

OASIS – DATA ANALYSIS PLATFORM FOR ENTERPRISE

KEIJI YOSHIDA – DATA ENGINEER, LINE CORPORATION

INTRODUCTION

- We have created a web-based data analysis platform named "OASIS"
- Employees can analyze data of a Hadoop cluster by writing Spark applications
- 100+ employees use it every day



Agenda

1. Motivation
2. Features & System Architecture
3. Use Cases

Agenda

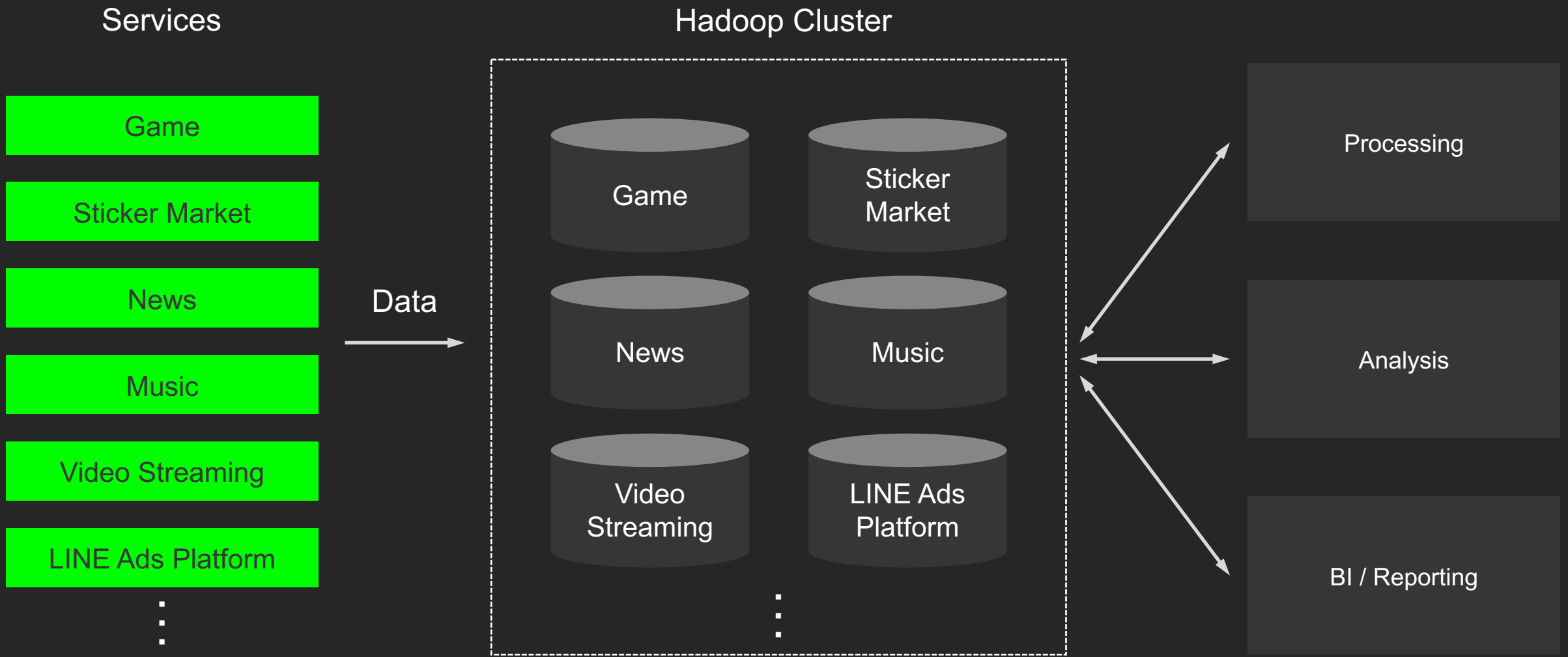
1. Motivation
2. Features & System Architecture
3. Use Cases

LINE CORPORATION

- Based in Tokyo, Japan
- Provides a messaging application named “LINE”
- 164M monthly active users across Japan, Thailand, Taiwan, and Indonesia
- Also provides various related services such as games, sticker market, etc.

The word "LINE" is written in a bold, green, sans-serif font.

DATA PLATFORM AT LINE



PUBLICATION OF HADOOP CLUSTER

- Enable all employees analyze their services' data as they like
- Speed up their data analysis process and decision making

REQUIREMENTS

1. Security

- Each employee can access only the data related to their service

2. Stability

- Queries must not affect the performance of other queries

3. Features

- Each employee can extract data from the Hadoop cluster as they like
- Results can be visualized and shared within a team or a department

SOLUTIONS

1. Security

- Kerberize the Hadoop cluster and install Apache Ranger

2. Stability

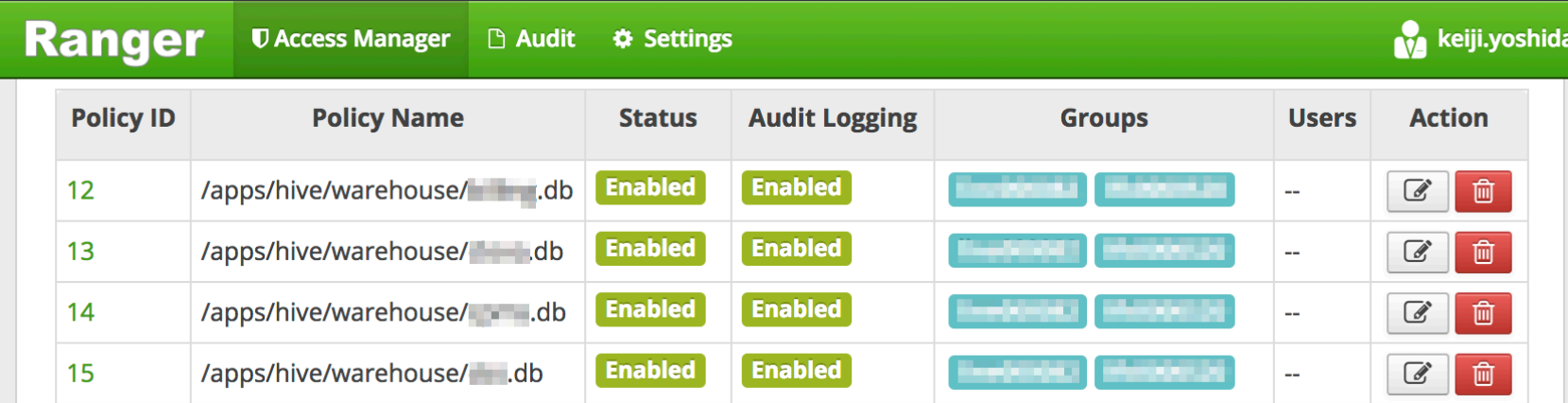
- Use Apache Spark as a query and application execution engine

3. Features

- Try Apache Zeppelin for the Web UI

APACHE RANGER

- Framework to manage access control over a Hadoop cluster
- Used to control each employee's data access



The screenshot shows the Apache Ranger web interface. The top navigation bar is green and contains the 'Ranger' logo, 'Access Manager', 'Audit', and 'Settings' menus. The user 'keiji.yoshida' is logged in. Below the navigation bar is a table with the following columns: Policy ID, Policy Name, Status, Audit Logging, Groups, Users, and Action. The table contains four rows of policy data, all with a status of 'Enabled' and 'Enabled' for audit logging.

Policy ID	Policy Name	Status	Audit Logging	Groups	Users	Action
12	/apps/hive/warehouse/...db	Enabled	Enabled	View Edit	--	Edit Delete
13	/apps/hive/warehouse/...db	Enabled	Enabled	View Edit	--	Edit Delete
14	/apps/hive/warehouse/...db	Enabled	Enabled	View Edit	--	Edit Delete
15	/apps/hive/warehouse/...db	Enabled	Enabled	View Edit	--	Edit Delete

SOLUTIONS

1. Security

- Kerberize the Hadoop cluster and install Apache Ranger

2. Stability

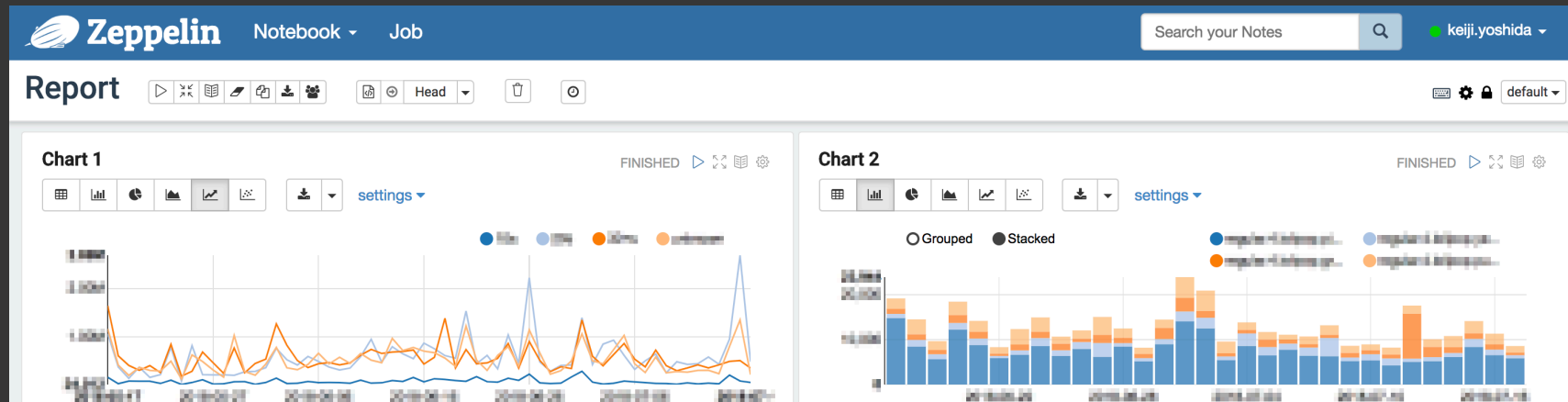
- Use Apache Spark as a query and application execution engine

3. Features

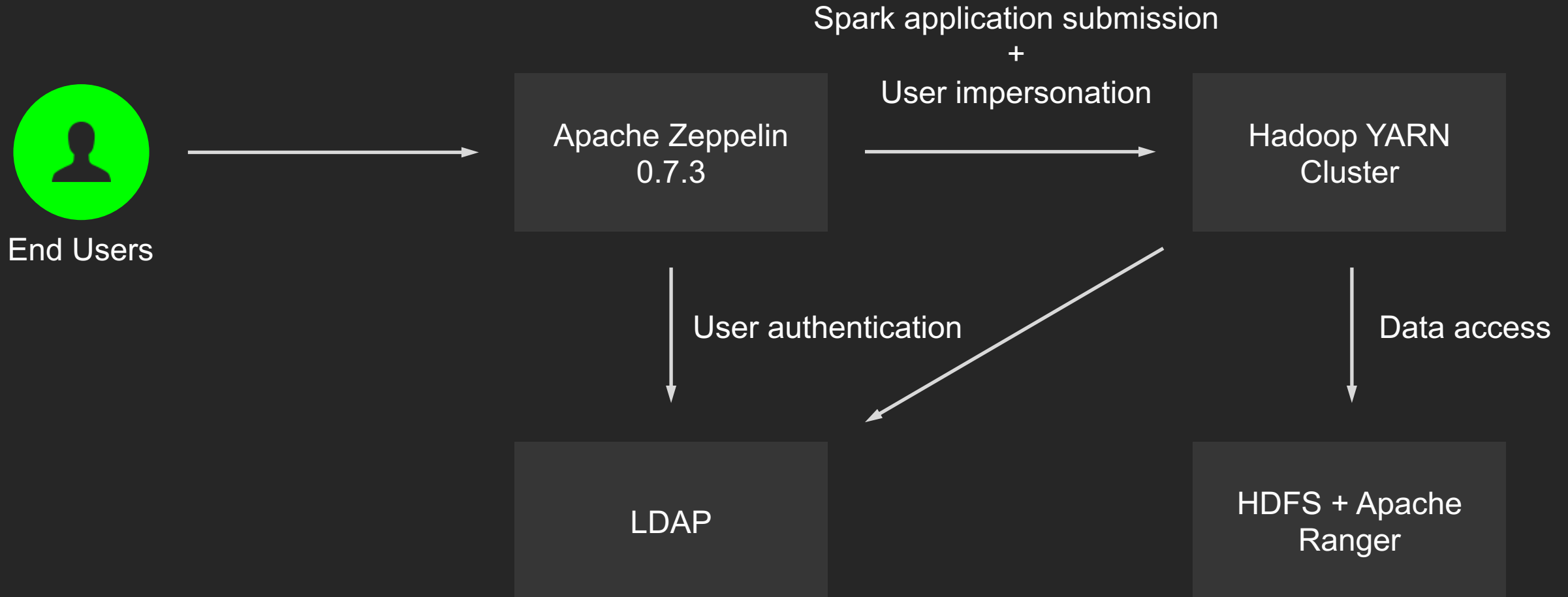
- Try Apache Zeppelin for the Web UI

APACHE ZEPPELIN

- Web-based data analysis tool
- Supports Apache Spark and user impersonation
- Results can be shared within multiple users as a form of a notebook



SYSTEM ARCHITECTURE



ISSUES OF APACHE ZEPPELIN 0.7.3

1. Security

- An arbitrary user can be set to scheduled notebook's execution user
- Users can access the data which they don't have access rights to

ISSUES OF APACHE ZEPPELIN 0.7.3

2. Stability

- Runs only on a single server
- Does not support the “yarn-cluster” mode of Apache Spark
- Freezes when a Spark driver program consumes many server resources

ISSUES OF APACHE ZEPPELIN 0.7.3

3. Features

- Users have to set access control to each notebook
- Users cannot execute a notebook while changing only its parameters without saving it

OASIS

Data Labs / user / keiji.yoshida

OASIS Report Sample

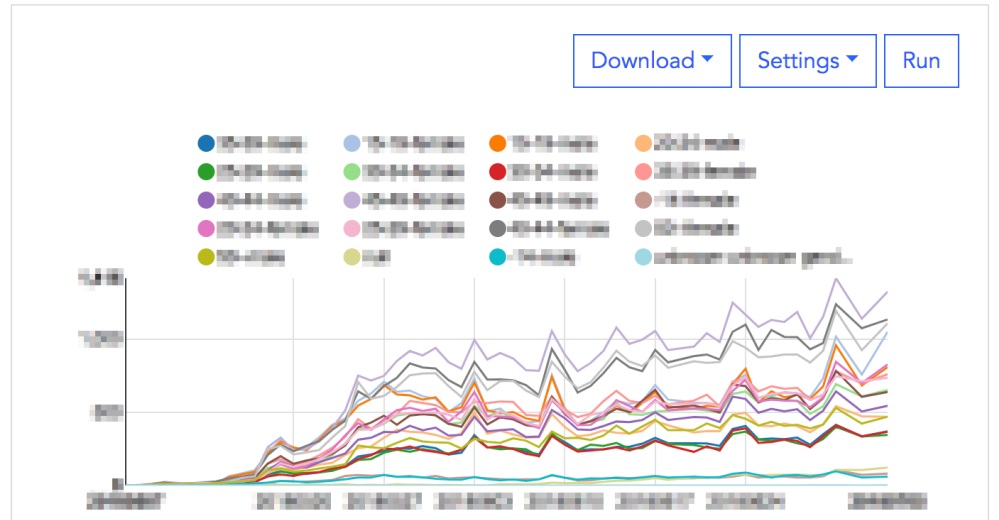
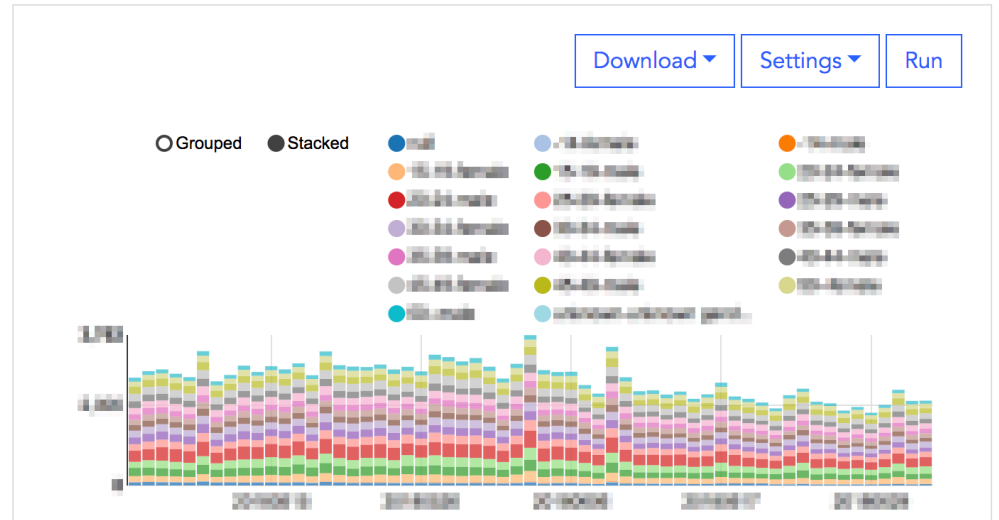
Edit

Space Public

keiji.yoshida, 1 second ago keiji.yoshida 30 11 ** ?

2 paragraphs

Run All Paragraphs



COMPARISON

Apache Zeppelin

For a single team

- Does not go with Apache Ranger
- Runs only on a single server
- Access control per notebook



OASIS

For an enterprise

- Works well with Apache Ranger
- Runs on multiple servers
- Access control per team



Agenda

1. Motivation
2. Features & System Architecture
3. Use Cases

OVERVIEW

- User impersonation
- Data Visualization
- Notebooks Sharing
- Scalable

TOP PAGE

OASIS Interactive data analytics tool

-  [Billing](#) (1)
-  [Data Labs](#) (8)
-  [LINE Ads Platform](#) (6)
-  [LINE BAITO](#) (4)
-  [LINE BLOG](#) (1)
-  [LINE Delima](#) (1)
-  [LINE Fortune](#) (1)
-  [LINE Game](#) (1)
-  [LINE LIVE](#) (9)
-  [LINE LIVE \(DE\)](#) (1)
-  [LINE MANGA](#) (3)
-  [LINE MOBILE](#) (8)
-  [LINE Music](#) (1)
-  [LINE NEWS](#) (3)
-  [LINE Pay](#) (2)
-  [LINE PORTAL SEARCH](#) (1)
-  [LINE Shopping \(JP\)](#) (1)

SPACE

- A root directory of notebooks for a team, a department, or a service
- Access right for each user is separately set in each space
- 2 types of access rights: "read write" and "read only"

Space 1



User A (read write)



User B (read only)

Space 2



User C (read write)



User D (read only)

NOTEBOOK CREATION

OASIS
1 interpreter keiji.yoshida

Data Labs / user / keiji.yoshida

Sample Notebook

Schedule @ 0 11 ** ?

JST (GMT+09:00)

Exp. @ 2018/08/24

Access Control @ Space Public

Executor @ keiji.yoshida

Save
Cancel

No parameters

1 paragraph
Run All Paragraphs
Add Paragraph

Sample Paragraph ⚙

```

1 %sparksql
2 select yyyy, type, degree_celsius from datalake_dev.tokyo_temperature
        
```

Download
Run

Table
 Bar Chart
 Line Chart

X-Axis

yyyy

X-Type

Time

X-Format @

%Y/%m

Y-Axis

degree_celsius

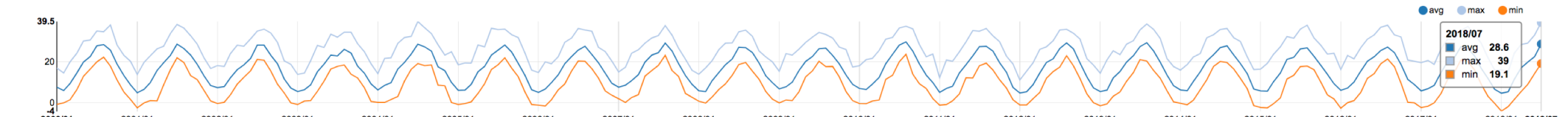
Y-Aggr

Sum

Group

type

Height



SPARK APPLICATION

- A single Spark application launches per notebook session
- Notebook author's account is used to access files
- Spark, Spark SQL, PySpark, and SparkR are available
- Each language of a single notebook session shares a single Spark application

SPARK APPLICATION

1. Spark

```

1 %spark
2 val r = new scala.util.Random(100)
3 val x = for (i <- 1 to 100) yield r.nextInt(100)
4 val y = for (i <- 1 to 100) yield r.nextInt(100)
5 val x_df = sc.parallelize(x).toDF("v")
6 val y_df = sc.parallelize(y).toDF("v")
7 x_df.createOrReplaceTempView("temp_x")
8 y_df.createOrReplaceTempView("temp_y")

```

Run

2 secs / keiji.yoshida / 2018-07-24 13:41:57

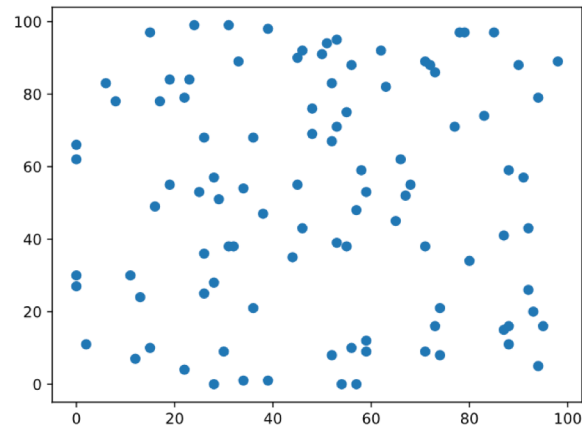
2. PySpark

```

1 %pyspark
2 import matplotlib.pyplot as plt
3 import sys
4 x = spark.sql('select v from temp_x').collect()
5 y = spark.sql('select v from temp_y').collect()
6 plt.plot([v.v for v in x], [v.v for v in y], 'o')
7 plt.savefig(sys.stdout, format='svg')

```

Run



3. Spark SQL

```

1 %sparksql
2 select v from temp_x

```

Download ▾

Run

Table Bar Chart Line Chart

Height

224

v
15
50
74
88
91
66
36

UTILIZATION OF IN-MEMORY CACHING

Spark SQL 1

```
1 %sparksql cacheTable=foo
2 select yyyy, type, degree_celsius
3 from datalake_dev.tokyo_temperature
4
```

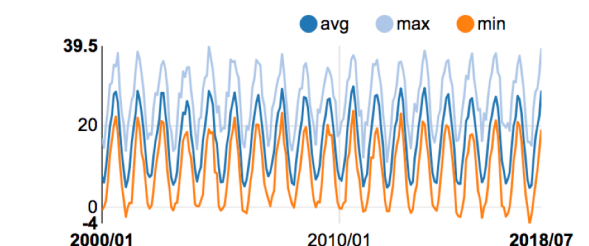
Download Run

Table Bar Chart Line Chart

X-Axis: yyyy X-Type: Time X-Format: %Y/%m

Y-Axis: degree_ce Y-Aggr: Sum Group: type

Height: 200



Spark SQL 2

```
1 %sparksql
2 select yyyy, type, degree_celsius
3 from foo
4 where yyyy between '2017/01' and '2017/12'
```

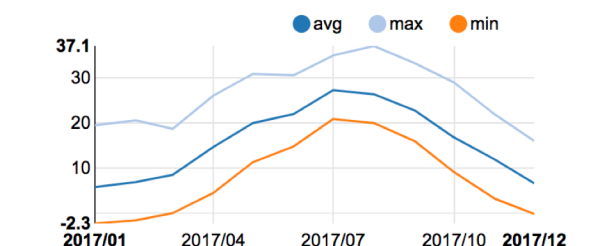
Download Run

Table Bar Chart Line Chart

X-Axis: yyyy X-Type: Time X-Format: %Y/%m

Y-Axis: degree_ce Y-Aggr: Sum Group: type

Height: 200



Spark SQL 3

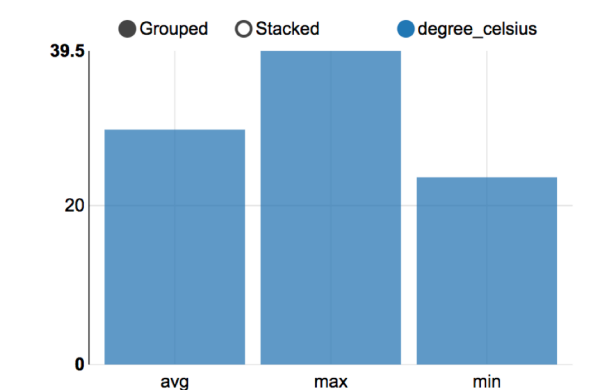
```
1 %sparksql
2 select yyyy, type, degree_celsius
3 from foo
4
```

Download Run

Table Bar Chart Line Chart

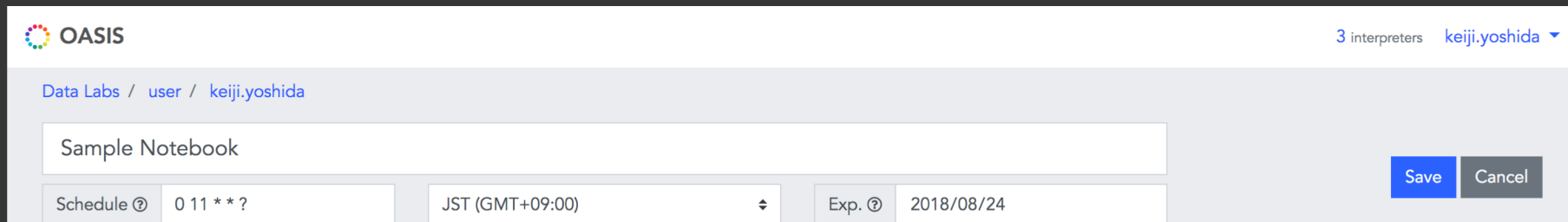
X-Axis: type Y-Axis: degree_ce Y-Aggr: Max

Group: Height: 292



SCHEDULING

- Notebooks can be executed automatically on a prescribed schedule
- Contents of notebooks can be kept updated by this feature
- This feature is also used for creating a light weight ETL processing



The screenshot shows the OASIS scheduling interface. At the top left is the OASIS logo. On the top right, it displays '3 interpreters' and the user 'keiji.yoshida'. Below the header, there is a breadcrumb trail: 'Data Labs / user / keiji.yoshida'. The main form contains a text input field with the value 'Sample Notebook'. Below this, there are three input fields: 'Schedule' with the value '0 11 **?', a dropdown menu showing 'JST (GMT+09:00)', and 'Exp.' with the value '2018/08/24'. To the right of these fields are two buttons: 'Save' (blue) and 'Cancel' (grey).

PARAMETERS

- Parameters can be injected into a notebook during its execution
- “read only” users can execute a notebook while changing its parameters

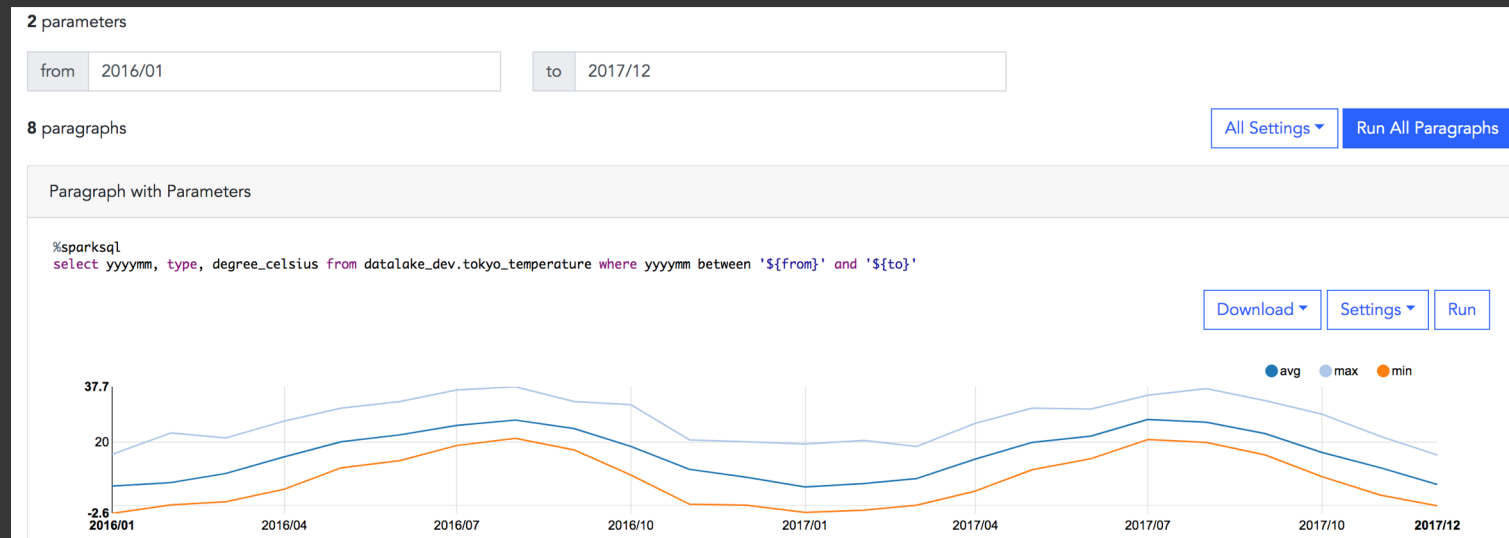


TABLE DEFINITIONS

OASIS
1 interpreter keiji.yoshida

Tables
Upload
✕

DB	filter...
adobios_dev	
analysis	
asia_suggest	
billing	
charmy	
class	
cpna	
customer_market	
datastream	
dataflow	
datahub	
datalake	
datalake_dev	
dataopen	

Table	filter...
lineitems_user_join_view	
lineitemings_order_status	
mileage_user_book	
ml_sns_output	
previous_membership_daily	
previous_type	
price_moder	
sales_history	
shop	
table_history_subscription	
table_user_data	
table_user_gettableid	
test	
tokyo_temperature	

Column	Type
yyyyymm	varchar
type	varchar
degree_celsius	decimal(3,1)

FILE UPLOAD

OASIS 3 interpreters keiji.yoshida ▾

Data Labs / use

Sample No

Space Public

keiji.yoshida, 2

2 parameters

from 2016/01

8 paragraphs

Paragraph with 1

37.7

20

Tables **Upload** ✕

DB datalake_dev Table tokyo_temperature

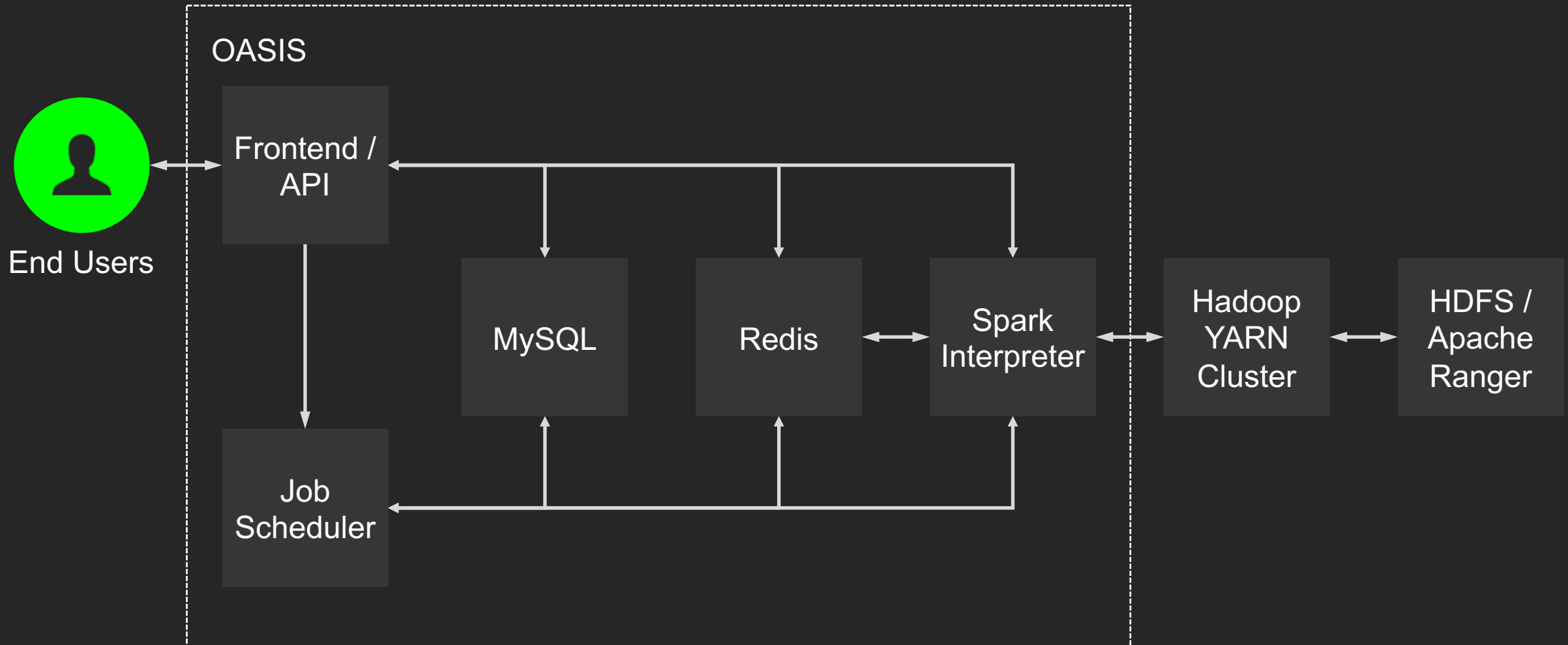
Use the first row as a header
 Drop the same name table if it exists

yyyyymm	type	degree_celsius
string	string	decimal
(Precision)	(Precision)	(3,1)
2000/01	avg	7.6
2000/02	avg	6
2000/03	avg	9.4
2000/04	avg	14.5
2000/05	avg	19.8

Top 5 rows are shown.

Load Cancel

SYSTEM ARCHITECTURE



HADOOP CLUSTER

- 500 DataNodes / NodeManagers
- HDFS usage: 20PB
- 150+ Hive databases
- 1,500+ Hive tables

Agenda

1. Motivation
2. Features & System Architecture
3. Use Cases

STATS

- 1,500+ users
 - 100+ daily active users
 - 300+ monthly active users
- 30+ spaces (i.e. departments, teams, or services)
- 1,100+ notebooks
 - 200+ scheduled notebooks

USE CASES

1. Report
2. Interactive dashboard
3. ETL
4. Monitoring
5. Ad hoc analysis

1. REPORT

OASIS
6 interpreters keiji.yoshida

99_user / keiji.yoshida

Report
 Execution Schedule
 Time Zone: JST
 Expiration Date: 2018-08-26
 Next: 2018-07-27 11:00:00

Edit

Space Public

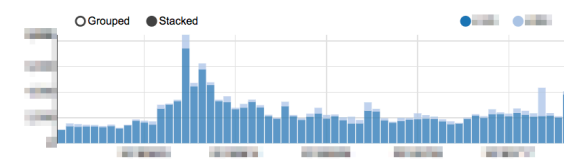
keiji.yoshida, 1 second ago keiji.yoshida 0 11 ** ?

4 paragraphs

All Settings Run All Paragraphs

パラグラフ 1 (2018-07-25 12:12:08) 実行ログ (CSV)

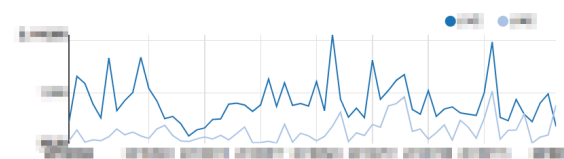
Download Settings Run



14 secs / 124 rows / keiji.yoshida / 2018-07-25 12:12:08

パラグラフ 2 (2018-07-25 12:12:37) 実行ログ (CSV)

Download Settings Run



13 secs / 124 rows / keiji.yoshida / 2018-07-25 12:12:37

パラグラフ 3 (2018-07-25 12:14:09) 実行ログ (CSV)

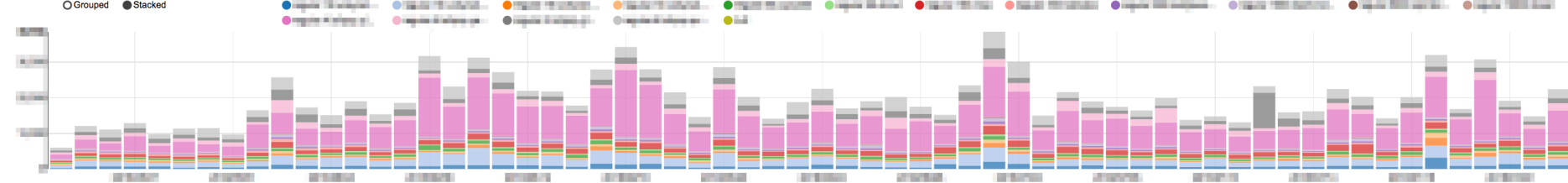
Download Settings Run

date	実行ログ (CSV)
20180701	実行ログ (CSV)
20180702	実行ログ (CSV)
20180703	実行ログ (CSV)
20180704	実行ログ (CSV)
20180705	実行ログ (CSV)
20180706	実行ログ (CSV)
20180707	実行ログ (CSV)

3 secs / 24 rows / keiji.yoshida / 2018-07-25 12:14:09


ギフト回数を by item

Download Settings Run



12 secs / 993 rows / keiji.yoshida / 2018-07-25 12:14:27

2. INTERACTIVE DASHBOARD



 OASIS
3 interpreters [keiji.yoshida](#) ▾

[HOME](#) / [99_user](#) / [keiji.yoshida](#)

Dashboard

Edit ▾

Space Public

 keiji.yoshida, 9 seconds ago  keiji.yoshida

2 parameters

from 20180701

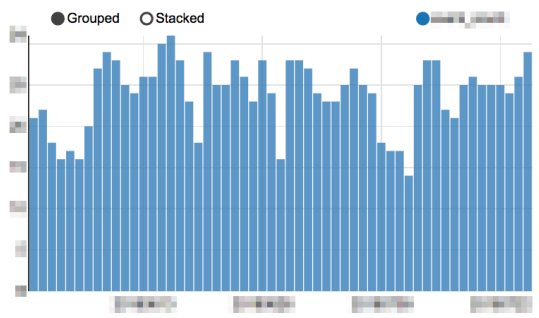
to 20180731

3 paragraphs

All Settings ▾
Run All Paragraphs

Paragraph 1

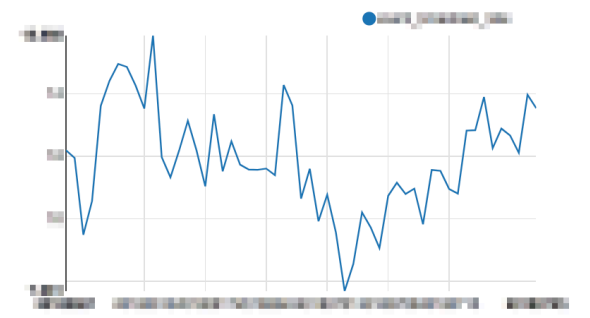
Download ▾
Settings ▾
Run



5 secs / 55 rows / keiji.yoshida / 2018-07-26 12:51:59

Paragraph 2

Download ▾
Settings ▾
Run



5 secs / 55 rows / keiji.yoshida / 2018-07-26 12:52:05


Paragraph 3

Download ▾
Settings ▾
Run

main_id	main_breadcrumb_cnt	main_download_cnt	gifted_breadcrumb
DOMAIN01	47	21	
DOMAIN01	46	22	
DOMAIN01	136	26	
DOMAIN01	137	26	
DOMAIN01	87	26	
DOMAIN01	29	13	
DOMAIN01	46	26	
DOMAIN01	24	14	
DOMAIN01	6	1	
DOMAIN01	26	26	
DOMAIN01	22	11	

45 secs / 1,208 rows / keiji.yoshida / 2018-07-26 12:52:57

3. ETL

 OASIS 2 interpreters keiji.yoshida ▾

[Data Labs](#) / [user](#) / [keiji.yoshida](#)

ETL

Space Public

keiji.yoshida, 40 seconds ago • keiji.yoshida • 0 10 ** ?

Execution Schedule
Time Zone: JST
Next: 2018-07-27 10:00:00

Edit ▾

1 paragraph

All Settings ▾ Run All Paragraphs

ETL

```
%pyspark
query="""
select
  some_transform(col1)
, another_transform(col2)
, dt
from
  database.table_src
where
  dt = date_format(date_add(now(), -1), 'yyyyMMdd')
"""
spark.sql(query).repartition(1).coalesce(1).write.insertInto('database.table_dst', True)
```

Settings ▾ Run

269 secs / keiji.yoshida / 2018-07-26 10:15:41

4. MONITORING

OASIS
3 interpreters keiji.yoshida ▾

Data Labs / monitoring

Hourly Log Monitoring

Space Public

keiji.yoshida, 1 second ago ● keiji.yoshida ⌚ 10 ***?

Execution Schedule
Time Zone: JST
Next: 2018-07-26 15:10:00

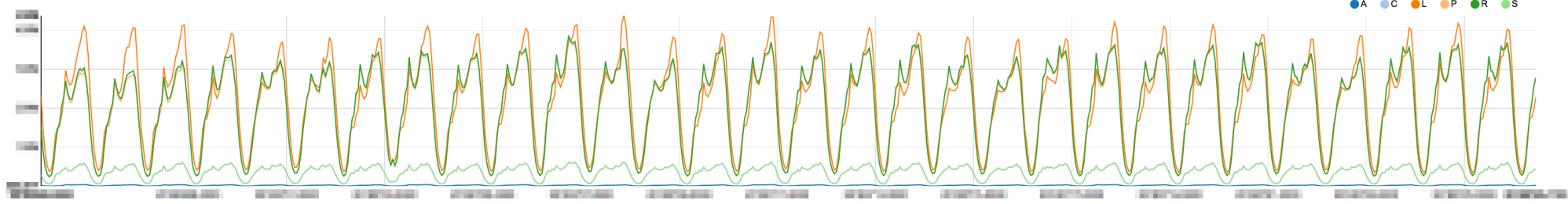
Edit ▾

All Settings ▾
Run All Paragraphs

2 paragraphs

Hourly Log Numbers

Download ▾
Settings ▾
Run



3 secs / 4,392 rows / keiji.yoshida / 2018-07-26 14:11:27

Send an alert to Slack when a change rate exceeds the threshold (30%)

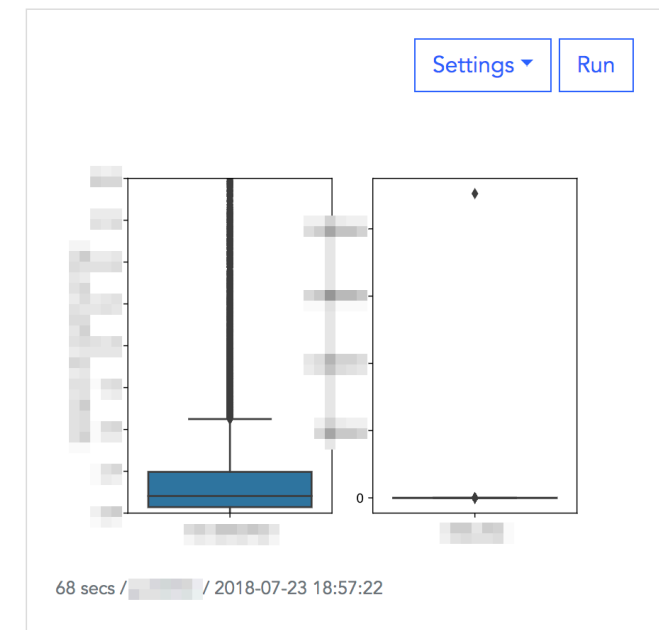
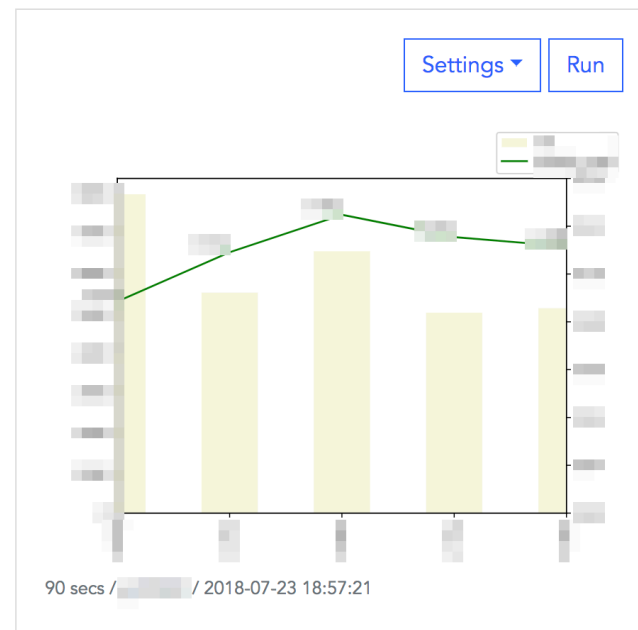
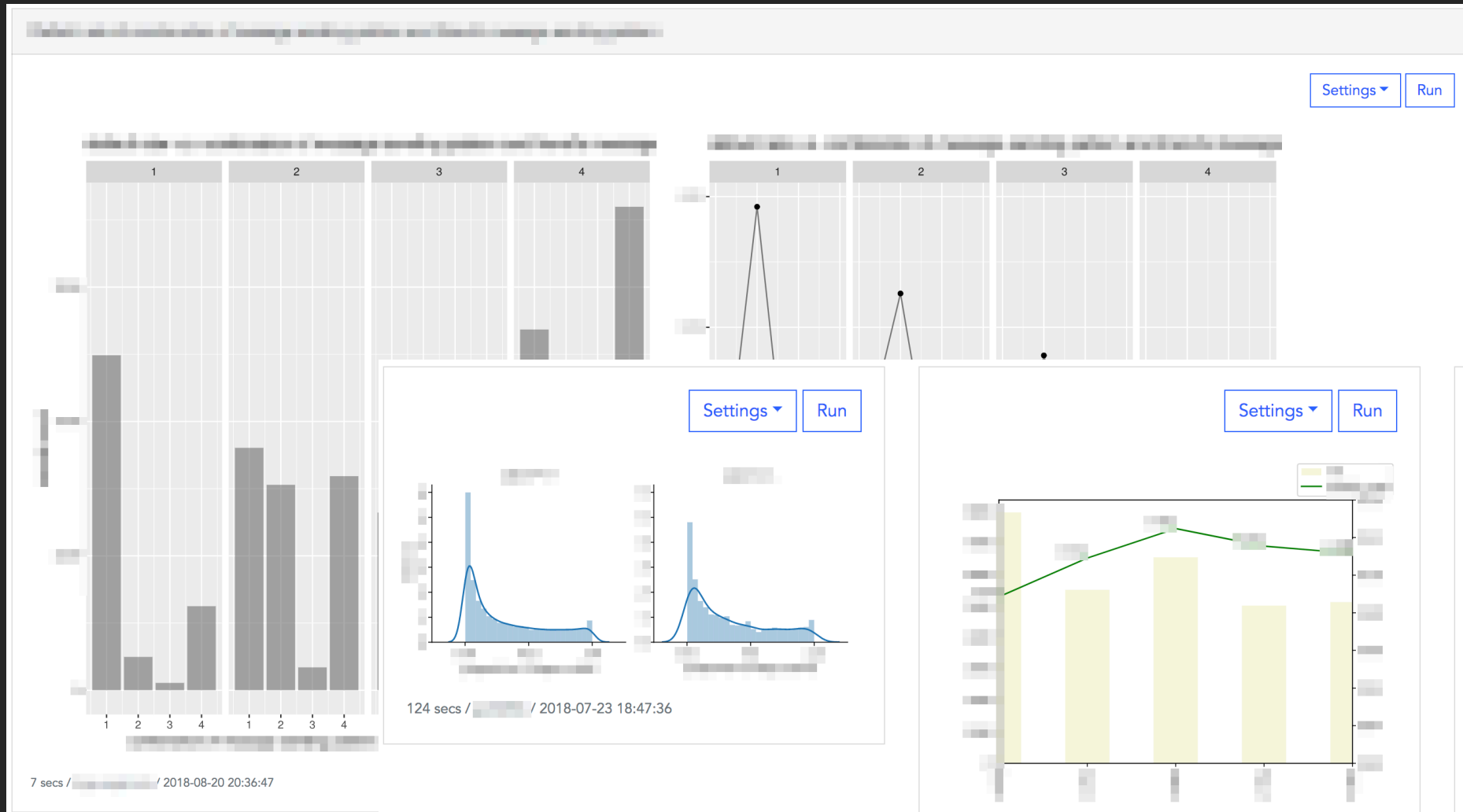
Settings ▾
Run

```

+-----+
| dt|type|cnt|cnt_last_week|ratio|threshold|
+-----+
+-----+
+-----+
+-----+
                    
```

40 secs / keiji.yoshida / 2018-07-26 14:12:16

5. AD HOC ANALYSIS



RECAP

- We created OASIS to solve the issues of Apache Zeppelin
- Extracted data can be visualized and shared within a team
- OASIS utilizes the user impersonation feature of Apache Spark
- At LINE, OASIS is used for reporting, data monitoring, ad hoc analysis, etc.

THANK YOU