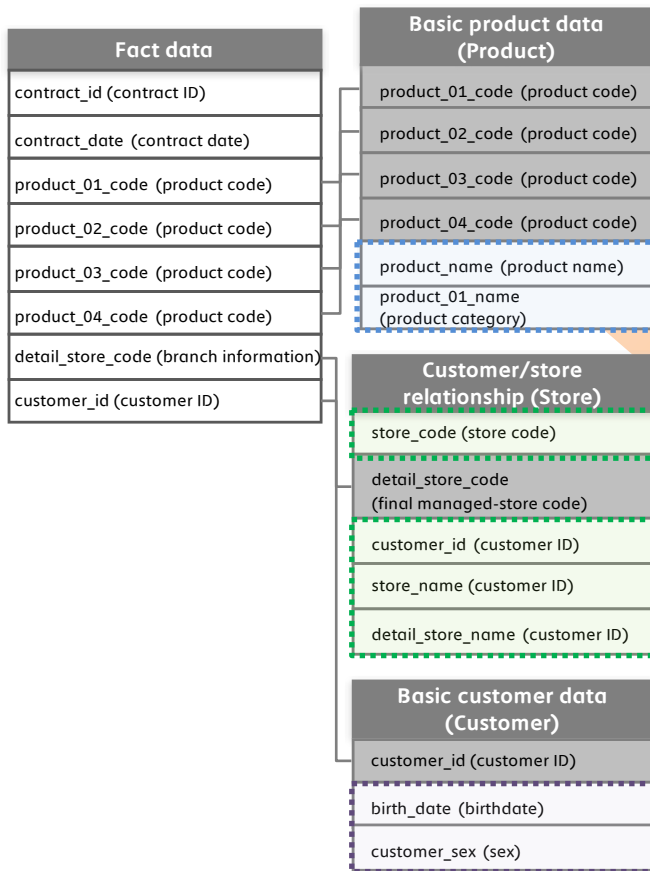
# Technology



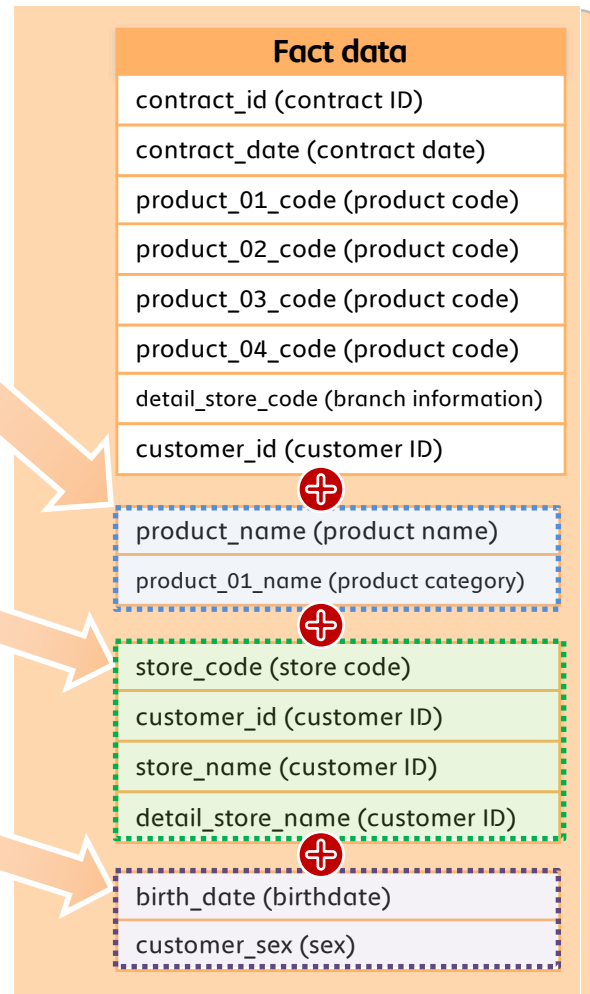
# Major Features - Big OLAP Cube

- ❖ Creation of one big OLAP cube (data mart) by combining various dimension data based on a large amount of fact data

## Legacy method



## Big OLAP Cube



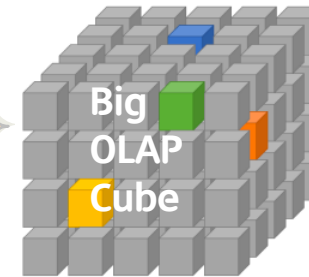
### Advantage ① ETL expense minimization

- Reduce ETL expenses for data mart creation by minimizing the number of data marts
- Minimize the effects of structural changes by reducing the volume of mart data
- Capable of meeting diverse demands by saving all fact data



### Advantage ② Fast speed and free schema changes

- A distributed architecture enables storing large amounts of data and ensures fast data processing
- With a dynamic schema approach, schema changes do not require schema redefining.



*Joined to create a big OLAP cube*

### Advantage ③ Real-time processing

- Can process data at the record level in real time as tables are stored with no data loss

## Major Features - Sub-Second Processing Engine (Optimized Druid for Metatron)

- ❖ Druid is a data processing engine optimized for time series data that was developed under the leadership of Impliedata and MetaMarkets starting in 2012.
- ❖ SKT developed core functions required for Metatron, and now has branched off as an independent project.

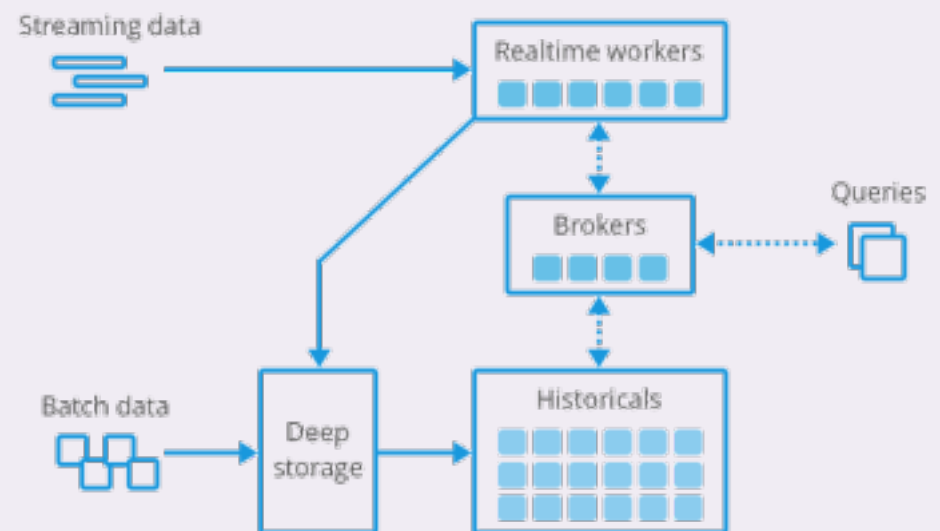
### ■ Development in open ecosystem

- Hosts Druid user meetups in S. Korea
- Contributes opinions and codes (300+) to Druid on Github and Google Groups

### ■ Not just simple use of open-source code, but in-house development of core functions, performance improvements, etc, (under a separate branch) including:

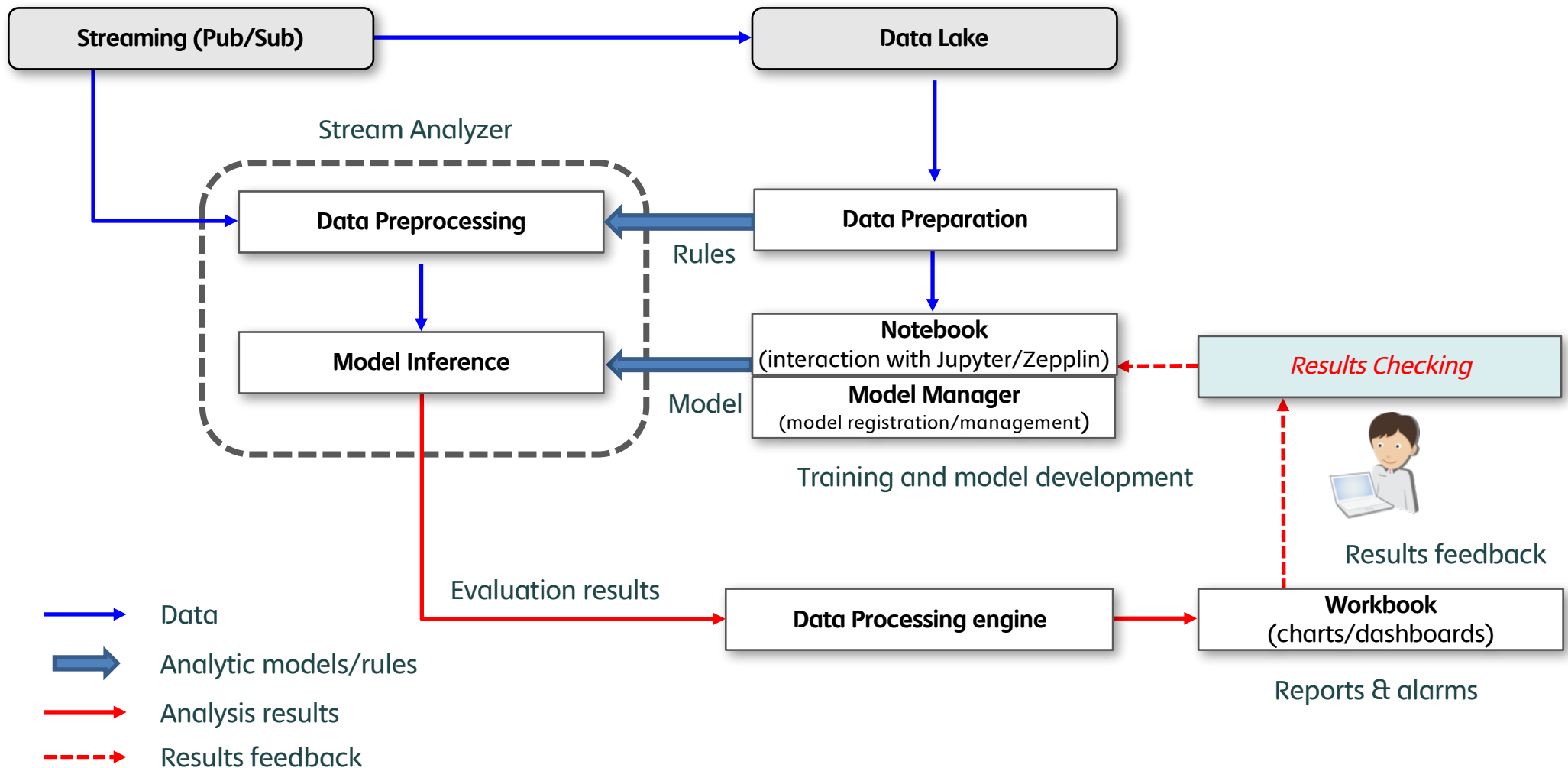
- Joins between data sources
- Data search (Lucene index, spatial index)
- Extended query set (sketch queries)
- Query statistics
- Window functions
- Virtual column map type
- Multi-valued metric
- ...

- Supports both real-time and batch processing
- Over time, data is transferred to memory, local storage, and then deep storage, by which the engine is capable of even terabytes of big data.
- Each function module (query processing, storage, and indexing) can be maintained on separate servers, which can be scaled out to accommodate increasing usage.



## Major Features – Automated ML/DL Support

- ❖ Automation of the entire process from ML/DL-based analytics model development to commercial application
- ❖ Reduced development schedules and lower costs by eliminating development processes for fitting a newly developed model into the system
- ❖ Employment of incremental learning, etc. to build systems that gradually improve over cycles







# Metatron Discovery Functions

<https://metatron.app/index.php/documents/>

# The Analytics Environment – Workspace

Workspace:  
Analytics space  
(personal, shared)

**Admin workspace** Owner

워크북 3 | 노트북 6 | 워크벤치 0 | 1 데이터 소스

**Permission settings**  
Select parties to share workspace content with 생성일 2017.11.20 by Administrator

Workspace List

Admin workspace

Search by content or folder name

전체

new folder

new folder1

new folder

new folder

notebook • New

s

마지막 업데이트일 20분 전

workbook

test3-workbook

마지막 업데이트일 8시간 전

workbook

jerry\_test

마지막 업데이트일 하루 전

notebook

sales r

마지막 업데이트일 2일 전

notebook

chart analysis

마지막 업데이트일 2일 전

notebook

notebooktest001

마지막 업데이트일 2일 전

notebook

notebooktest002

마지막 업데이트일 2일 전

notebook

sales py

마지막 업데이트일 2일 전

Module selection  
Select workbook, notebook, or workbench as an analytics module

전체 선택 | Clone Workbook | Move selections | Delete selections

1 + 워크북

2 + 노트북

3 + 워크벤치

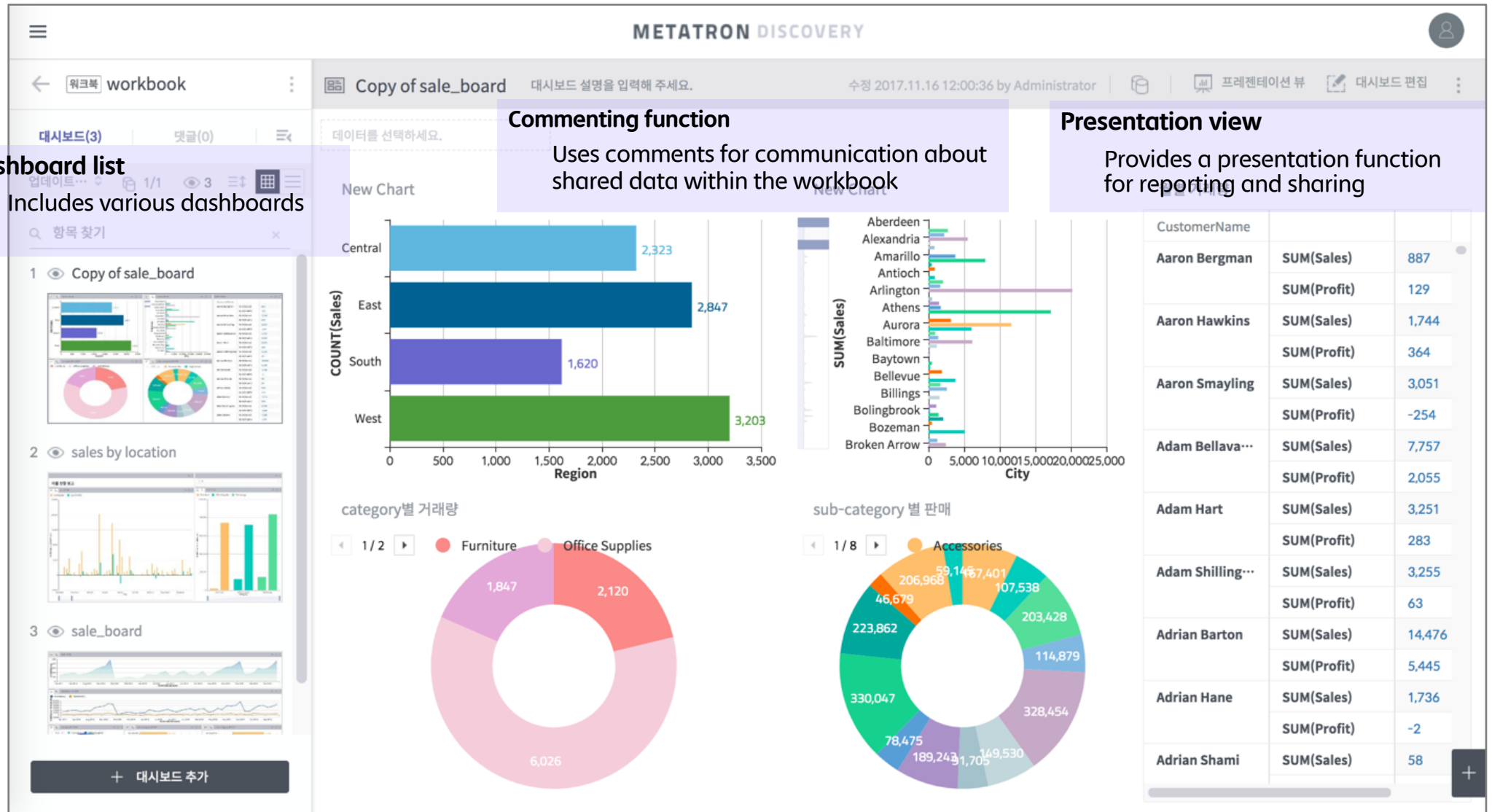
1 Workbook:  
Visualization-based analytics

2 Notebook:  
ML-based advanced analytics

3 Workbench:  
SQL-based analytics

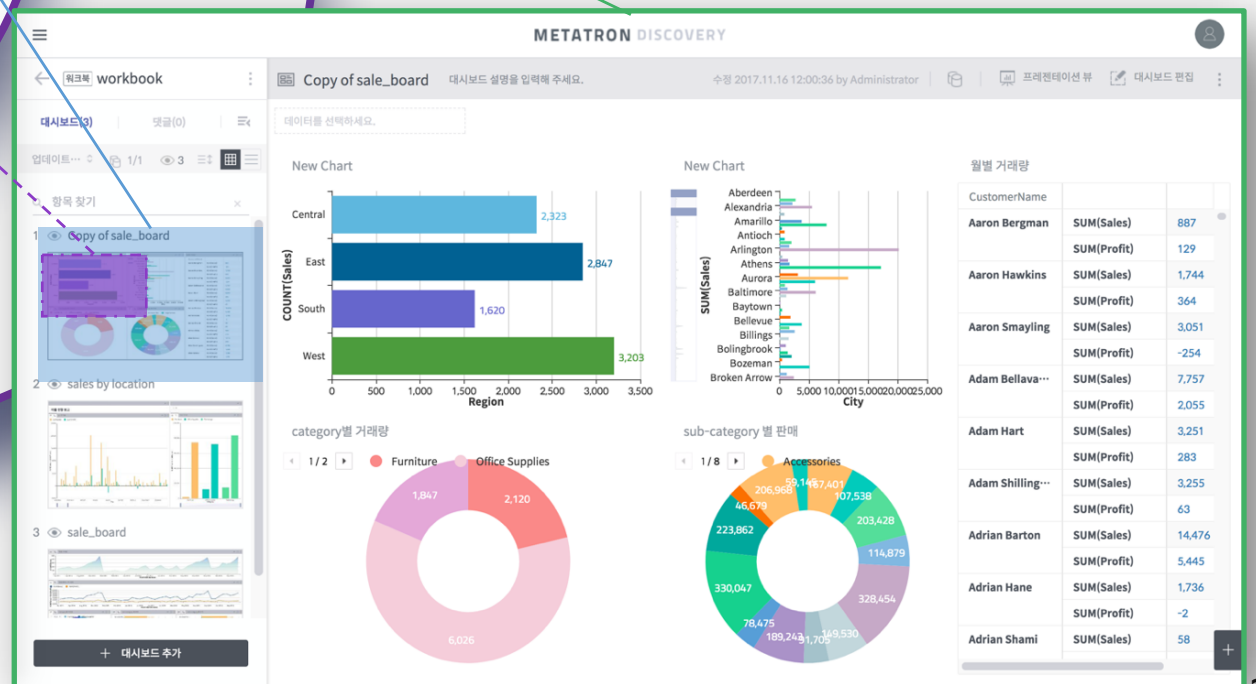
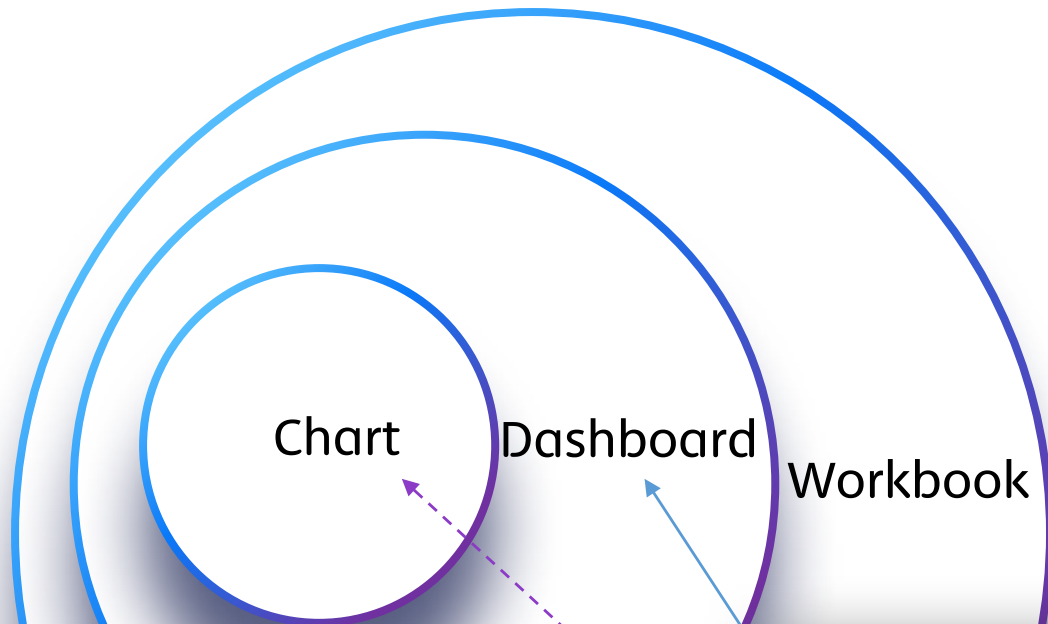
# 1 Workbook – Visualization-Based Analytics

- The PowerPoint-type environment provides a familiar and attractive user experience.

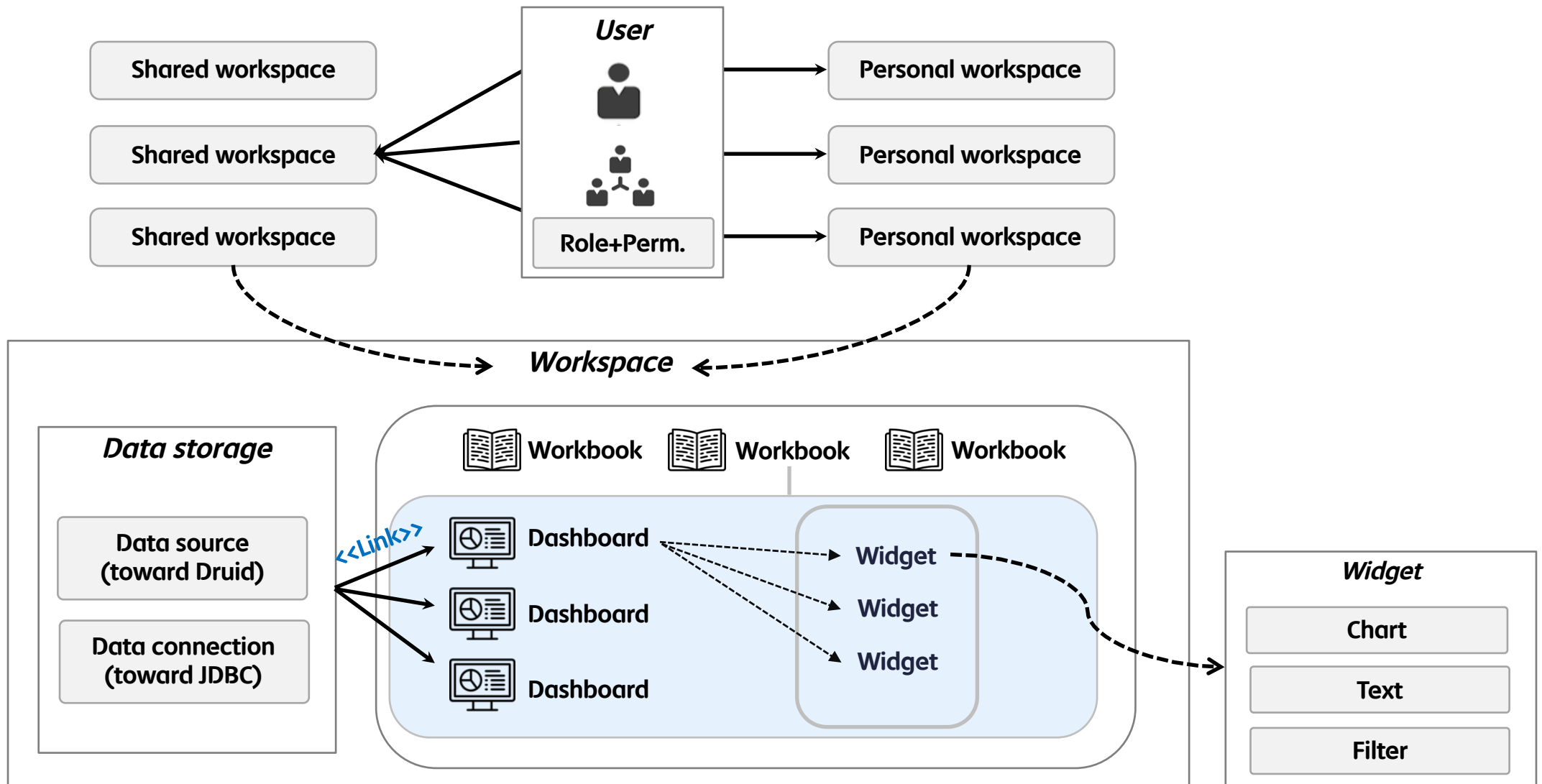




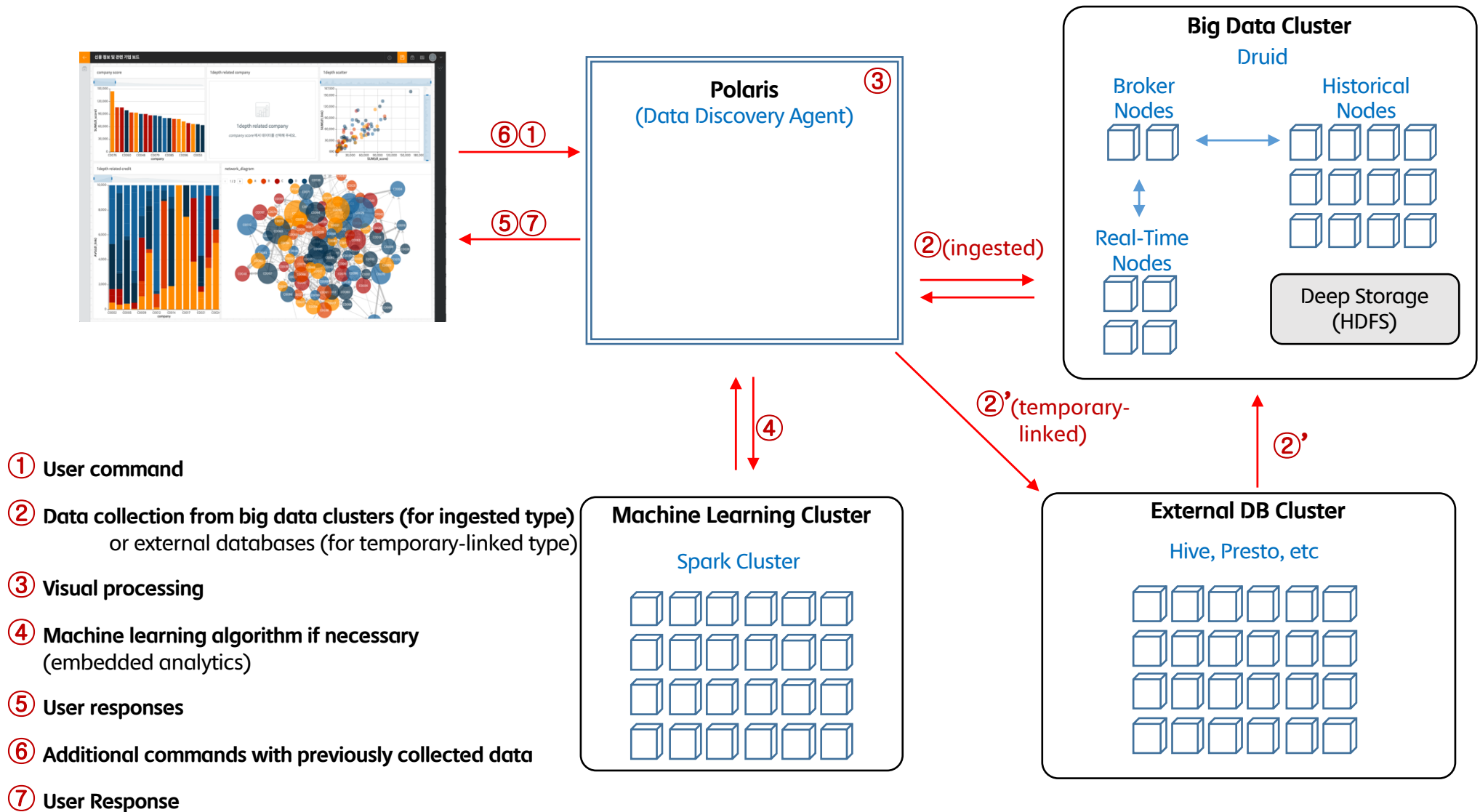
## (Reference) Workbook hierarchy



# Workbook structure



# Workbook Operation (Data Visualization)

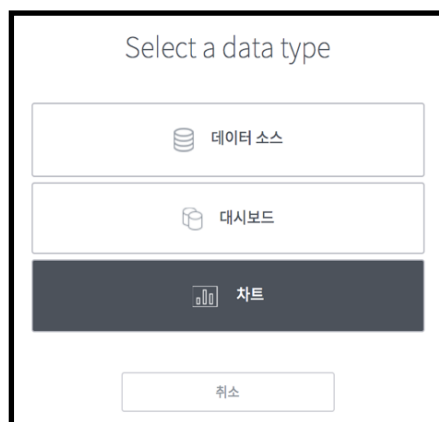




## 2 Notebook – ML-Based Advanced Analytics

### ■ Selection of analytics target

- Choose data source, dashboard, or chart
- Select data for analysis



Select a data type

데이터 소스

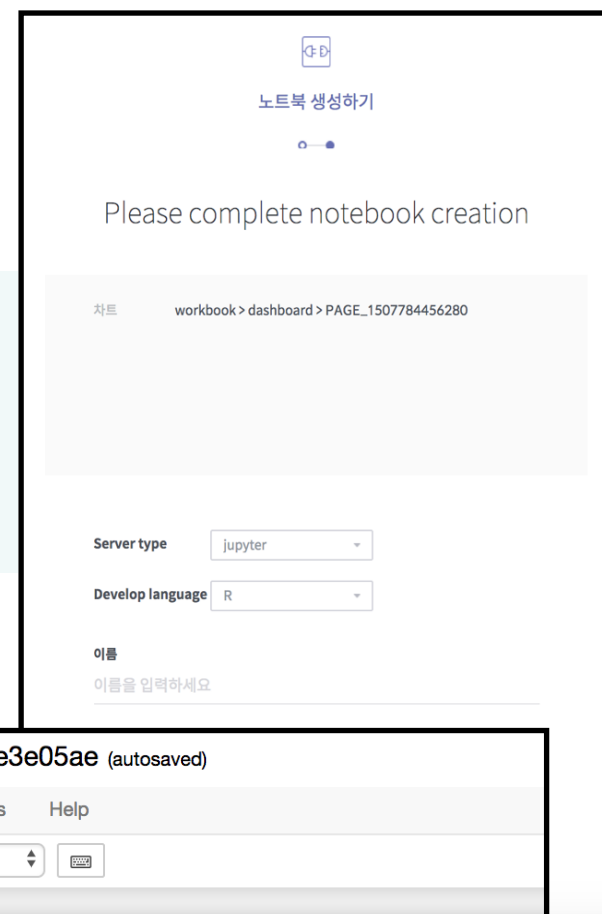
대시보드

차트

취소

### ■ Notebook details entry and analysis

- Select server type
- Select development language (Jupyter: R/PYTHON, Zeppelin: Spark)
- Enter notebook name and description



노트북 생성하기

Please complete notebook creation

차트 workbook > dashboard > PAGE\_1507784456280

Server type jupyter

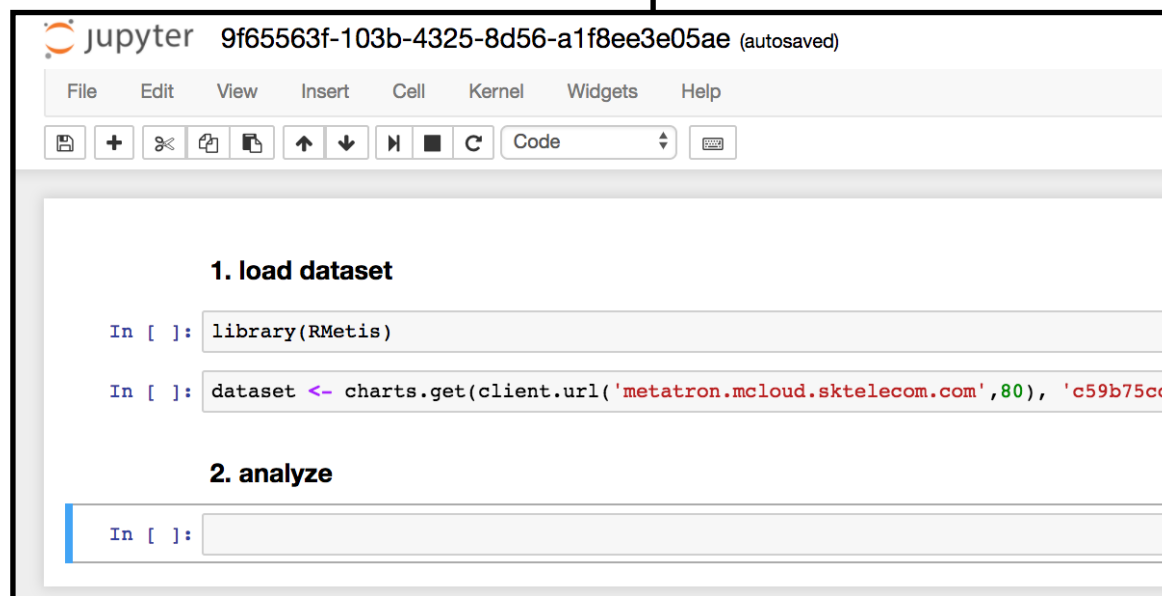
Develop language R

이름

이름을 입력하세요

### ■ Jupyter notebook pop-up

- Write code to load the target dataset to analyze with the development language
- Saved by analyst after writing R code



```
jupyter 9f65563f-103b-4325-8d56-a1f8ee3e05ae (autosaved)
```

File Edit View Insert Cell Kernel Widgets Help

Code

1. load dataset

```
In [ ]: library(RMetis)
```

```
In [ ]: dataset <- charts.get(client.url('metatron.mcloud.sktelecom.com',80), 'c59b75cc-
```

2. analyze

```
In [ ]:
```

# Notebook – ML-Based Advanced Analytics

## ■ Notebook API creation

- Select return type (html, json)
- notebook code  
`response.write(_user_object_)`

Notebook

sales analysis


Data type

CHART


Data source

PAGE\_1507685416920

Develop language

 R

Code

[Detail](#) 

Name

sales analysis report

Description

URL

<http://metatron.mcloud.sktelecom.com/>

Return type

HTML

API result

Result

☒ Edit API  Delete API

### API information

Return Type ☒ HTML ☐ JSON 

이름

이름을 입력하세요

설명

설명을 입력하세요

Cancel

Done

## sales analysis report

james

Thu Oct 12 15:41:21 2017

```
library(RMetis)
```

```
## Loading required package: httr
```

```
## Loading required package: jsonlite
```

```
## Loading required package: methods
```

```
dataset <- charts.get(client.url('localhost',4200), 'c59b75cc-5b1d-4c7a-b5fa-e7ec8e29c16c')
```

```
summary(dataset)
```

```
##      SUM(Sales)      Category  
## Min.   :719127      Length:3  
## 1st Qu.:730566      Class :character  
## Median :742006      Mode  :character  
## Mean   :765785  
## 3rd Qu.:789114  
## Max.   :836221
```

```
regression_model <- list(coefficients = 2.5, intercept = 0.0)  
response.write(regression_model)
```

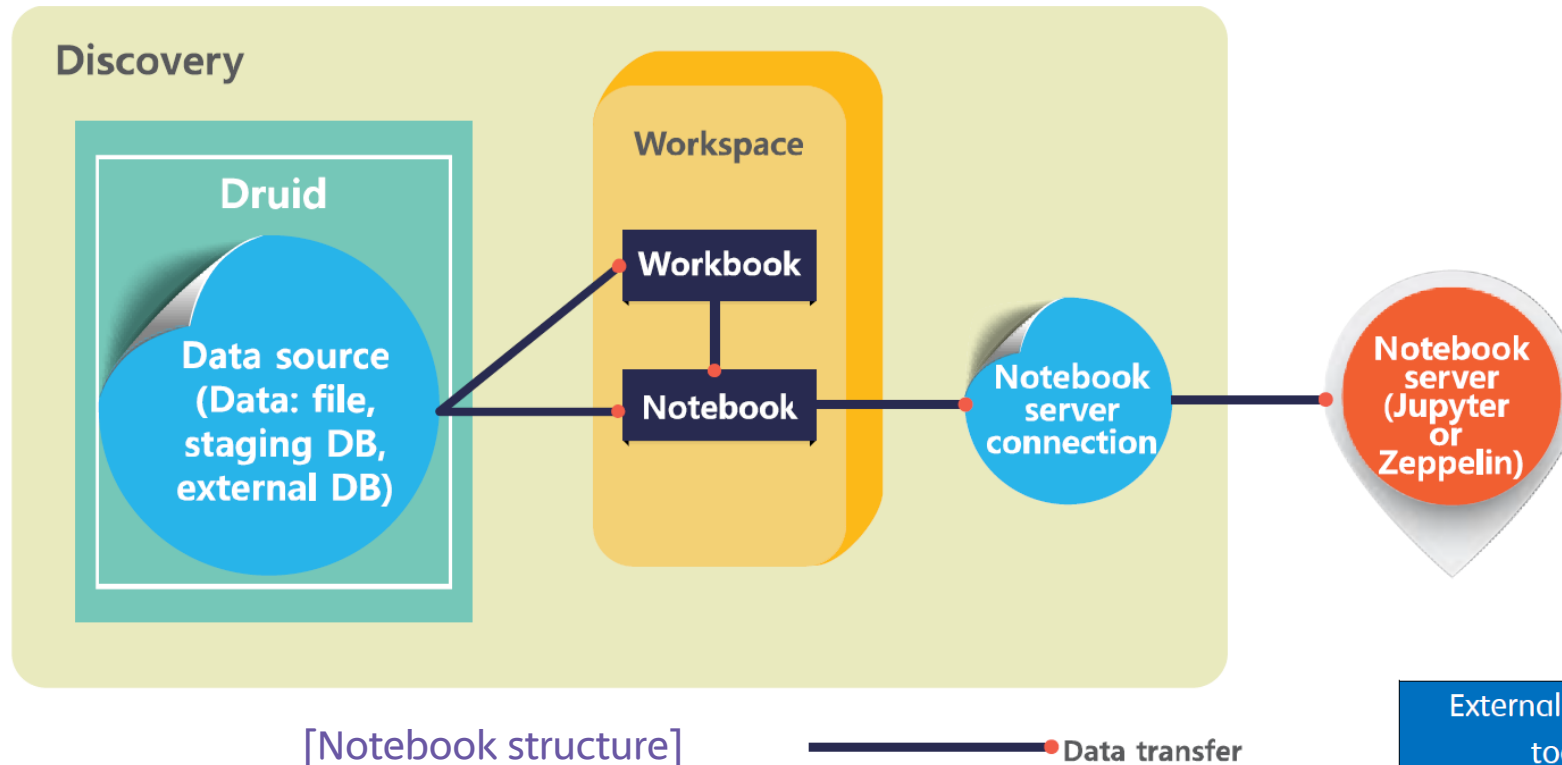
```
## {"coefficients":[2.5],"intercept":[0]}
```

## ■ RESTful service provided through generated URL

- Click to view results

```
{"coefficients":[2.5],"intercept":[0]}
```

# Notebook Structure



External analysis tools	Available languages
Jupyter	R
	Python
Zeppelin	Spark

[External analytics tools and available languages]



### 3 Workbench – SQL-Based Analytics

**Schema view**  
View table information, data preview and column schema

**History**  
View the query history and results list

**SQL statements execution**  
Execute all statements or a block of them

**Dynamic chart creation**  
Configure the data source dynamically while performing chart creation tests

**Shortcut to data source creation**  
Ingest query results directly into the data source

**Online Excel view**  
View results data on linked Online Excel

**Table list**

- contract\_temp
- contract\_orc\_part
- abc1
- employee
- sample\_ingestion
- sample\_ingestion\_time

**SQL Editor**

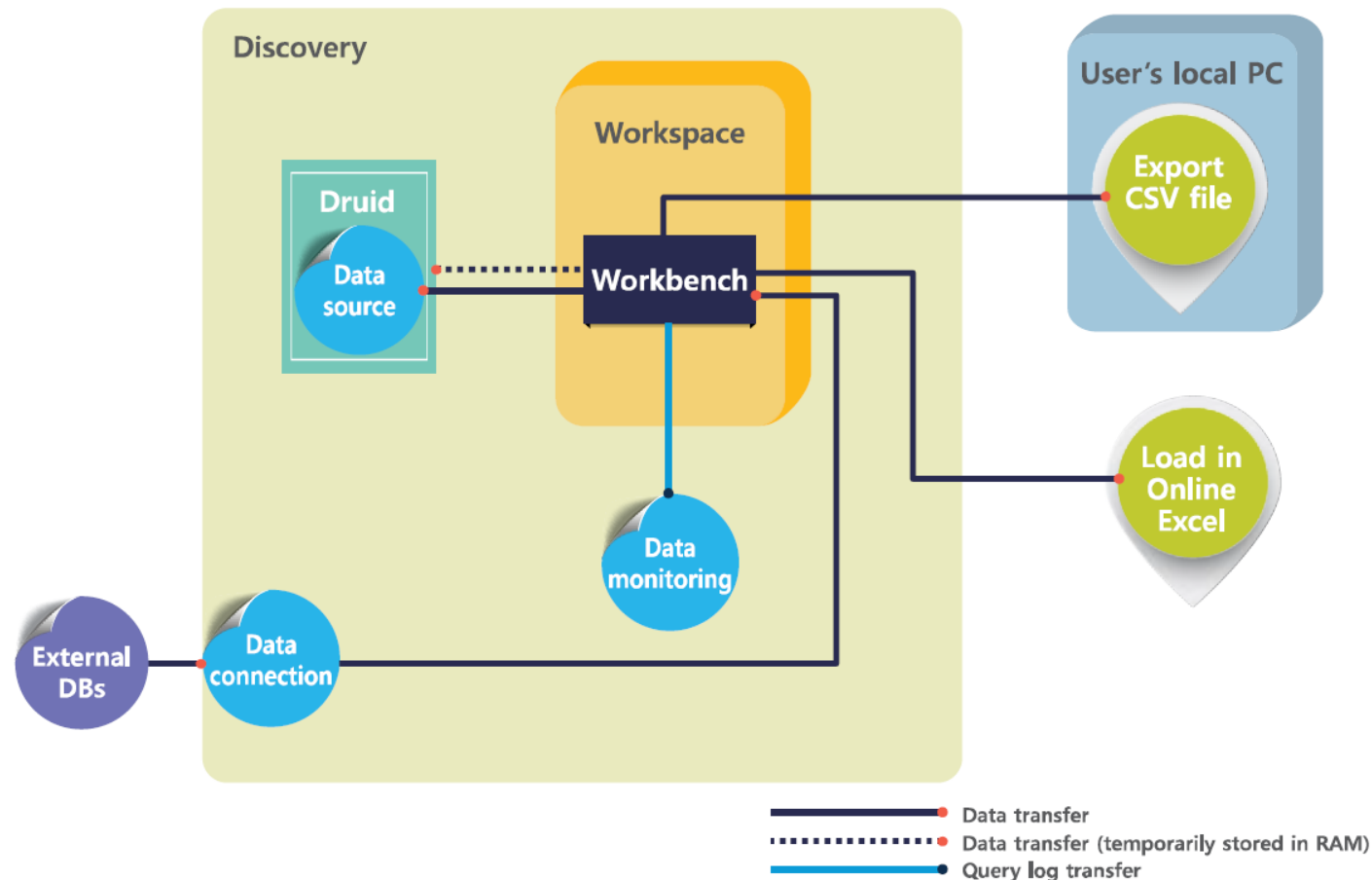
```
1 SELECT
2 *
3 FROM
4 default.contract_orc_part;
```

**Results Table**

	contract_id	product_01_code	product_02_code	product_03_code	product_04_code
1	cid000000001	01	01	004	0002
2	cid000004158	04	04	004	0025
3	cid000005953	02	04	004	0034
4	cid000005982	01	02	004	0030
5	cid000006210	02	02	003	0019
6	cid000014008	02	04	005	0005
7	cid000016535	02	05	005	0014
8	cid000021993	02	03	003	0044
9	cid000022344	04	04	005	0041

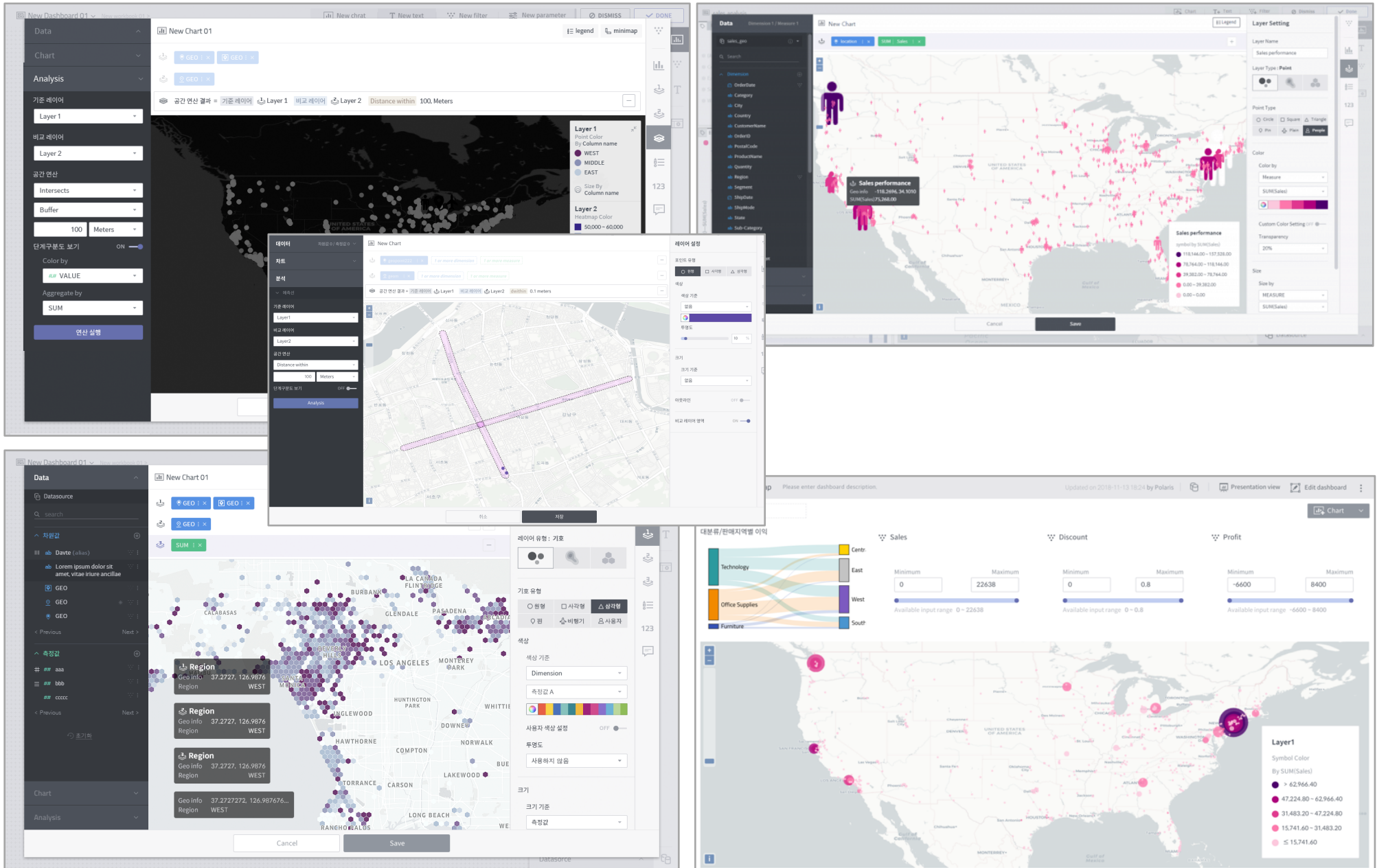
37 Rows

# Workbench Structure and Features



- [Various external databases](#) (Oracle, MySQL, Hive, Presto, Tibero) on different servers can be loaded in one space.
- [Easy and free exploration on connected database schemas](#) facilitates views and selection of tables and columns.
- A built-in query editor enables [easy querying, adding, deleting and editing of data](#).
- [Query results are shown in real time](#) and downloadable as local files.
- Data outputs are [visualized with a variety of charts](#) by importing them into the Druid engine.
- Query logs are accessible in the data monitoring menu.

# Map-based Analytics



# Data Preparation

## ■ ETL tasks for data visualization and analytics that used to be boring and repetitive can now be performed easily by anyone

- A consistent abstraction level ensured for different data sources (RDB, Hive, log files, etc.)
- Back-and-forth, step-by-step editing by looking at and comparing pre- and post-processed sets of sample data

ETL functionality for visualizing and analyzing data

Creation of data snapshots

View results data on linked Online Excel

ETL

The image displays a collage of screenshots from the METATRON DISCOVERY interface, illustrating various ETL (Extract, Transform, Load) workflow components:

- Customer\_data [W]:** A dataset view showing columns like Customer\_ID, c\_name, c\_address, and c\_nationkey. It includes a 'Join' configuration window for selecting datasets to join.
- Rule list:** A list of transformation rules applied to the data, such as 'rename rename col: c\_custkey to: 'C customer\_ID'', 'split split col: c\_phone on: '/' limit: 3 ignoreCase: true', and 'settype settype col: split\_c\_phone1 t type: Integer'.
- Data preview:** A window showing a sample of the processed data, displaying columns like Customer\_ID, c\_name, c\_address, c\_nationkey, c\_acctbal, c\_mktsegment, c\_com, Tel\_Region\_Code, split\_c\_phone2, and split\_c\_phone1.
- Dataflow diagram:** A visual representation of the ETL process, showing the flow of data from source datasets (Customer\_data, Nation\_code, Region\_code, Order\_data) through various transformations and joins to the final output (Order\_data [W]).

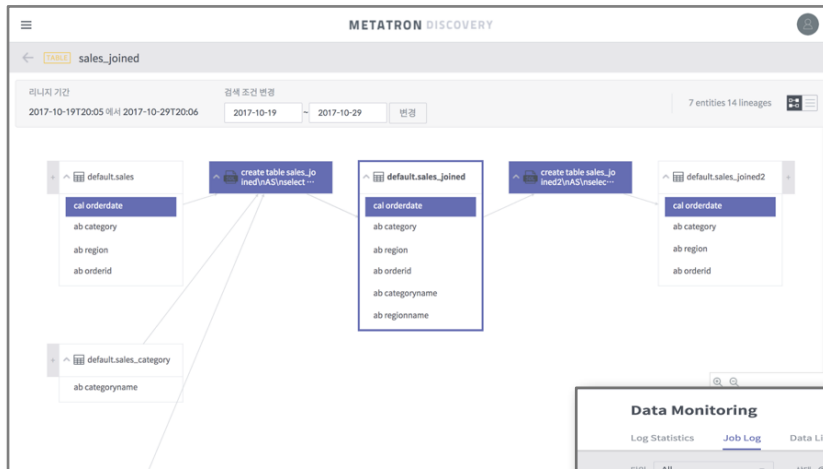
Creation of data flows

Application of rules



# Data Management (Lineage & Job Log)

- Data Lineage: Manages relationships between pre- and post-processed data using query logs
- Job Log: Manages the history of queries issued in workbenches and various other statistical information



## Column history using graphing

An MDM view of the connections of each data query with tables and columns

The screenshot shows the 'Data Monitoring' section with the 'Job Log' tab selected. It displays a table of executed queries with columns for status, job name, job ID, application ID, queue, user, start time, and execution time. The table lists several successful queries, including 'select \* from polaris\_v2.book', 'select \* from book', 'SELECT \* FROM polaris\_v2.book', 'SELECT \* FROM polaris\_v2.audit', 'SELECT \* FROM polaris\_v2.audit', 'select \* from polaris\_v2.book', 'select \* from polaris\_v2.book', 'select \* from polaris\_v2.book', 'select \* from polaris\_v2.analysis\_binary', 'select \* from polaris\_v2.analysis\_binary', 'select \* from polaris\_v2.book\_notebook', and 'SHOW SCHEMAS'.

상태	Job name	Job ID	Application ID	Queue	사용자 (ID)	시작시간	작업시간
SUCCESS	select * from polaris_v2.book				admin	2018-02-20 15:00:00	0 sec
SUCCESS	select * from polaris_v2.book				adm		
SUCCESS	select * from book				adm		
SUCCESS	SELECT * FROM polaris_v2.book				adm		
SUCCESS	SELECT * FROM polaris_v2.audit				adm		
SUCCESS	SELECT * FROM polaris_v2.audit				adm		
SUCCESS	select * from polaris_v2.book				adm		
SUCCESS	select * from polaris_v2.book				adm		
SUCCESS	select * from polaris_v2.book				adm		
SUCCESS	select * from polaris_v2.analysis_binary				adm		
SUCCESS	select * from polaris_v2.analysis_binary				adm		
SUCCESS	select * from polaris_v2.book_notebook				adm		
SUCCESS	SHOW SCHEMAS				had		

## View and search of query history

Detail view of individual queries

The screenshot shows the 'METATRON DISCOVERY' interface with the 'Query History' tab selected. It displays the details of a specific query: 'select \* from polaris\_v2.book'. The query was performed on 2018-02-20 11:35:22 by user 'admin'. The connection information is shown as MySQL, host 'metatron-poc-a01', port '3306', and JDBC URL 'jdbc:mysql://metatron-poc-a01:3306/'. Below this, a table lists the query history with columns for query date, user, execution time, result, and a link to the detail view.

Query date	User	작업시간	Result	
2018-02-20 15:50:40	Administrator	2 ms	SUCCESS	<a href="#">Detail&gt;</a>
2018-02-20 13:51:08	Administrator	2 ms	SUCCESS	<a href="#">Detail&gt;</a>
2018-02-20 13:50:50	Administrator	1 ms	SUCCESS	<a href="#">Detail&gt;</a>
2018-02-20 13:36:30	Administrator	2 ms	SUCCESS	<a href="#">Detail&gt;</a>
2018-02-20 13:36:12	Administrator	32 ms	SUCCESS	<a href="#">Detail&gt;</a>

# Integrator

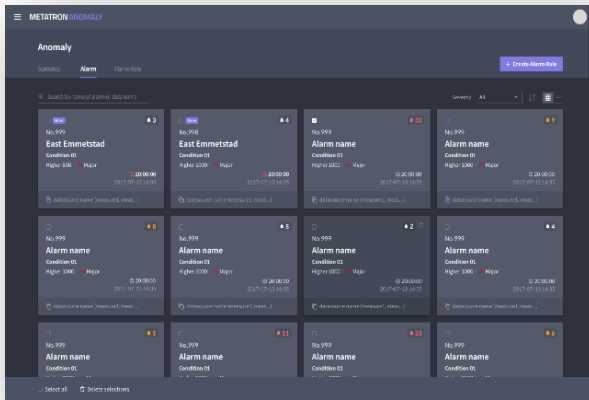
- Provides a GUI for registering and managing periodic jobs and viewing their processing results

The image displays the METATRON DISCOVERY GUI, which is used for managing periodic jobs and viewing their processing results. The interface is divided into several sections:

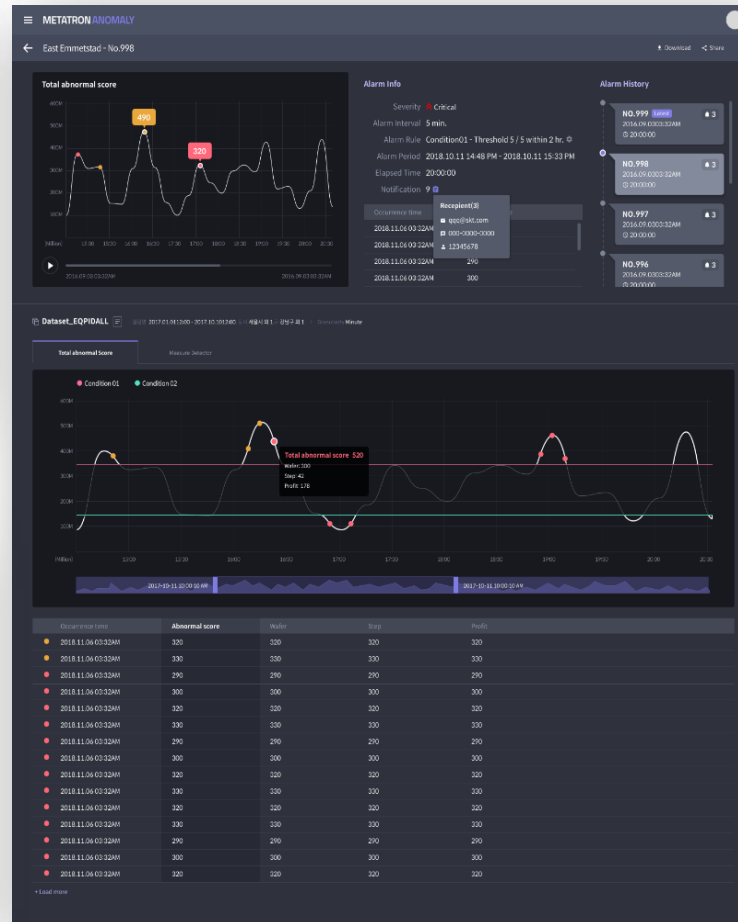
- Workflow settings:** This section on the left allows users to configure workflow details. It includes fields for "Tags" (with a placeholder "# 태그명을 입력하세요"), "Execution ID" (set to "rpda"), and an "Alert" toggle (currently OFF).
- Workflow diagram:** The top section shows a visual representation of the workflow. It starts with a "START" node, followed by a sequence of "HIVE" and "FS" (File System) nodes, and ends with an "END" node. The workflow is titled "rp\_da\_gps\_link\_matching\_process" and is identified as an "RPDA GPS Map Matching JOB".
- Manual run / Scheduled run:** This section provides options to manually execute the workflow or schedule it. It includes a "Time" selector (set to "오늘" - Today) and a "Status" filter (set to "ALL").
- Monitoring dashboard:** The bottom right section shows a Gantt chart for monitoring job execution. It displays a timeline for "09.26. Wed" and "09.27. Thu". The chart shows various job statuses: "Prep" (yellow), "Running" (green), "Succeeded" (blue), "Suspended" (purple), "Failed" (red), and "Killed" (grey). A legend on the right explains these statuses: "Prep" (Job 이 준비 중인 상태), "Running" (Job 이 실행 중인 상태), "Succeeded" (Job 이 일시 중지된 상태), "Suspended" (Running 중이었던 Job 이 성공적으로 완료된 상태), "Failed" (Running 중이었던 Job 이 오류가 발생하여 실패한 상태), and "Killed" (사용자에 의해 Job 이 중단된 상태).

# Anomaly Detection

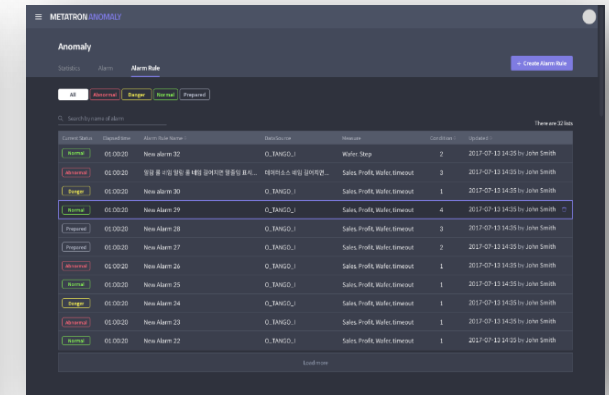
- Detects anomalous situations using an ML prediction model and generates alarms on them for immediate actions



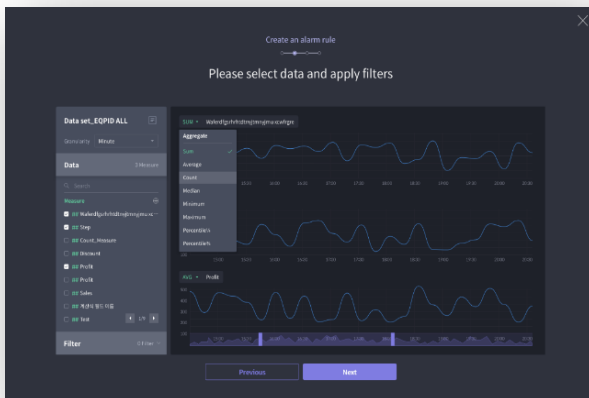
Alarm list



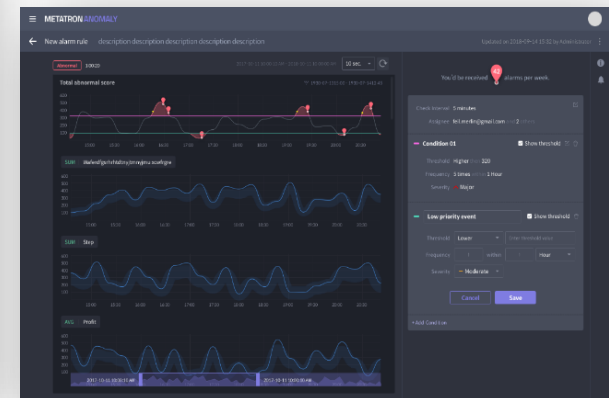
Alarm details & report



Alarm rule creation



Alarm rule creation



Alarm rule details

# END

[metatron@sk.com](mailto:metatron@sk.com)