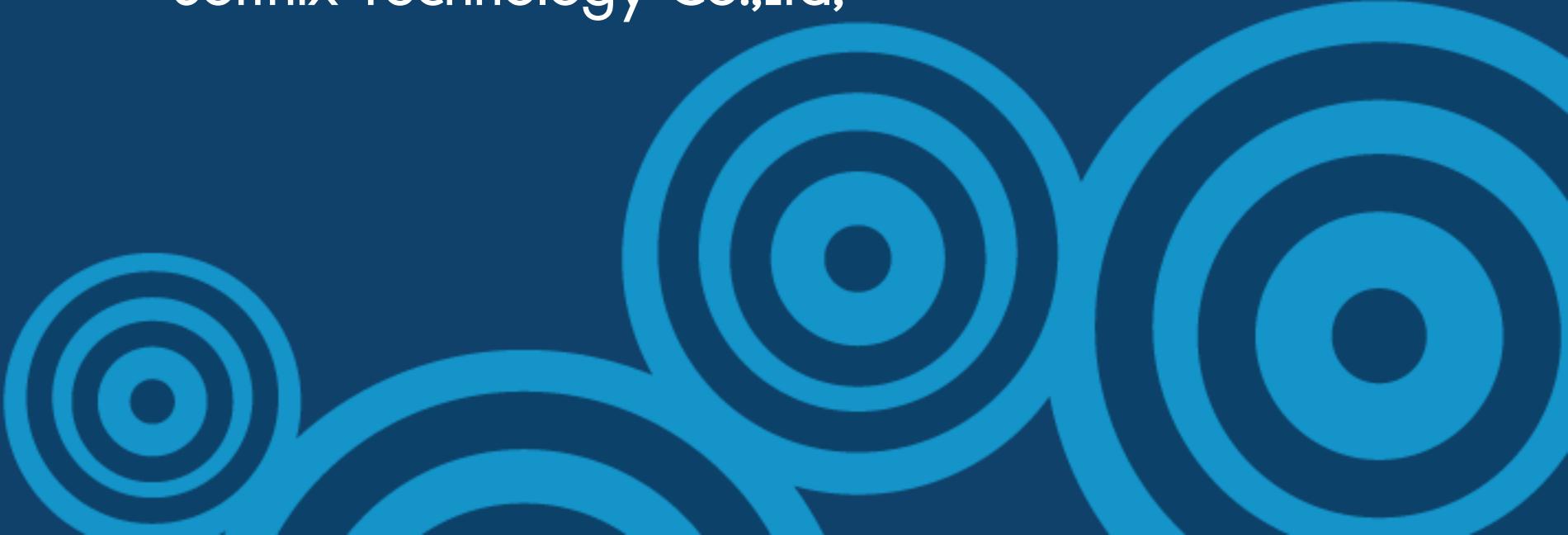




Big Data Administration Training

Softnix Technology Co.,Ltd,





Course Outline



Day 1

- Big Data Fundamental
- Softnix Data Platform
- Cloudera
- HDFS , Hive , Presto

Day 2

- Python Basic
- Web Scraping

Day 3

- Apache Airflow + Work Shop
- Introduction to Graph Data Base

Day 4

- Machine Learning

Day 5

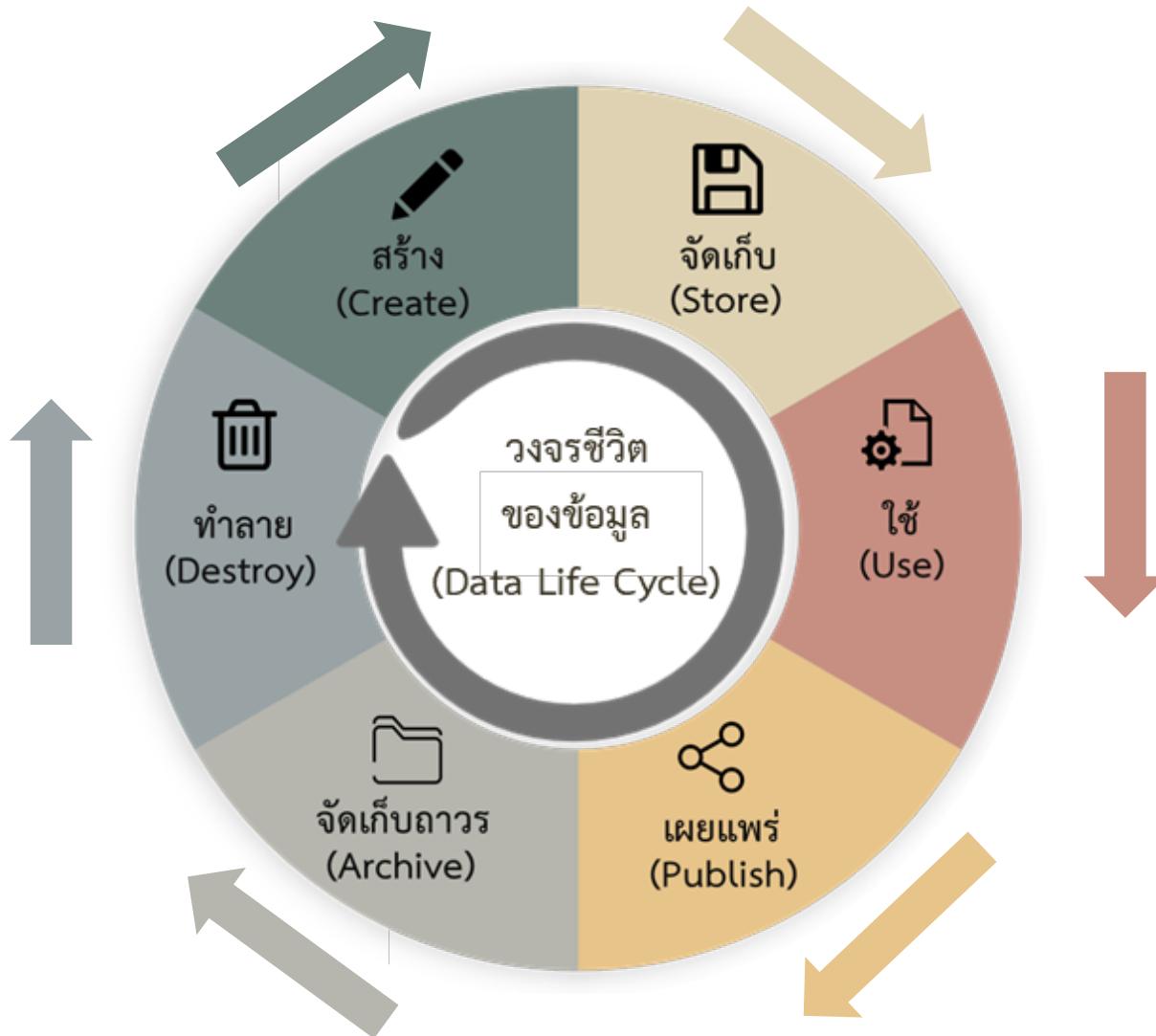
- Use Case , LAB

The six Vs of big data

Big data is a collection of data from various sources, often characterized by what's become known as the 3Vs: *volume, variety and velocity*. Over time, other Vs have been added to descriptions of big data:

VOLUME	VARIETY	VELOCITY	VERACITY	VALUE	VARIABILITY
The amount of data from myriad sources. 	The types of data: structured, semi-structured, unstructured. 	The speed at which big data is generated. 	The degree to which big data can be trusted. 	The business value of the data collected. 	The ways in which the big data can be used and formatted. 

วงจรชีวิตข้อมูล



วงจรชีวิตและองค์ประกอบของภารกิจข้อมูล



แนะนำธรรมาภิบาลข้อมูล



ด้านโครงสร้าง

คณะกรรมการธรรมาภิบาลข้อมูล (Data Governance Council)

- กลุ่มบุคคลที่มาจากผู้บริหารระดับสูงขององค์กร ทั้งด้านธุรกิจและไอที
- มีหน้าที่ในการกำหนดความต้องการ ให้ข้อเสนอแนะ และอนุมัตินโยบายข้อมูล เกณฑ์การวัดคุณภาพ ระเบียบ และข้อบังคับอื่น ๆ ที่เกี่ยวข้องกับข้อมูล รวมไปถึงการจัดลำดับความสำคัญของข้อมูลในการกำกับดูแล

ทีมบริกรข้อมูล (Data Steward Team)

- บุคคลที่ทำหน้าที่รับผิดชอบในการ
 - **นิยามเมทาดาta**
 - **นิยามความต้องการด้านคุณภาพ และความมั่นคงปลอดภัย**
 - **ร่างนโยบายและกระบวนการเกี่ยวกับธรรมาภิบาลข้อมูลและการบริหารจัดการข้อมูล**
 - **ตรวจสอบความสอดคล้องกันระหว่างนโยบายกับการดำเนินการต่อข้อมูล**
 - **ตรวจสอบคุณภาพข้อมูล**
 - **วิเคราะห์ผลจากการตรวจสอบ**
 - **รายงานผลลัพธ์ไปยังคณะกรรมการธรรมาภิบาลข้อมูลและผู้ที่เกี่ยวข้องอื่น ๆ**

ผู้มีส่วนได้เสียกับข้อมูล (Data Stakeholder)

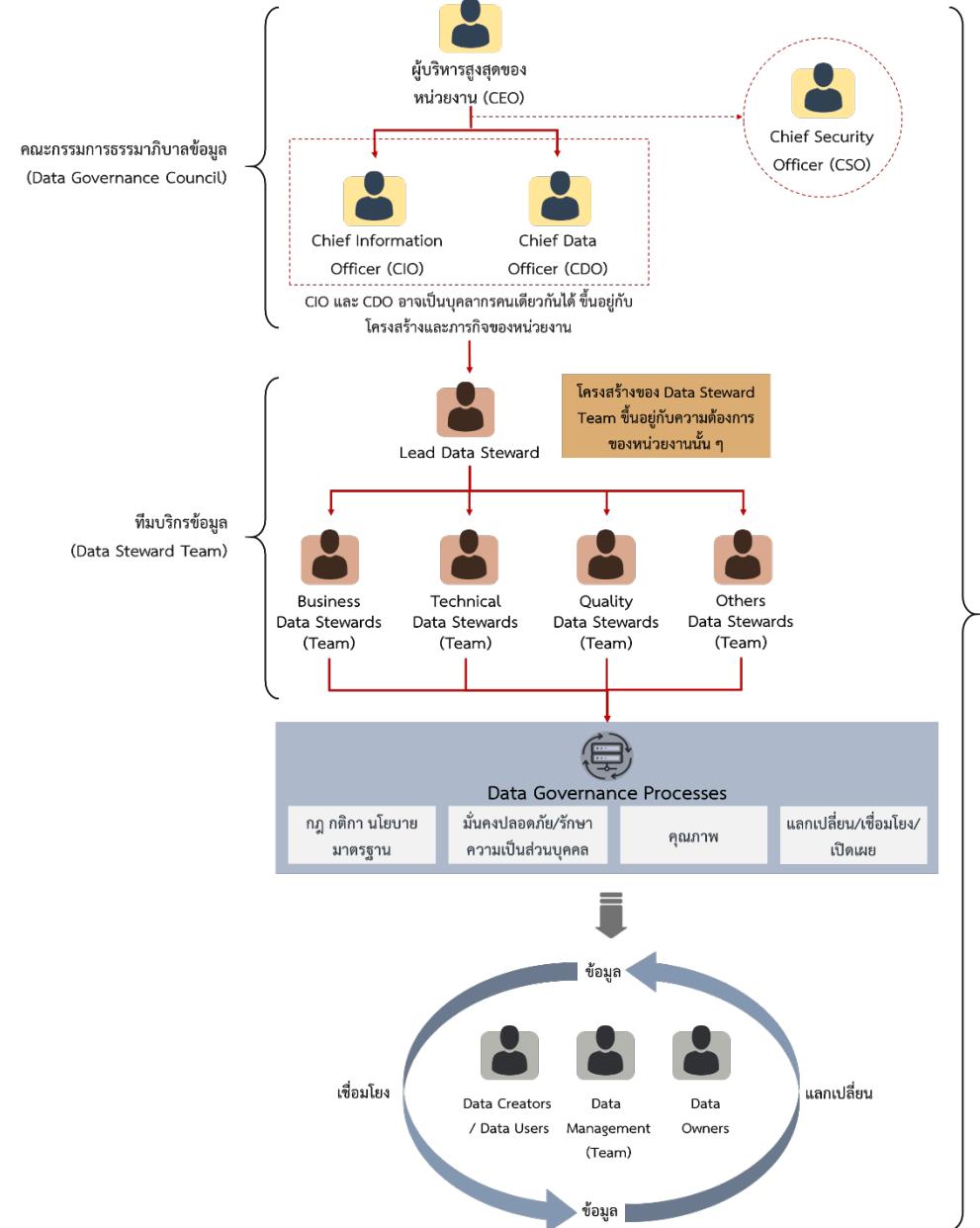
- บุคคลหรือกลุ่มบุคคลทั้งหมดที่เกี่ยวข้องกับข้อมูล
 - **ผู้บริหารข้อมูลระดับสูง (Chief Data Officer)**
 - **คณะกรรมการธรรมาภิบาลข้อมูล (Data Governance Council)**
 - **สำนักงานธรรมาภิบาลข้อมูล (Data Governance Office)**
 - **บริกรข้อมูล (Data Steward)**
 - **ผู้ดูแลข้อมูลด้านเทคนิค (Data Custodian)**
 - **ผู้สร้างข้อมูล (Data Creator)**
 - **ผู้ใช้ข้อมูล (Data User)**

ด้านโครงสร้าง

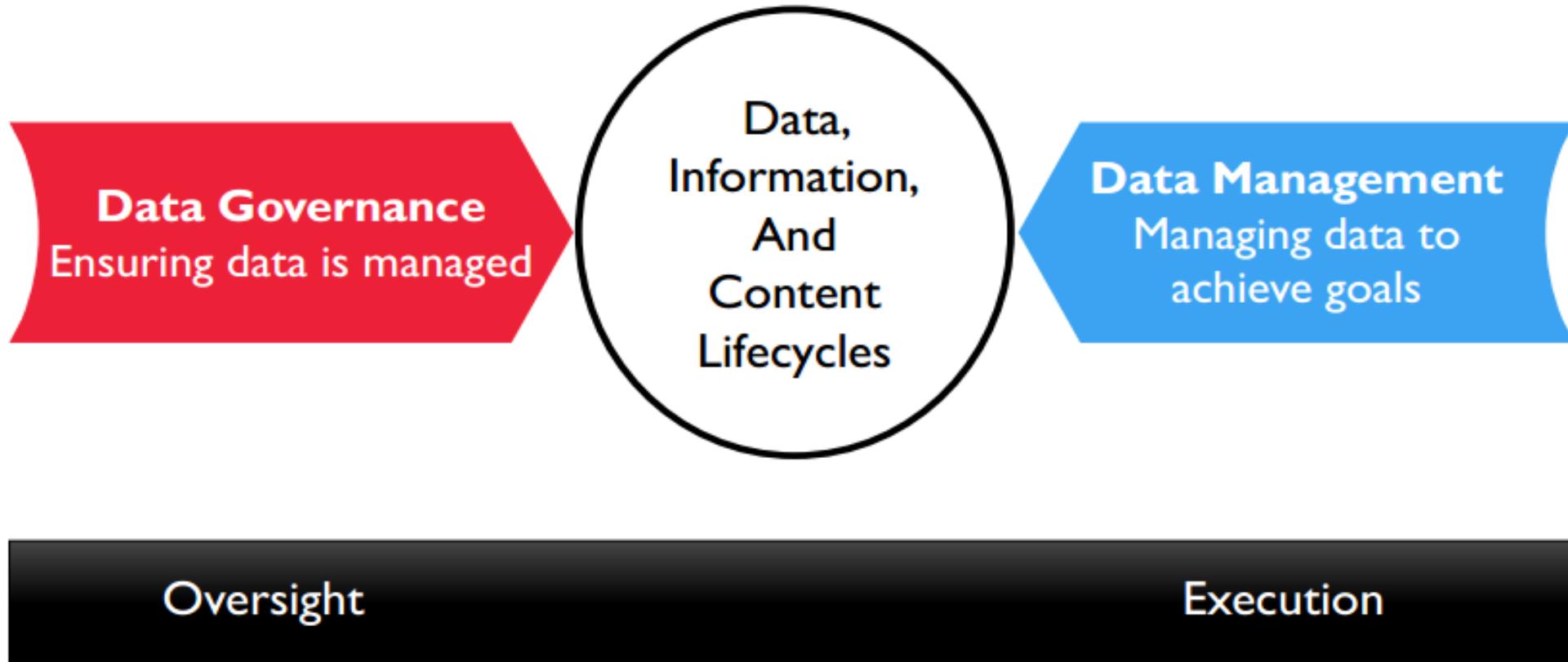
ห่วงงานสามารถจัดตั้งส่วนงานธรรมาภิบาลข้อมูลในรูปแบบที่แตกต่างกัน เช่น รูปแบบทีมเสมือน (Virtual Team) ที่คัดเลือกมาจากส่วนงานต่าง ๆ

ตัวอย่างโครงสร้างธรรมาภิบาลข้อมูลแบ่งออกเป็น

- คณะกรรมการธรรมาภิบาลข้อมูล (Data Governance Council)
- ทีมบริกรข้อมูล (Data Steward Team)
- ผู้มีส่วนได้เสียกับข้อมูล (Data Stakeholder)



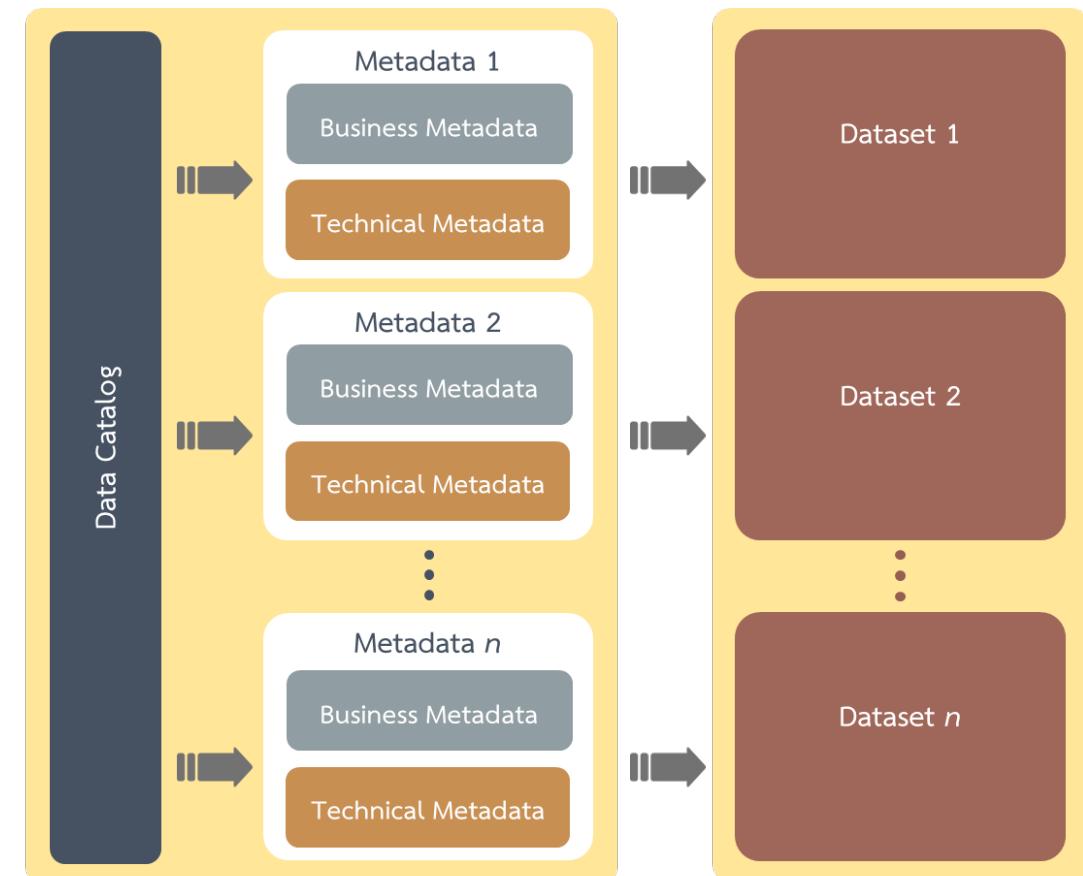
ความสัมพันธ์ระหว่างธรรมาภิบาลข้อมูลและการบริหารจัดการข้อมูล



ด้านนิยามและกฎเกณฑ์ - การนิยามข้อมูล



หมวดหมู่ของข้อมูล



Metadata Repository / Data Dictionary

ความสัมพันธ์ระหว่างบัญชีข้อมูล เมทาดาต้า และชุดข้อมูล

ข้อมูล (Data)

ข้อมูลที่มีโครงสร้าง (Structured Data)

เป็นข้อมูลที่มีการนิยามโครงสร้างของข้อมูลไว้ ซึ่งเป็นโครงสร้างที่ทำให้ง่ายต่อการค้นหา เช่น

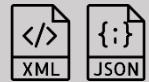
- ฐานข้อมูล
- Comma-Separated Values – CSV



ข้อมูลกึ่งโครงสร้าง (Semi-structured Data)

เป็นข้อมูลที่มีการนิยามโครงสร้างของข้อมูลไว้ แต่มีโครงสร้างเป็นแบบลำดับชั้น (Hierarchy) เช่น

- Extensible Markup Language – XML
- JavaScript Object Notation - JSON



ข้อมูลที่ไม่มีโครงสร้าง (Unstructured Data)

เป็นข้อมูลที่ไม่ได้มีการนิยามโครงสร้างของข้อมูลไว้ มักจะอยู่ในรูปแบบ เช่น

- กระดาษ
- ข้อความ
- รูปภาพ
- เสียง
- วิดีโอ



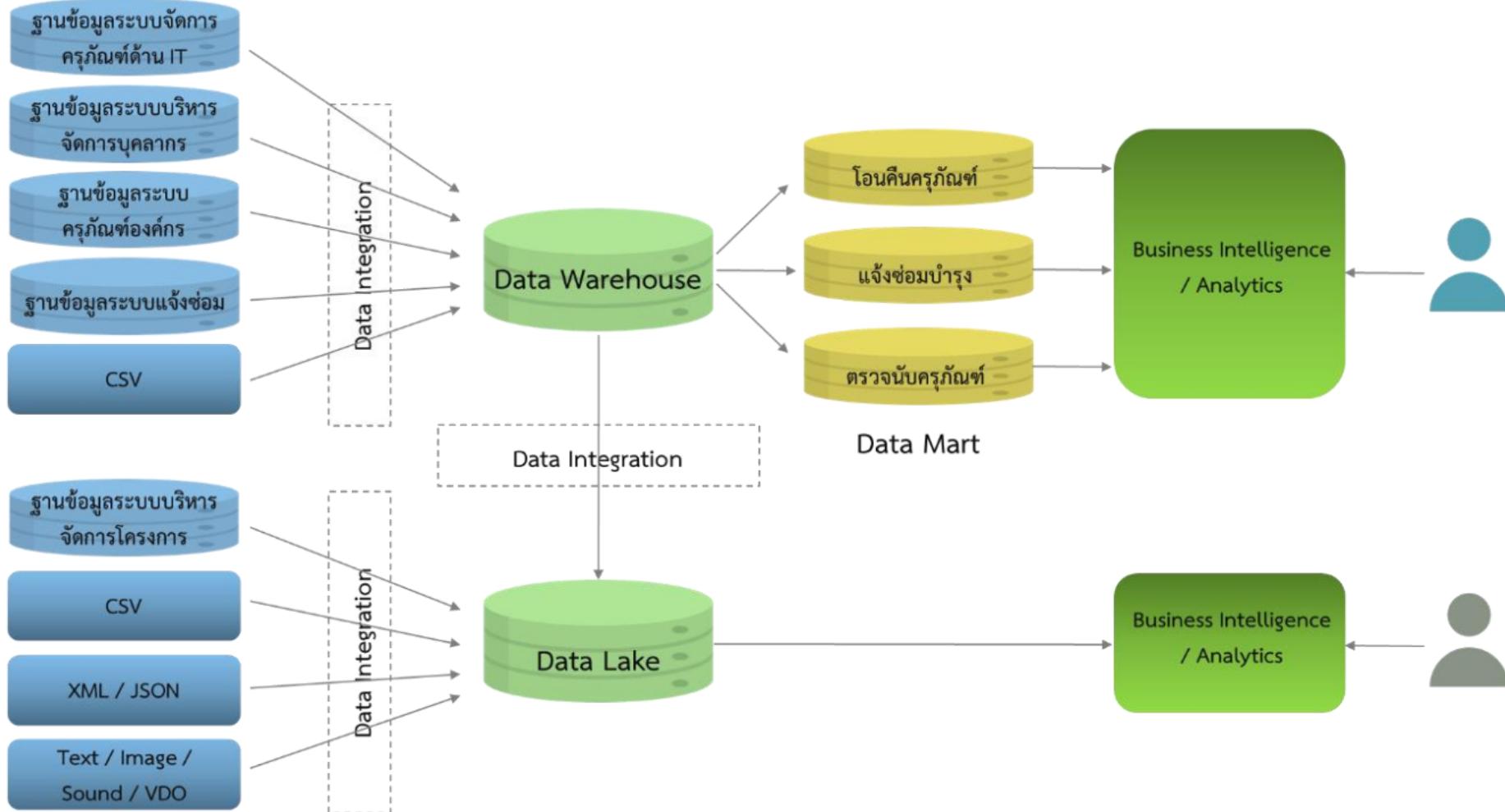
คลังข้อมูล ทะเลสาบข้อมูล ระบบรายงานอัจฉริยะ และดาตากอนาไลติกส์ (Data Warehouse, Data Lake, Business Intelligence, and Data Analytics)

• คลังข้อมูล (Data Warehouse)

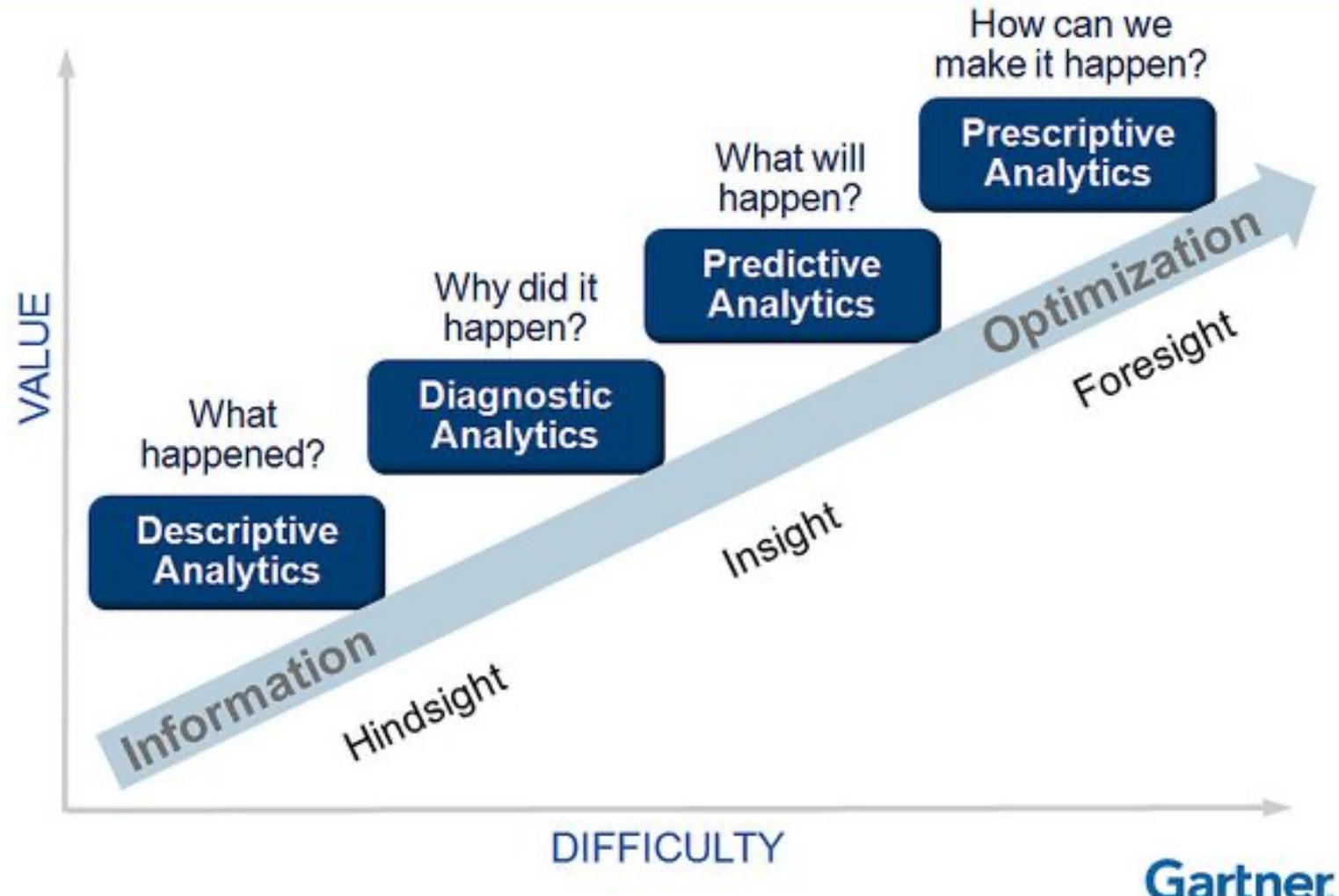
- เป็นข้อมูลที่ได้จากการบูรณาการข้อมูล (Data Integration) ซึ่งเกิดจากการรวมข้อมูลจากแหล่งข้อมูลต่าง ๆ ที่มีหลากหลายรูปแบบมาเก็บในคลังข้อมูล โดยผ่านกระบวนการของ Extract Transform Load (ETL) ในรูปแบบข้อมูลที่มีโครงสร้าง และถูกจัดทำให้อยู่ในรูปแบบที่เหมาะสมสำหรับการนำไปวิเคราะห์ข้อมูล ทั้งในรูปแบบของรายงานอัจฉริยะ (Business Intelligence) และดาตากอนาไลติกส์ (Data Analytics)

• ทะเลสาบข้อมูล (Data Lake)

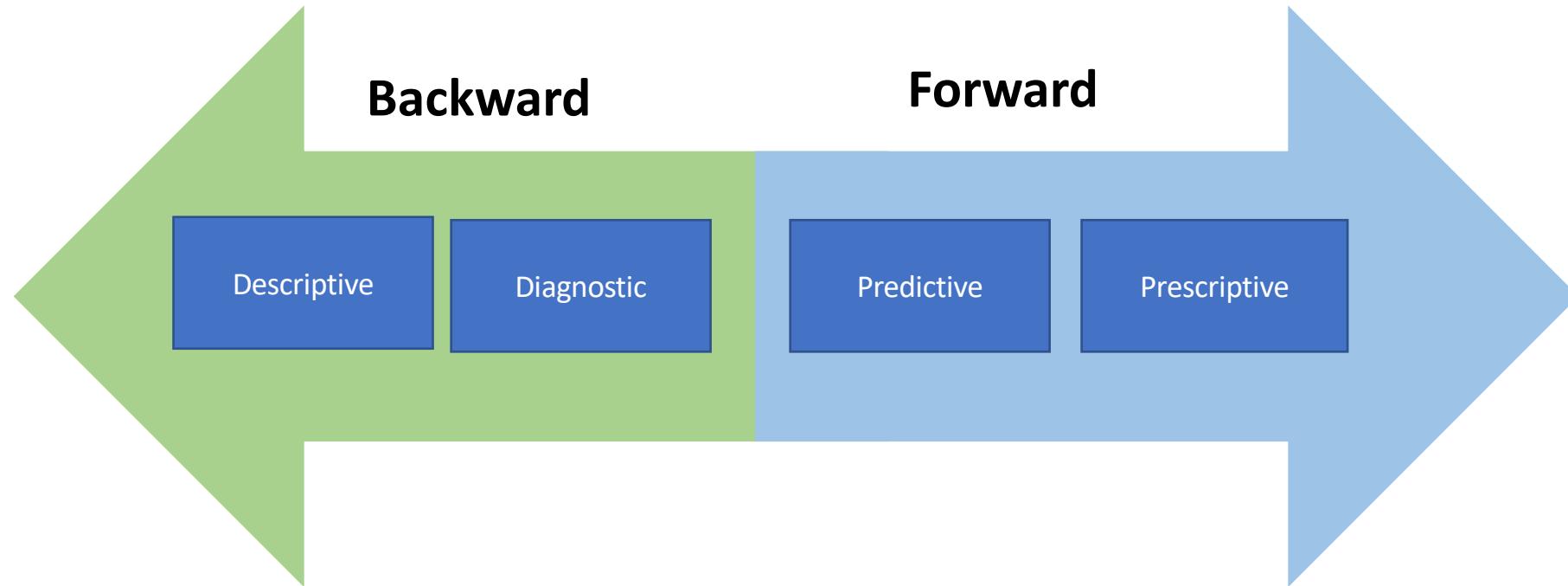
- เป็นแหล่งสำหรับเก็บรวบรวมข้อมูลที่มีหลากหลายรูปแบบ ข้อมูลที่จัดเก็บเป็นข้อมูลที่มีโครงสร้าง ข้อมูลกึ่งโครงสร้าง และข้อมูลที่ไม่มีโครงสร้าง โดยข้อมูลถูกเก็บรักษาไว้ในรูปแบบที่เหมือนหรือใกล้เคียงกับรูปแบบที่ได้รับมาจากแหล่งข้อมูลต้นฉบับ และสามารถใช้เป็นที่สำรองข้อมูลต้นฉบับได้



Analytic Value Escalator



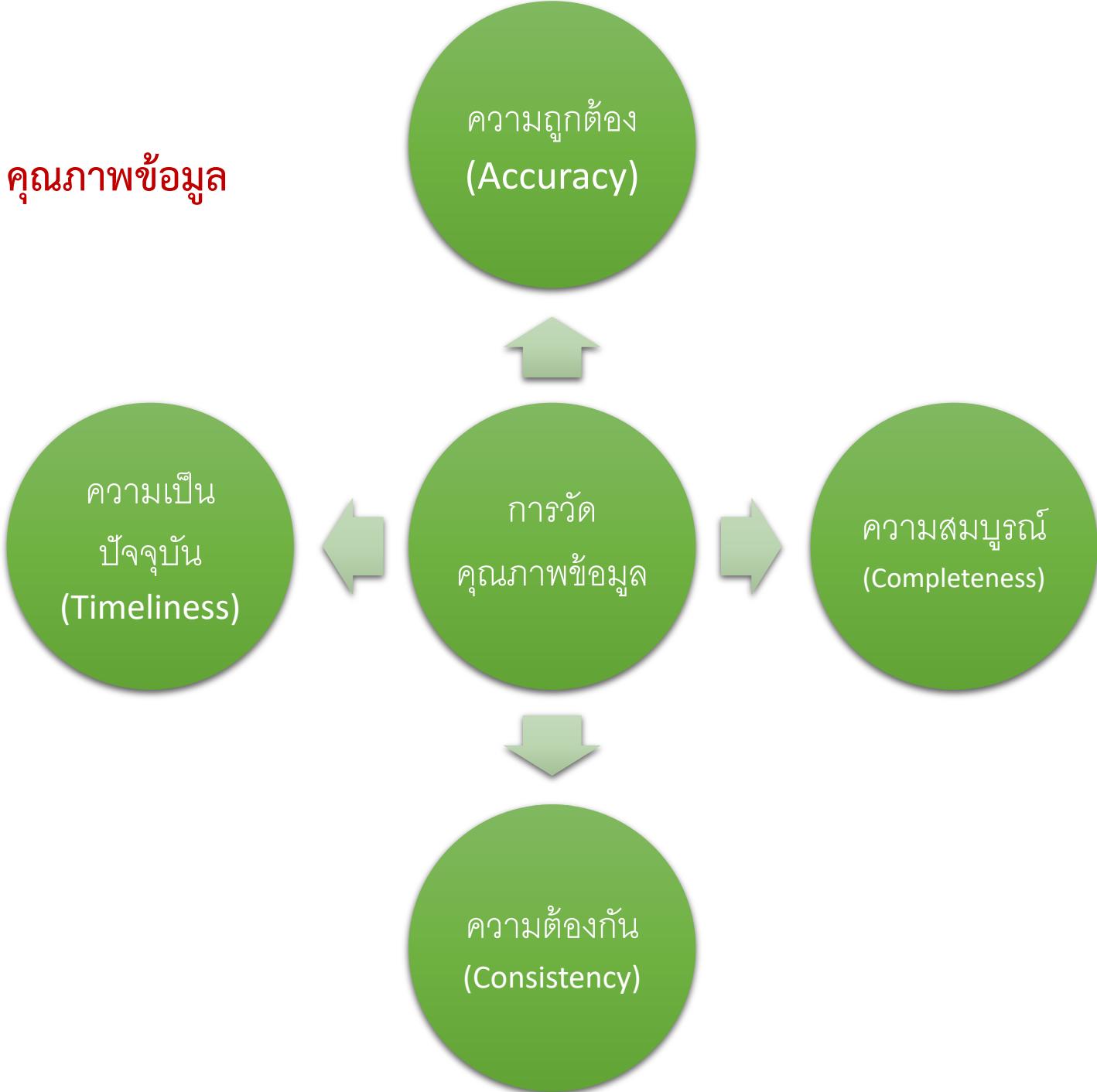
Types of Data Analytics



คุณภาพของข้อมูล (Data Quality)

- คุณภาพของข้อมูล (Data Quality) เป็นเครื่องมือในการวัดความน่าเชื่อถือและประสิทธิภาพของการนำข้อมูลไปใช้ ต้องมีการวางแผน การดำเนินการ และการควบคุมกิจกรรมต่าง ๆ รวมถึงการปรับปรุง เพื่อให้ข้อมูลมีคุณภาพ เนื่องจากข้อมูลที่มีคุณภาพสูงทำให้การดำเนินงานของหน่วยงานเป็นไปอย่างมีประสิทธิภาพ
- การทำให้ข้อมูลมีคุณภาพ ประกอบด้วย การทำให้ข้อมูลมีความถูกต้อง (Accuracy) ข้อมูลมีความครบถ้วน (Completeness) ข้อมูลมีความต้องกัน (Consistency) ข้อมูลมีความเป็นปัจจุบัน (Timeliness) ข้อมูลตรงตามความต้องการของผู้ใช้ (Relevancy) และข้อมูลมีความพร้อมใช้ (Availability)

ด้านการวัดผล - คุณภาพข้อมูล



ด้านการวัดผล - ตัวอย่าง

คุณภาพข้อมูล	รูปแบบการวัด	หน่วยวัด	ตัวอย่าง
ความถูกต้อง	ແຄວxຟິລົດ	ຮ້ອຍລະ	<p>ถ้าพบร่วมມື ៨០ ພິລົດທີ່ມີຄວາມຄຸກຕ້ອງຕລອດທັງ ១,០០០ ດັນ ດັນນັ້ນຊຸດຂໍ້ອມູນລົດນີ້ມີຄວາມຄຸກຕ້ອງ</p> $= (១,០០០ \times ៨០) / (១,០០០ \times ១០០) \times ១០០$ $= ៨០$ <p>กรณีที่ ១ : ພິຈາລະນາເພາະແຄວຂໍ້ອມູນ</p> <p>ถ้าพบร่วມການບັນທຶກຂໍ້ອມູນ ៩០០ ດັນ ໂດຍໄໝສັນໃຈຈຳນວນພິລົດທີ່ມີການບັນທຶກ ດັນນັ້ນຂໍ້ອມູນຊຸດນີ້ມີຄວາມຄຽບຄ້ວນ</p> $= (៩០០ / ១,០០០) \times ១០០$ $= ៩០$ <p>กรณีที่ ២ : ພິຈາລະນາແຄວແລະພິລົດຂໍ້ອມູນທີ່ມີຄວາມຈຳເປັນເທົ່ານັ້ນ</p> <p>ถ้าກຳທັນດໄທມື ៨០ ພິລົດທີ່ມີຄວາມຈຳເປັນຕ້ອງບັນທຶກຂໍ້ອມູນ ແລ້ວພຽງມື ៦០ ພິລົດຈາກ ៨០ ພິລົດ ທີ່ມີການບັນທຶກຂໍ້ອມູນທັງ ១,០០០ ດັນ ດັນນັ້ນຂໍ້ອມູນ</p> $= (១,០០០ \times ៦០) / (១,០០០ \times ៨០) \times ១០០$ $= ៧៥$ <p>กรณีที่ ៣ : ພິຈາລະນາແຄວແລະພິລົດຂໍ້ອມູນ</p> <p>ถ้าพบรຽງມື ៦០ ພິລົດຈາກ ១០០ ພິລົດ ທີ່ມີການບັນທຶກຂໍ້ອມູນທັງ ១,០០០ ດັນ ດັນນັ້ນຂໍ້ອມູນຊຸດນີ້ມີຄວາມຄຽບຄ້ວນ</p> $= (១,០០០ \times ៦០) / (១,០០០ \times ១០០) \times ១០០$ $= ៦០$ <p>ถ้าພຽງມື ២០ ພິລົດທີ່ເກີບຫ້າຂໍ້ອນກັບຊຸດຂໍ້ອມູນອື່ນແລະມີຮູບແບບຂອງພິລົດທີ່ແຕກຕ່າງກັນ ເຊັ່ນ ຮູບແບບວັນທີ ຮູບແບບຮັສ ດັນນັ້ນຂໍ້ອມູນຊຸດນີ້ມີຄວາມ</p> <p>ຄວາມຕ້ອງກັນ</p> <p>ພິລົດ</p> <p>ຮ້ອຍລະ</p>
ความสมบูรณ์			

ด้านการวัดผล - ตัวอย่าง

ความมั่นคงปลอดภัยของข้อมูล	หน่วยวัด	ตัวอย่าง
ความลับ		<ul style="list-style-type: none"> จำนวนการพิจารณาบทหวานโดยความมั่นคงปลอดภัยสารสนเทศและมาตรการควบคุมและป้องกันการเข้าถึงข้อมูล
ความถูกต้อง	จำนวนครั้ง	<ul style="list-style-type: none"> จำนวนการส่งเสริม สื่อสารสร้างความตระหนักรู้ด้านความมั่นคงปลอดภัย
ความพร้อมใช้		<ul style="list-style-type: none"> จำนวนการทดสอบความต่อเนื่องของการให้บริการและการนำข้อมูลกลับมาใช้ ร้อยละของการปฏิบัติตามนโยบายข้อมูลที่ได้กำหนดไว้ $= \frac{\text{จำนวนข้อกำหนดของนโยบายที่เป็นไปตามที่กำหนด}}{\text{จำนวนข้อกำหนดของนโยบายทั้งหมด}} \times 100$
ความลับ		<ul style="list-style-type: none"> ร้อยละของจำนวนเหตุล้มเหลวเมิดความมั่นคงปลอดภัยที่ควบคุมได้ เช่น ร้อยละของการถูกเผยแพร่ข้อมูลโดยไม่ได้รับอนุญาต
ความถูกต้อง	ร้อยละ	$= \frac{\text{จำนวนเหตุล้มเหลวเมิดความมั่นคงปลอดภัยที่ควบคุมได้}}{\text{จำนวนเหตุล้มเหลวเมิดความมั่นคงปลอดภัยทั้งหมด}} \times 100$
ความพร้อมใช้		<ul style="list-style-type: none"> ร้อยละของจำนวนข้อร้องเรียนด้านความมั่นคงปลอดภัยข้อมูลที่ลดลง เช่น ร้อยละของจำนวนข้อร้องเรียนจากการเปิดเผยข้อมูลส่วนบุคคลโดยไม่ได้รับการอนุญาต $= \frac{\text{จำนวนข้อร้องเรียนปีปัจจุบัน} - \text{ปีที่ผ่านมา}}{\text{จำนวนข้อร้องเรียนปีที่ผ่านมา}} \times 100$

Softnix Data Platform

Catalog & Metadata Management



Version 2.2.3

TIME INTERVAL SELECTED
Off

DATE SELECTED
Today

Administrator

Detail of model

Model name
people_dead

Model description
เป็นโมเดลที่รวมระหัสชื่อมูล ข้อมูลอัตตราการผ่าตัวตาย กรมสุขภาพจิต, ข้อมูลผู้ป่วยโรคซึมเศร้า จากศูนย์วิจัยและฝึกอบรมประจำสำนักงานแห่งชาติ ที่เชื่อมโยงกับ โรคซึมเศร้า และ ข้อมูลประชากรและจำนวนผู้เสียชีวิต จากสำนักงานสถิติแห่งชาติ

Category

Data access levels
public

Spatial coverage
Thailand

Update frequency
historical_only

Contact name
รพจ. เลยราชนครินทร์ โครงการช่วยเหลือผู้ที่เสี่ยงต่อการฆ่าตัวตาย รพจ.ชอนแก่นราชนครินทร์

Contact email

Created by
Administrator

Created at
Aug 19, 2019, 2:08:14 PM

Updated at
Aug 19, 2019, 2:08:14 PM

Detail of data store

Data store name
Hadoop40

Data store description

Host
192.168.10.41

Port
8090

Catalog
hive

Username
hdfs

Created by
Administrator

Created at
Aug 19, 2019, 2:07:36 PM

Updated at
Aug 19, 2019, 2:07:36 PM

Detail of selected field

Field name	Type	Description
province	varchar	จังหวัด
people_male	integer	จำนวนประชากรเพศชาย
people_female	integer	จำนวนประชากรเพศหญิง
people_all	integer	จำนวนประชากรทั้งหมด
dead_male	integer	จำนวนผู้เสียชีวิตเพศชาย
dead_female	integer	จำนวนผู้เสียชีวิตเพศหญิง
dead_all	integer	จำนวนผู้เสียชีวิตทั้งหมด

Data Lineage



Softnix Data Platform

Activity Auditing



Version 2.2.3

TIME INTERVAL SELECTED
Off

DATE SELECTED
Today

Export csv

Search

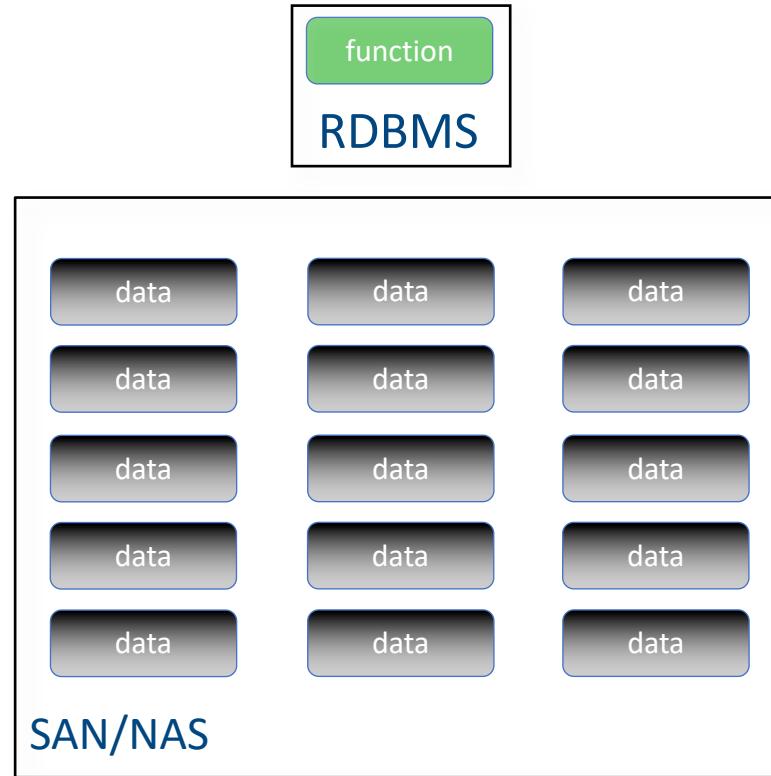
Time	Ip	User	Message
2019-08-30 23:49:53	192.168.10.254	admin	Login successful
2019-08-29 22:27:31	192.168.10.254	admin	Login successful
2019-08-29 15:07:13	192.168.10.254	admin	Store-model Recruitment updated successfully
2019-08-29 15:02:29	192.168.10.254	admin	Mpp-visualize Openness & Agreeable updated successfully
2019-08-29 14:52:57	192.168.10.254	admin	Login successful
2019-08-29 13:24:32	192.168.10.254	admin	Mpp-visualize Openness & Agreeable updated successfully
2019-08-29 13:14:42	192.168.10.254	admin	Mpp-visualize APTnumerical & APTverbal updated successfully
2019-08-29 13:13:55	192.168.10.254	admin	Mpp-visualize Openness & Agreeable updated successfully
2019-08-29 13:13:02	192.168.10.254	admin	Mpp-visualize Openness & Agreeable updated successfully
2019-08-29 13:10:36	192.168.10.254	admin	Mpp-visualize Openness & Agreeable updated successfully
2019-08-29 13:08:34	192.168.10.254	admin	Mpp-visualize Openness & Agreeable updated successfully



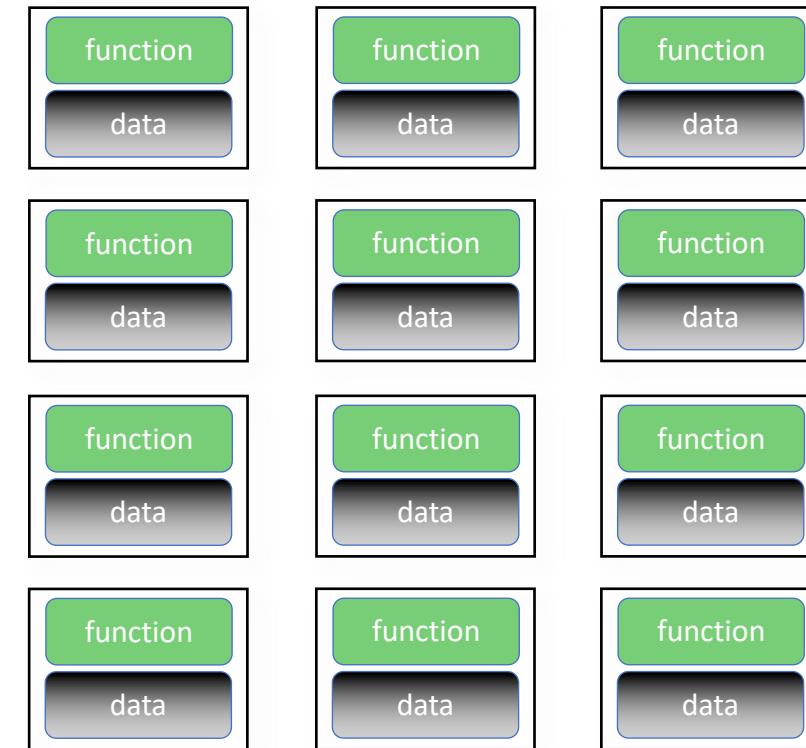
Softnix Data Platform & Workshop

Ship the Function to the Data

Traditional Architecture



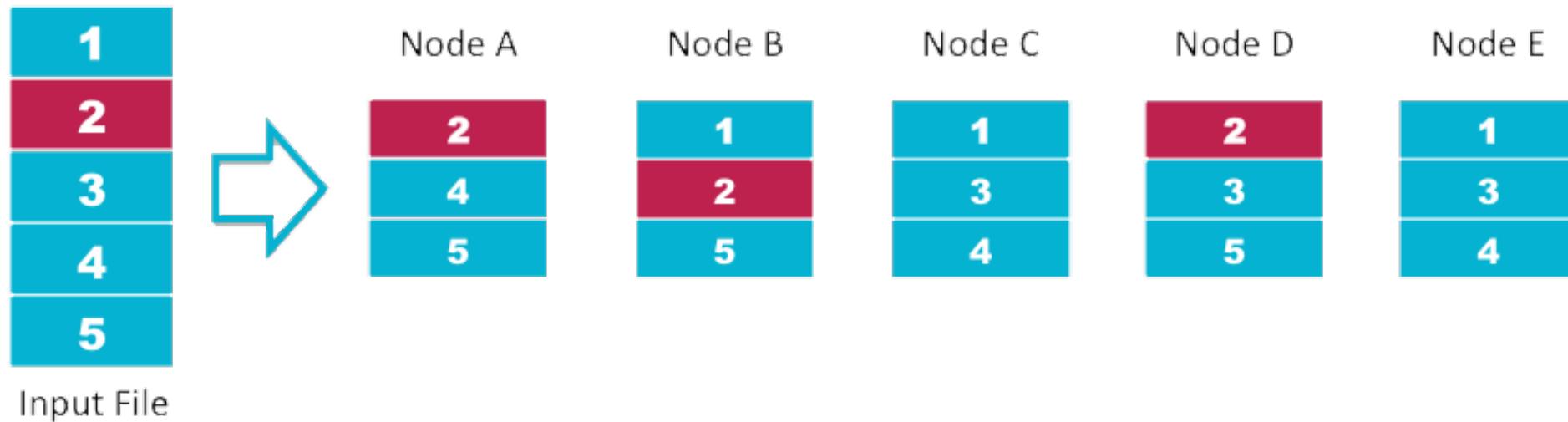
Distributed Computing



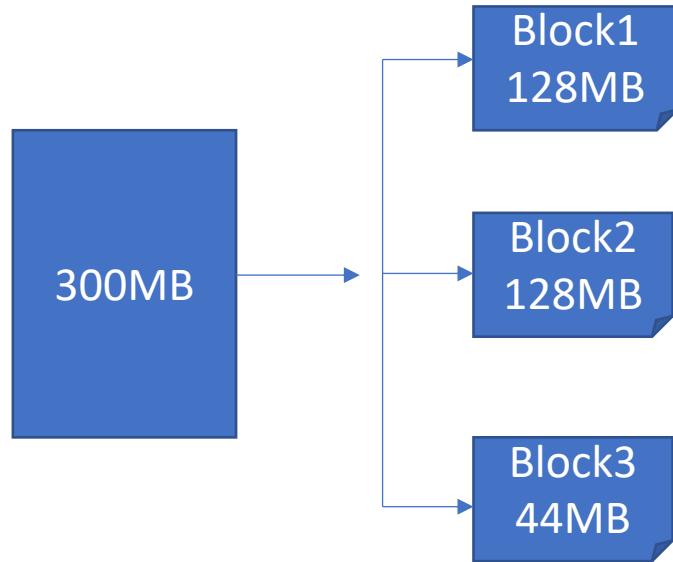
HDFS



HDFS Data Distribution



HDFS Block Concept

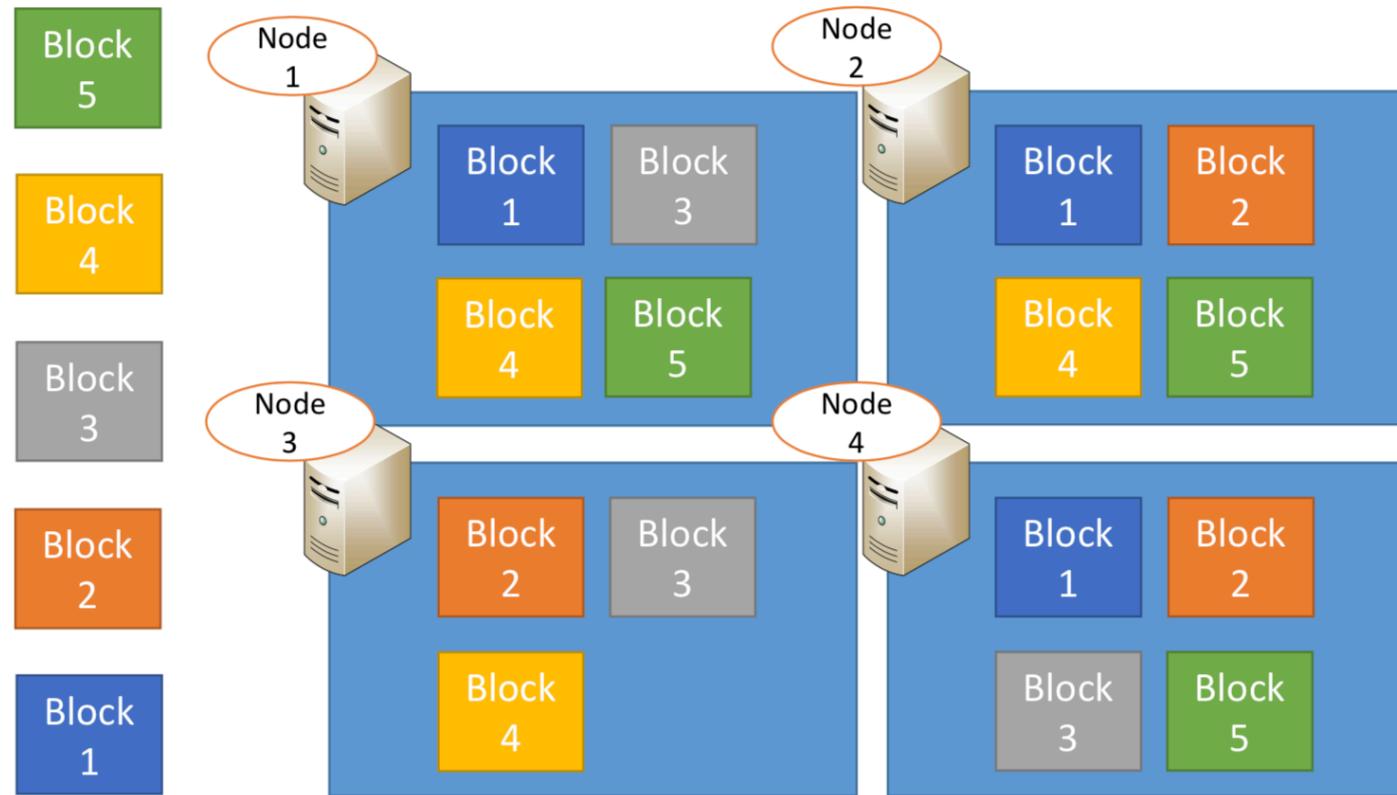


Default block size
128MB

File will be sliced into multiple-chunk called Block

- Default Size 128MB per block
- If file is smaller than block size, it will store **without padding**

HDFS Replication



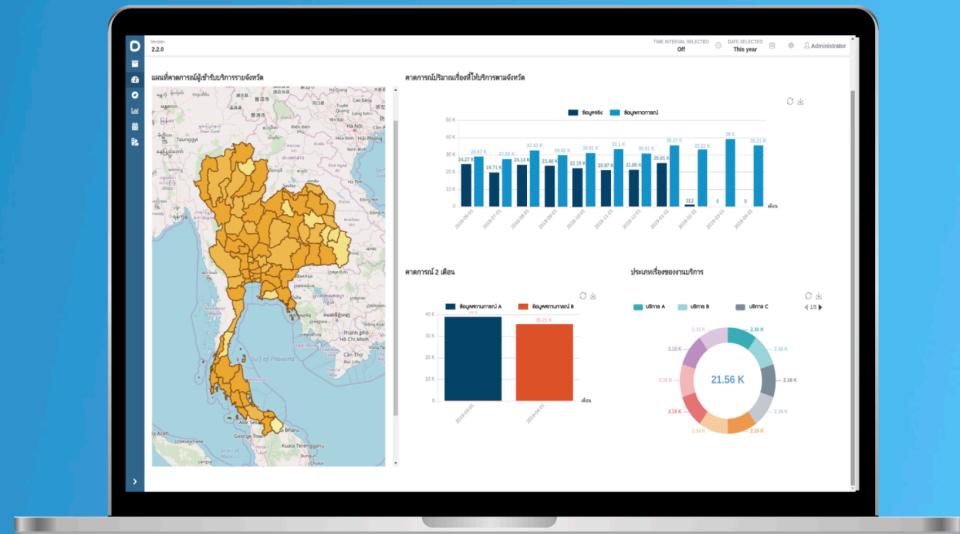
Default block size
128MB

SOFTNIX DATA PLATFORM

Enterprise Data Lake helps you accelerate time to insight.



Softnix
DATA PLATFORM



End 2 End Data Lake Platform



- Structured & Unstructured Data Sources
- Flexibility

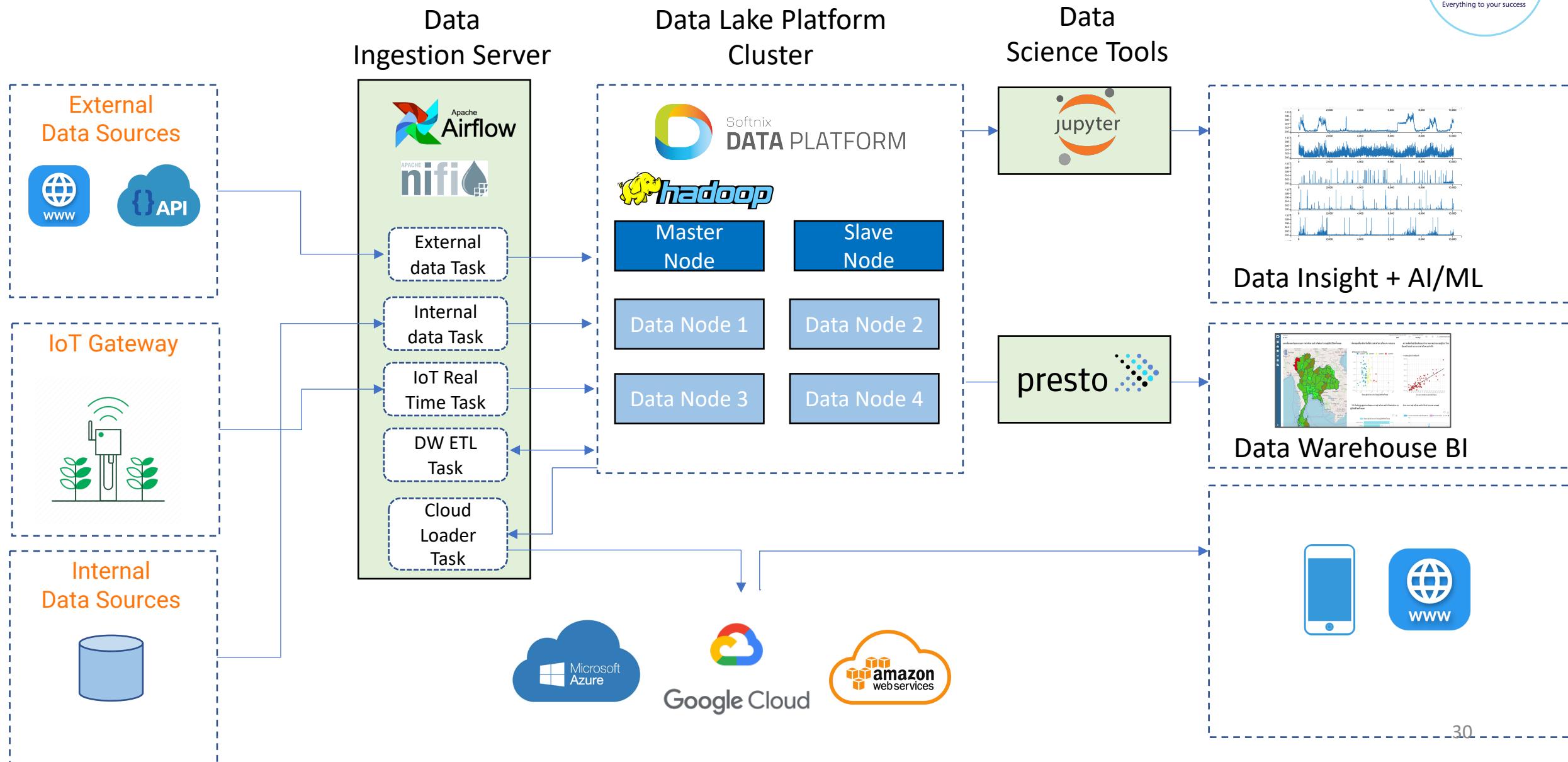
- Data Governance
- PDPA Compliance
- Security
- Scalability

- Faster No movement data
- Interactive BI
- RBAC
- Multi Tenancy

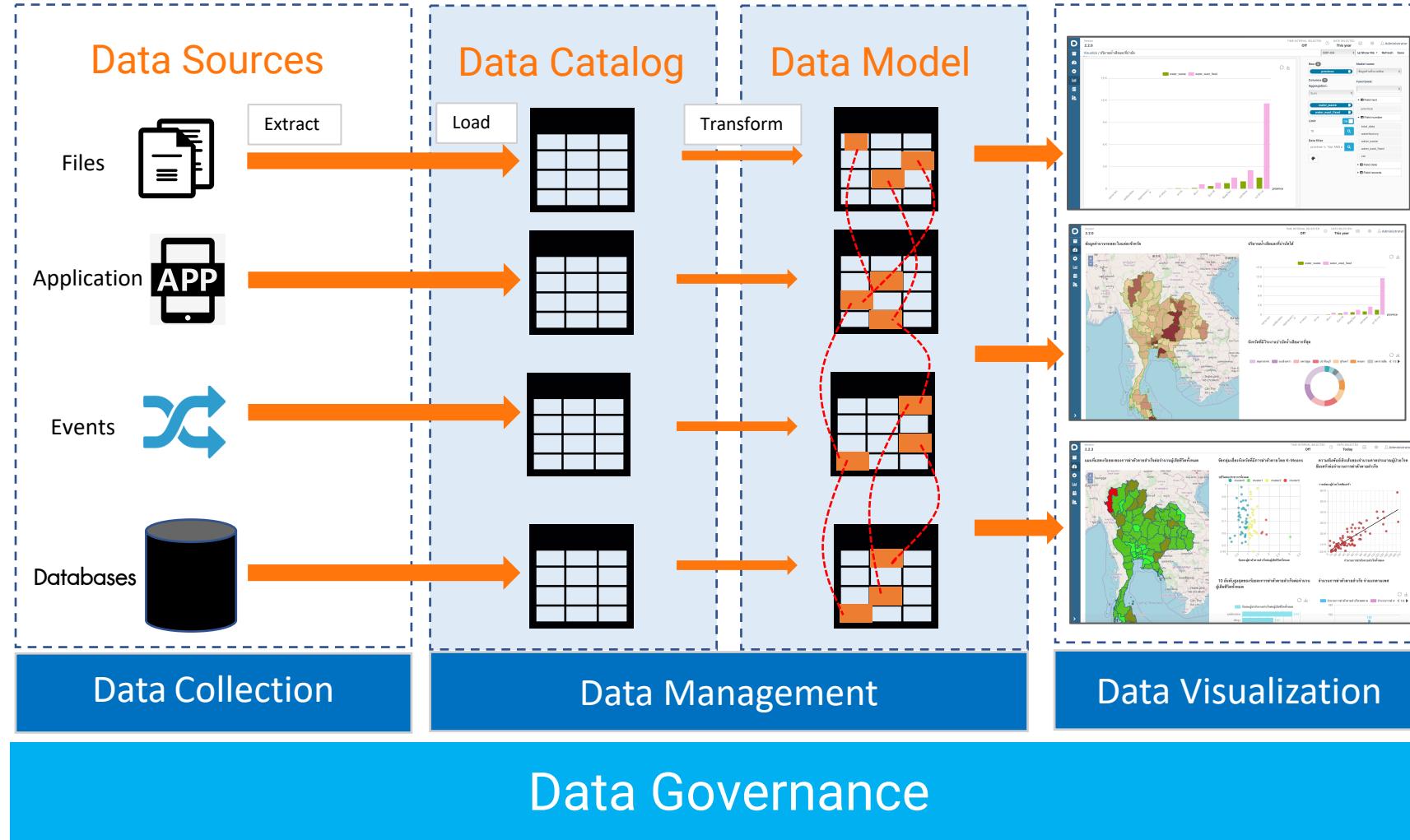
Time to Insight

Value in Days

Low Level Architecture



Softnix Data Platform Processes





VPN Client Download
<https://vpn.softnix.co.th:10443>

Access GUI
<https://forticlient.com/downloads>

User = demopoc2-7
Password = demo@1234

WIFI = superdivision3

http://192.168.10.78:3030

Username = afsc[1-12]
Password = afscmi2o2o

Softmix DATA PLATFORM Version 2.2.4

Project

เมนูหลัก

เมนูบัน

TIME INTERVAL SELECTED
Off

DATE SELECTED
Last 15 minutes

Administrator

Search +

« 1 2 »

Name	Description	Create By	Action
<input type="checkbox"/> Project_test		Administrator	

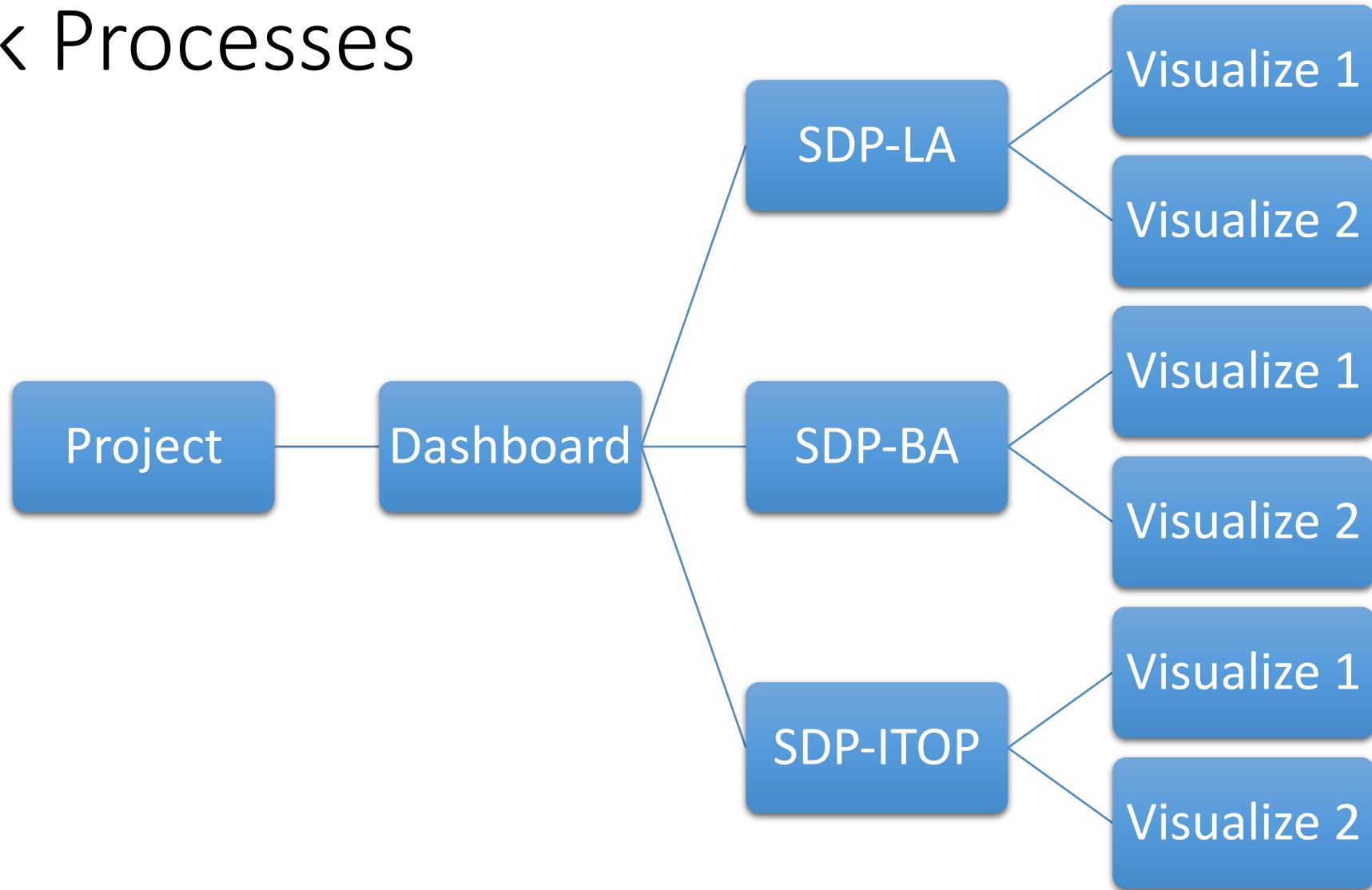
รายการโครงการ



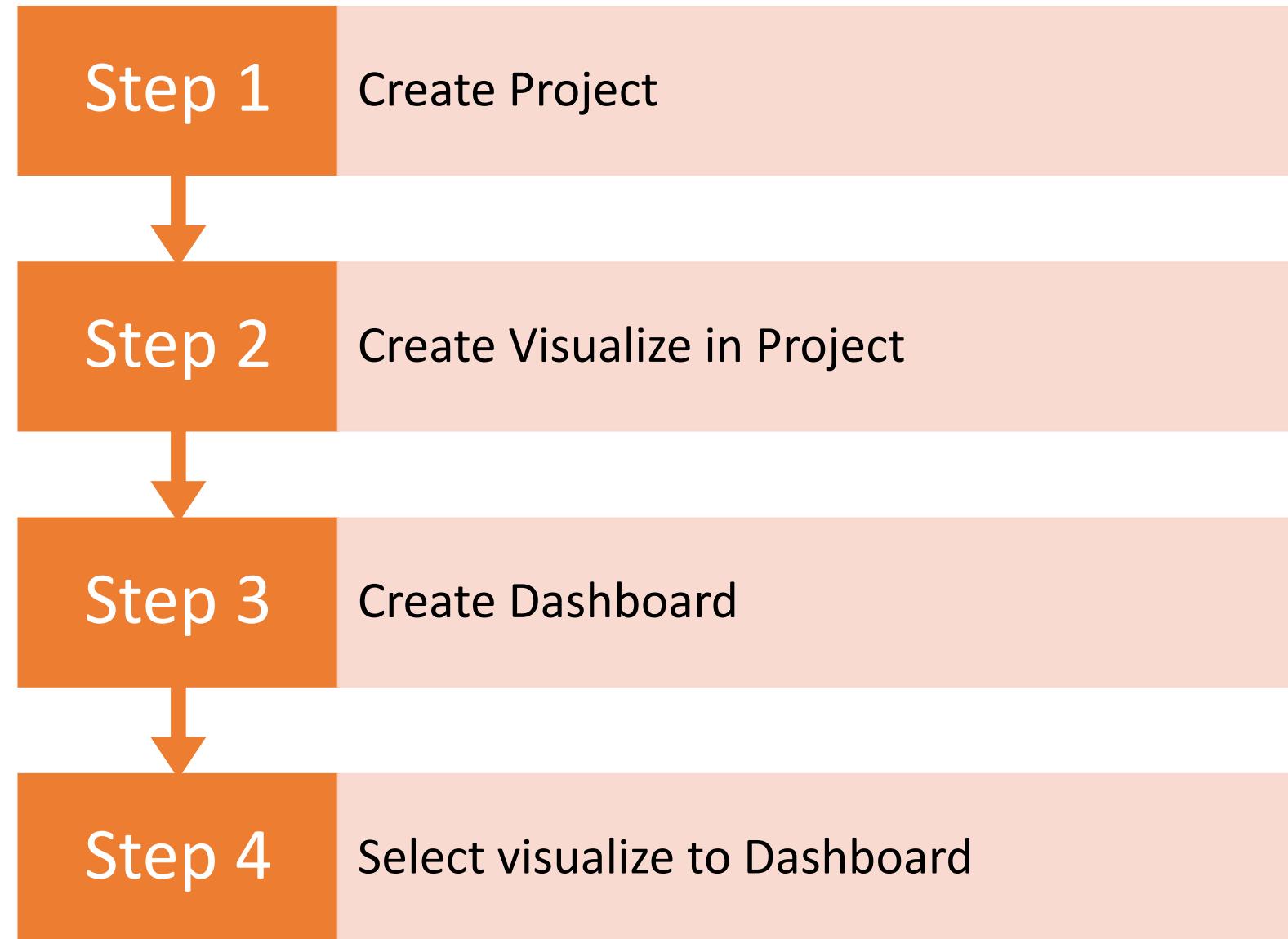
Module

1. Softnix Data Platform for Log Analytics (SDP-LA)
2. Softnix Data Platform for Business Analytics (SDP-BA)
3. Softnix Data Platform for IT Operational Services (SDP-ITOPS)

Work Processes



Step to Dashboard



Demo



Version 2.3.1

TIME INTERVAL SELECTED Off DATE SELECTED Today Administrator

counts

Category	Counts
politic	3.37 K
crime	712
foreign	659
dailynews	1
...	...

counts

Category	Counts
...	...

ประเทศไทยที่เกี่ยวข้อง

counts

Category	Counts
...	...

บุคคลที่เกี่ยวข้อง

counts

Category	Counts
...	...

หัวข้อข่าว

Title	Link
ก่อนหน้าข่าวล่าสุดที่เปิดให้เข้าชม.	https://dailynews.co.th/politics/785889
อุบัติเหตุทางเรือในกรุงเทพฯ ดับ 1 ราย	https://dailynews.co.th/politics/785888
เชฟรอน ยื่นขออนุญาตผลิตก๊าซธรรมชาติในไทย	https://dailynews.co.th/politics/785885
กลับมาแล้ว!!! สำนักงานพัฒนาธุรกรรมทางอิเล็กทรอนิกส์ จัดอบรมเชิงปฏิบัติการ	https://dailynews.co.th/politics/785884
สันติํ “พร้อมแล้วให้ไว้การค้ารวม หลังเจรจาต่อรองฯ ก้าวผล. นราฯ”	https://dailynews.co.th/politics/785882
นายกฯ บินเยือนเวียดนามปักป้ายเฉลิมพระเกียรติ 24 ก.ค.	https://dailynews.co.th/politics/785879
นักปีศาจ “รวมคนดี เมืองคอนเด็” ไม่แพ้พ่อแม่	https://dailynews.co.th/politics/785873



Data Catalog

- Manage Data Store Connection
- Manage Social Connection
- Create Data Model
- Metadata Data Model
- Data Profiling

Data Store Connection

- รองรับการเชื่อมต่อ Hadoop Cluster



TIME INTERVAL SELECTED: Off DATE: 1

Data Catalog		Social Connection	Details Connection	Select 1 model
<input type="checkbox"/> Name	Description			Create By
<input type="checkbox"/> people_dead	เป็นโมเดลที่รวมระหว่างข้อมูล ข้อมูลอัตราการมาตัวตาย กรมสุขภาพ จิต, ข้อมูลผู้ป่วย โรคซึมเศร้า จากศูนย์วิจัยและฝึกอบรมประสานงาน เกี่ยวกับตัวชี้วัดเกี่ยวกับ โรคซึมเศร้า และ ข้อมูลประชากรและจำนวนผู้ เสียชีวิต จากสำนักงานสถิติแห่งชาติ			Administrator
<input type="checkbox"/> techtalk_time				Administrator

Hadoop Cluster

Detail of data store

Data store name:

Host: Port:

Description:

Catalog:

Authentication:

Username:

or

ZABBIX API

Detail of data store

Data store name:

Url:

Description:

Username:

Password:

or

ZABBIX Direct Database

Detail of data store

Data store name:

Host: Port:

Description:

Database:

Username:

Password:

or





Data Connection

- Facebook API
- Google Analytics API
- Instagram API
- Upload Files

Metadata

DATA PLATFORM
Version 2.2.4

Project Dashboard Discover Visualization Schedule Report Data Catalog

Detail model / people_dead

Data Model Data Source Data Profile

TIME INTERVAL SELECTED: Off DATE SELECTED: Today Administrator

Detail of model

Metadata

Model name: people_dead

Model description: เป็นโมเดลที่มีความซับซ้อนอยู่บ้าง ข้อมูลอัตตราการผ่าตัวตาย กรมสุขภาพจิต, ข้อมูลผู้ป่วยโรคซึมเศร้า จากศูนย์และฝึกอบรมปะร奔งานที่เกี่ยวกับตัวชี้วัดเกี่ยวกับโรคซึมเศร้า และ ข้อมูลประชากรและจำนวนผู้เสียชีวิต จากสำนักงานสถิติแห่งชาติ

Category:

- Data access levels: public
- Spatial coverage: Thailand
- Update frequency: historical_only
- Contact name: รพ. เลยราชนครินทร์ โครงการช่วยเหลือผู้ที่เสี่ยงต่อการฆ่าตัวตาย รพ.ช.ชอนแก่น ราชบุรี
- Contact email: [redacted]
- Created by: Administrator
- Created at: Aug 19, 2019, 2:08:14 PM
- Updated at: Aug 19, 2019, 2:08:14 PM

Detail of data store

Storage

Data store name: Hadoop40

Data store description:

Host	192.168.10.41
Port	8090
Catalog	hive
Username	hdfs
Created by	Administrator
Created at	Aug 19, 2019, 2:07:36 PM
Updated at	Aug 19, 2019, 2:07:36 PM

Detail of selected field

Data Type

Field name	Type	Description
province	varchar	จังหวัด
people_male	integer	จำนวนประชากรเพศชาย
people_female	integer	จำนวนประชากรเพศหญิง
people_all	integer	จำนวนประชากรทั้งหมด
dead_male	integer	จำนวนผู้เสียชีวิตเพศชาย
dead_female	integer	จำนวนผู้เสียชีวิตเพศหญิง
dead_all	integer	จำนวนผู้เสียชีวิตทั้งหมด
suicide_male	integer	จำนวนการฆ่าตัวตายสำเร็จเพศชาย
suicide_female	integer	จำนวนการฆ่าตัวตายสำเร็จเพศหญิง
suicide_all	integer	จำนวนการฆ่าตัวตายสำเร็จทั้งหมด

Data Viewer

Data Model Data Source Data Profile

Manage Data Source

Schema: default

Limit: 100 (ON)

Database

Table: Search table name

Table

- comment_logcafe
- dbs
- dhcp_list
- drone_image
- drone_videos
- jp_hostel
- log_firewall_secure
- log_frontend
- log_mail
- mail_auth_failed
- mockup_baac
- mockup_baac_view
- parquet_test
- people_dead

Export csv

province	people_male	people_female	people_all	dead_male	dead_female	dead_all	suicide_male	suicide_female	suicide_all
กรุงเทพมหานคร	233449	236398	469847	1471	981	2452	12	5	17
กาญจนบุรี	407843	407335	815178	3065	2238	5303	44	11	55
กำแพงเพชร	360218	366776	726994	2706	2190	4896	43	14	57
ขอนแก่น	887837	915043	1802880	7791	5778	13569	113	17	130
จันทบุรี	260753	271773	532526	2402	1718	4120	38	11	49
ฉะเชิงเทรา	347475	361419	708894	2948	2256	5204	25	7	32
ชลบุรี	734335	769228	1503563	6791	4377	11168	46	12	58
ชัยนาท	158180	170468	328648	1523	1346	2869	18	7	25
ชัยภูมิ	563910	574181	1138091	4285	3582	7867	57	16	73
ชุมพร	250440	255554	505994	1857	1302	3159	32	8	40
เชียงราย	573883	596349	1170232	5731	3897	9628	110	27	137
เชียงใหม่	782154	837946	1620100	8659	6193	14852	133	32	165
ตรัง	314099	327656	641755	2022	1564	3586	29	9	38
ตราด	109184	110635	219819	890	649	1539	15	3	18
ตาก	270499	267203	537702	1834	1410	3244	40	13	53

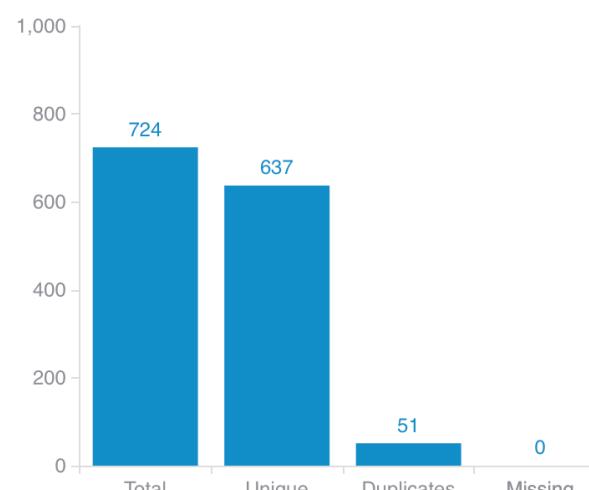
Data Profiling

Data Model **Data Source** **Data Profile**

Fields

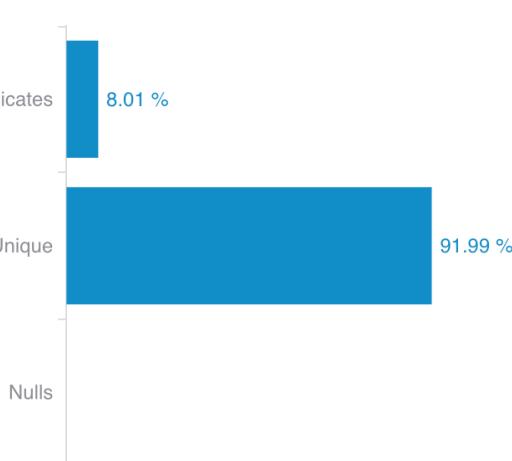
- name**
- date
- likes
- comment
- source
- link
- sentiment

Summary



Category	Count
Total	724
Unique	637
Duplicates	51
Missing	0

Relative Statistics



Category	Percentage
Duplicates	8.01 %
Unique	91.99 %
Nulls	0 %

Data Type String

Top Value

Value	Count
LazyLife Kit	20
mirik line	8
Tanachot Sreebuaram	5
saharat7	4
may thailand	4
Blue Egle	3
goo ka	3
Uder Ware	3
Rathaphong Cth KeKja	3
TheMoonlightIntheSky	3

Numeric Stats

max	47
avg	1.6477900552486189
min	0

Data Type Numeric (Integer)

ที่วางของ Power BI Desktop	แหล่งข้อมูลภายนอก	ลิงก์สำหรับข้อมูลเพิ่มเติม
	Cassandra	โปรแกรมควบคุม Cassandra ODBC
	Couchbase DB	Couchbase และ Power BI
	DynamoDB	โปรแกรมควบคุม DynamoDB ODBC
	Google BigQuery	โปรแกรมควบคุม BigQuery ODBC
	HBase	โปรแกรมควบคุม Hbase ODBC
	Hive	โปรแกรมควบคุม Hive ODBC
	IBM Netezza	ข้อมูล IBM Netezza
	Presto	โปรแกรมควบคุม ODBC Presto
	Project Online	บทความ Project Online
	Progress OpenEdge	บล็อกโพสต์โปรแกรมควบคุม Progress OpenEdge O

Connect Power BI Tools

ODBC Driver

<https://support.treasuredata.com/hc/en-us/articles/360000709167-Presto-ODBC-Connection>

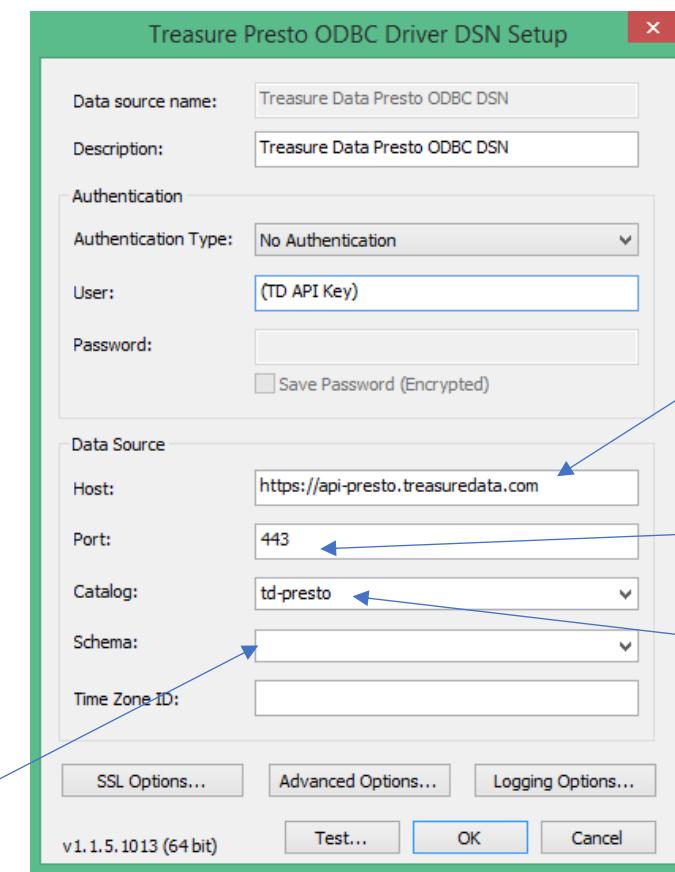
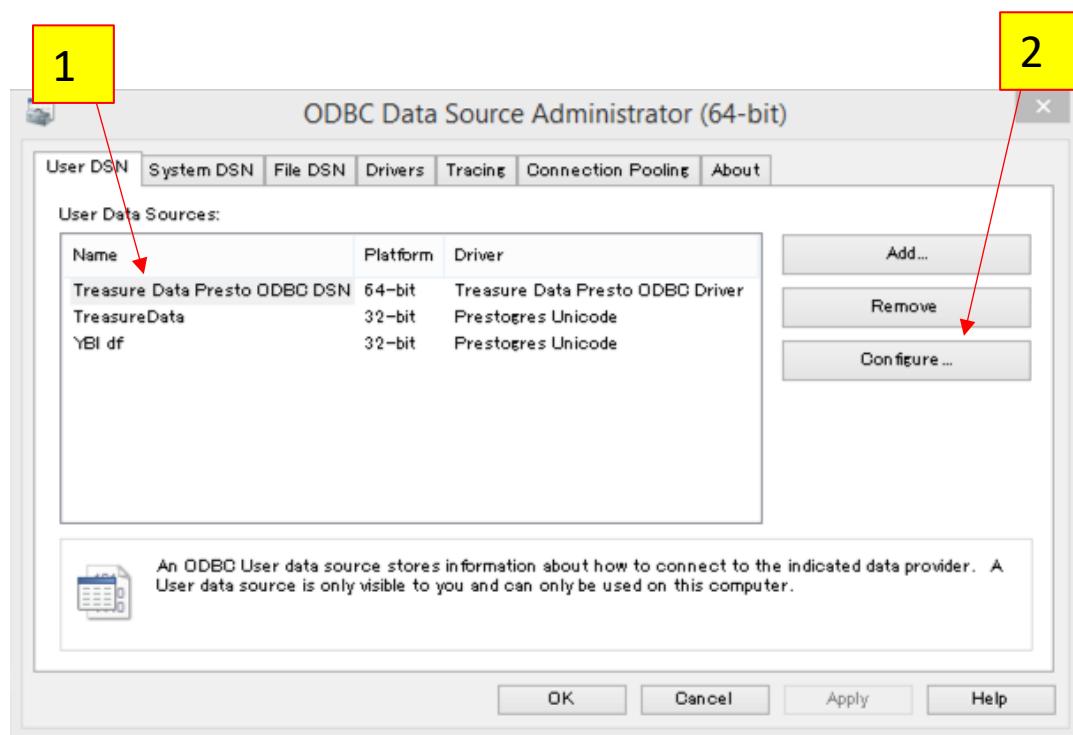
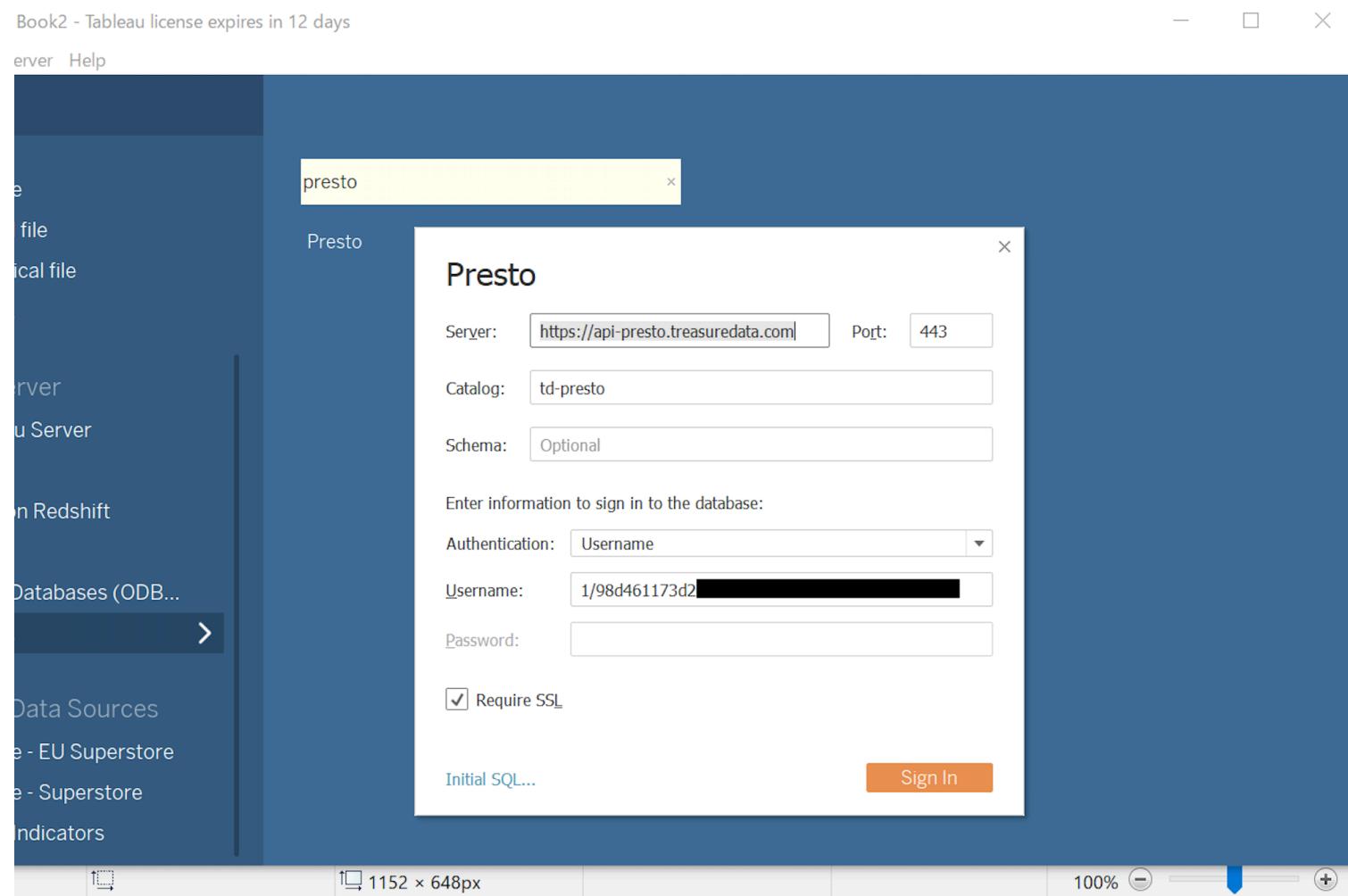


Tableau Presto



Enter information to sign in to the database:

Authentication: LDAP

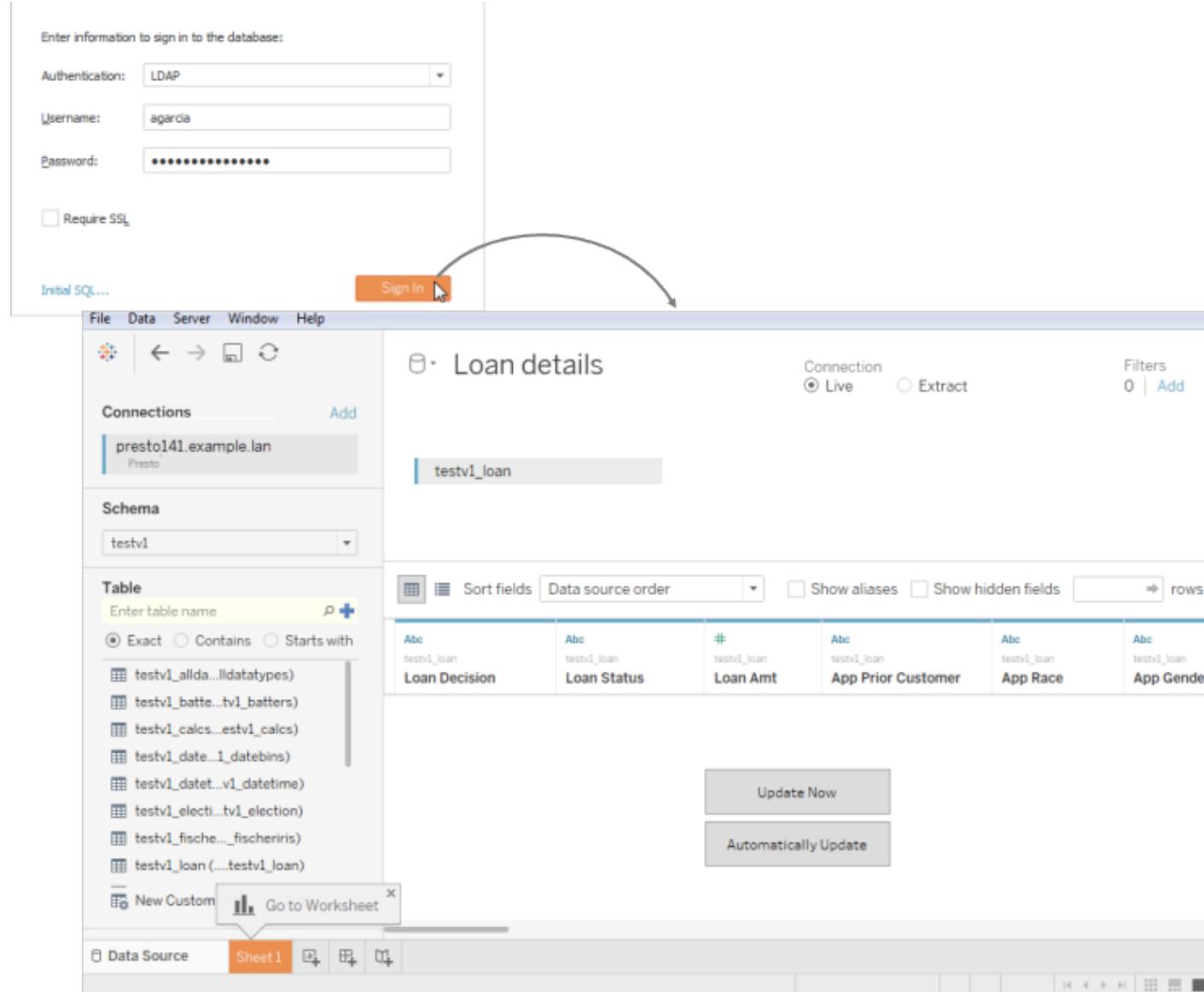
Username: agarda

Password: *****

Require SSL

Initial SQL...

Sign In



File Data Server Window Help

Connections Add

presto141.example.lan

Schema testv1

Table Enter table name

Exact Contains Starts with

- testv1_alldatatypes
- testv1_batteries_tv1_batters
- testv1_calcs...estv1_calcs
- testv1_date...1_datebins
- testv1_datet...v1_datetime
- testv1_electi...tv1_election
- testv1_fische..._fischeriris
- testv1_loan (...testv1_loan)

New Custom Go to Worksheet

Loan details

testv1_loan

Connection Live Extract Filters 0 | Add

Loan Decision	Loan Status	#	Loan Amt	App Prior Customer	App Race	App Gender
testv1_loan	testv1_loan	testv1_loan	testv1_loan	testv1_loan	testv1_loan	testv1_loan

Sort fields Data source order ▾

Show aliases Show hidden fields ↗ rows

Update Now

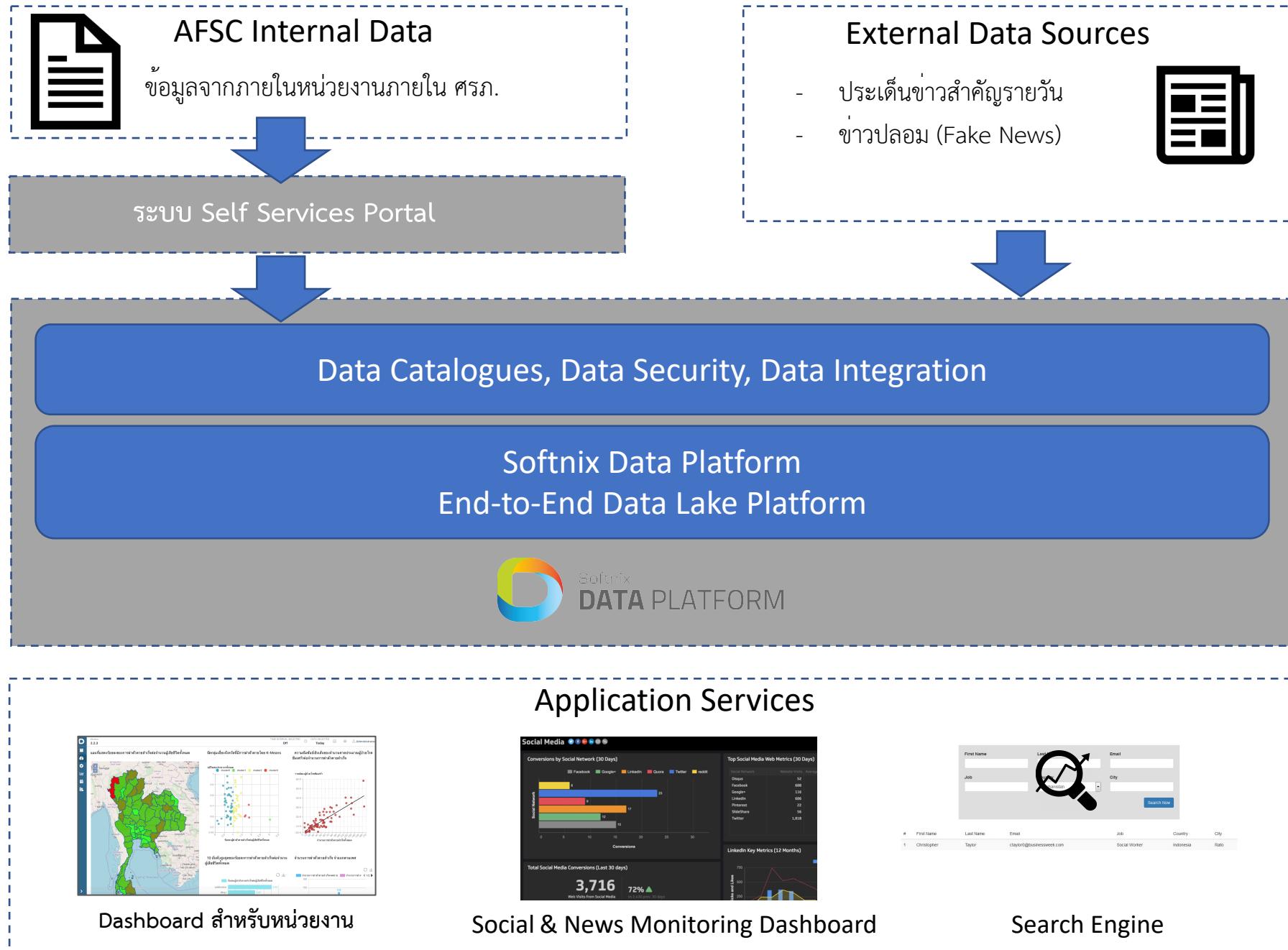
Automatically Update

Data Source Sheet1

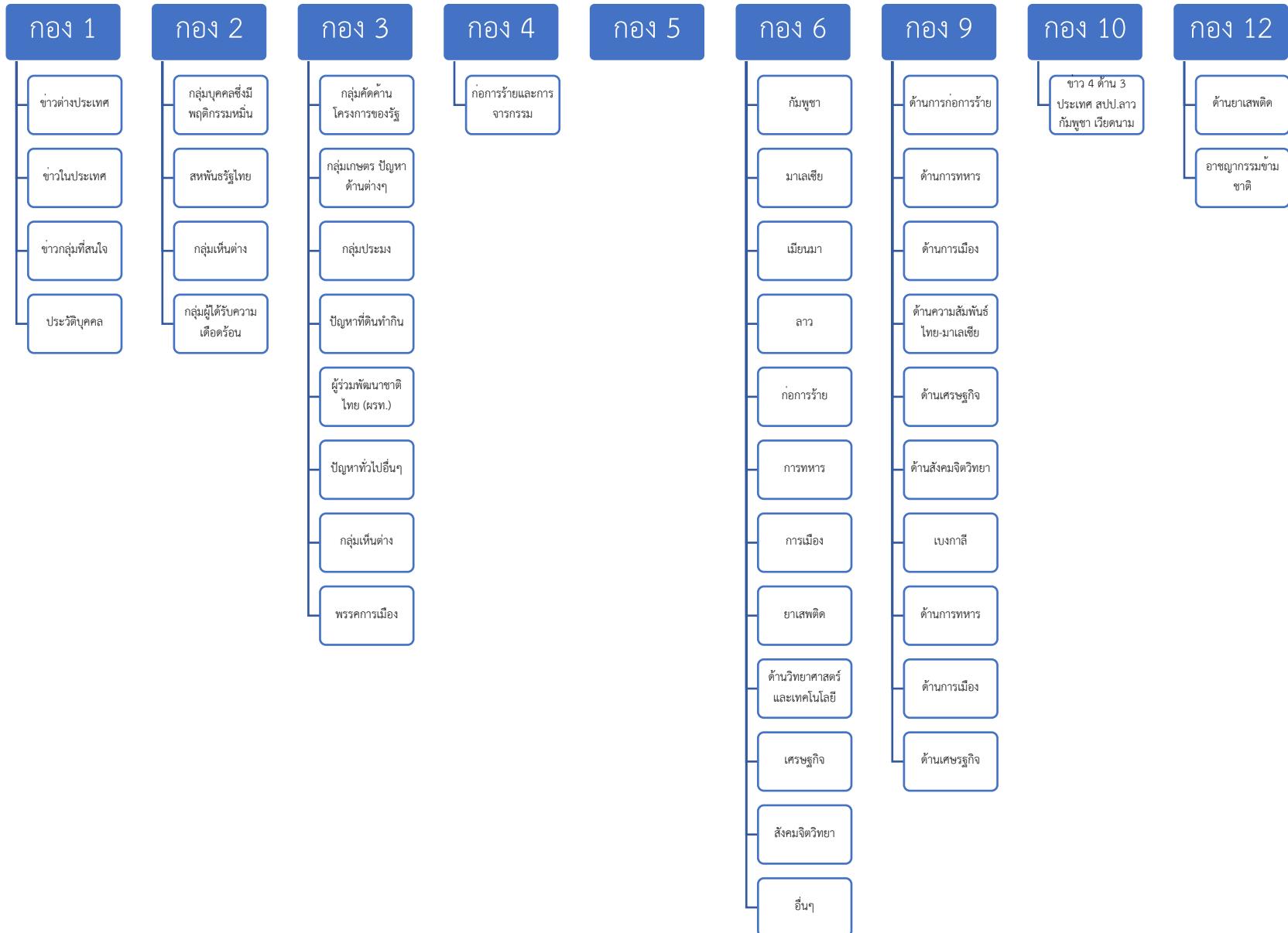
https://help.tableau.com/current/pro/desktop/en-us/examples_presto.htm



Armed Force Security Center (AFSC) Architecture Big Data Platform



AFSC Internal Data



ระบบจัดการนำเข้าข้อมูลด้วยตัวเองภายในหน่วยงาน



Version
2.3.1

Detail model /

Data Model Data Source Data Profile

TIME INTERVAL SELECTED Off DATE SELECTED Today Administrator

STEP 1 : UPLOAD CSV STEP 2 : MANAGE DATA SOURCE TITLE OF STEP 3

STEP 1 : Upload CSV

Choose files *.csv to upload

Field Name Field Type

Field Name	Field Type
Name	String
Surname	String
Date	Date
Number	Number
Float	Float

<< 1 2 >>

ระบบติดตามและตรวจสอบการนำเข้าข้อมูลอัตโนมัติ



Airflow DAGs Data Profiling ▾ Browse ▾ Admin ▾ Docs ▾ 21:25 UTC ⚡

Run: scheduled_2017-03-19T02:00:00 Layout: Left->Right Go Search for...

BranchPythonOperator PythonOperator SlackAPIPostOperator SubDagOperator success running failed skipped retry queued no status

```
graph LR; check_servers[check_servers] --> branch_check_cluster_up[branch_check_cluster_up]; branch_check_cluster_up --> slack_cluster_start[slack_cluster_start]; slack_cluster_start --> start_spark_cluster[start_spark_cluster]; start_spark_cluster --> branch_started_clust[branch_started_clust]; branch_started_clust --> slack_cluster_ok[slack_cluster_ok]; slack_cluster_ok --> crawler_dag_cluster_up_wkend[crawler_dag_cluster_up_wkend]; slack_cluster_start --> slack_unable_start_cluster[slack_unable_start_cluster]
```

TIME INTERVAL SELECTED Off DATE SELECTED Today

Detail model / crime

Data Model Data Source Data Profile

Fields

- address
- birthdate
- complain_date
- crime_case
- crime_number
- name
- id
- surname
- fullname
- nation
- org
- title
- want_d
- warrant_no

Summary

Total	Unique	Duplicates	Missing
10000	9626	366	0

Relative Statistics

Value	Count
พ.อ.ส.ส.ส./บ.ร.ส.ส.	3
พ.อ.ส.ส.ส./บ.ร.ส.ส.	2
พ.อ.ส.ส.ส./บ.ร.ส.ส.	2

Top Value

Value	Count
พ.อ.ส.ส.ส./บ.ร.ส.ส.	3
พ.อ.ส.ส.ส./บ.ร.ส.ส.	2
พ.อ.ส.ส.ส./บ.ร.ส.ส.	2

Detail drill down by chart

address	birthdate	complain_date	crime_case
36722 姣太平 บ้านบัววิชช์ สุรศรีท 32230	24/07/2509	10/06/2557	ผู้เสียหายภายนอกภารภัย หรือคนหาย
80/56 พีระมงคล บ้านนาทักษิณ นครศรีธรรมราช 30230	06/04/2494	29/12/2557	เหตุการณ์ทางเพศหรือเด็กในบ้าน
79/96 ห้องร้านเชิงชาร์จ ก้าวสู่ความสำเร็จ หาดทิ没能 80190	26/07/2493	20/01/2554	พบคนร้ายในบ้านค่าห้อง
195/35 บ้านเพชร บ้านกอกกลาง เที่ยวนคร 50230	23/01/2503	16/04/2554	พบคนร้ายในบ้านค่าห้อง
284/10 โภนทอง บ้านกอบบ้านบากอง หนองทราย 11110	17/11/2503	23/04/2560	บ้านถูกโจรกรรม
145/62 บุญเจต บ้านกอสันป่าเหลือง เชียงใหม่ 50120	18/12/2509	13/07/2560	ความเมื่อยล้าของผู้เสียหายในประเทศไทย
94/542 บ้านหนองบัว บ้านหนองบัว พระยาศรีเมืองราษฎร 13110	15/05/2505	27/03/2553	ผู้เสียหายร่างเข้าหากิน
44/993 บ้านกรุง บ้านกอบ้านกรุง บุรีรัมย์ 31180	28/07/2502	24/05/2553	ความเมื่อยล้าของผู้เสียหายในประเทศไทย
12/268 นาสุก บ้านกอชีวะวนหมุด บุรีรัมย์ 41280	29/08/2511	07/01/2551	พบคนร้ายในบ้านค่าห้อง
52/676 ทันชื่า บ้านกอสันค่า จังหวัดบุรีรัมย์ 22180	31/01/2509	07/08/2550	ความเมื่อยล้าบุก入

กราฟ crime_number

กราฟ ที่ 1 จำนวนผู้เสียหายในประเทศไทย

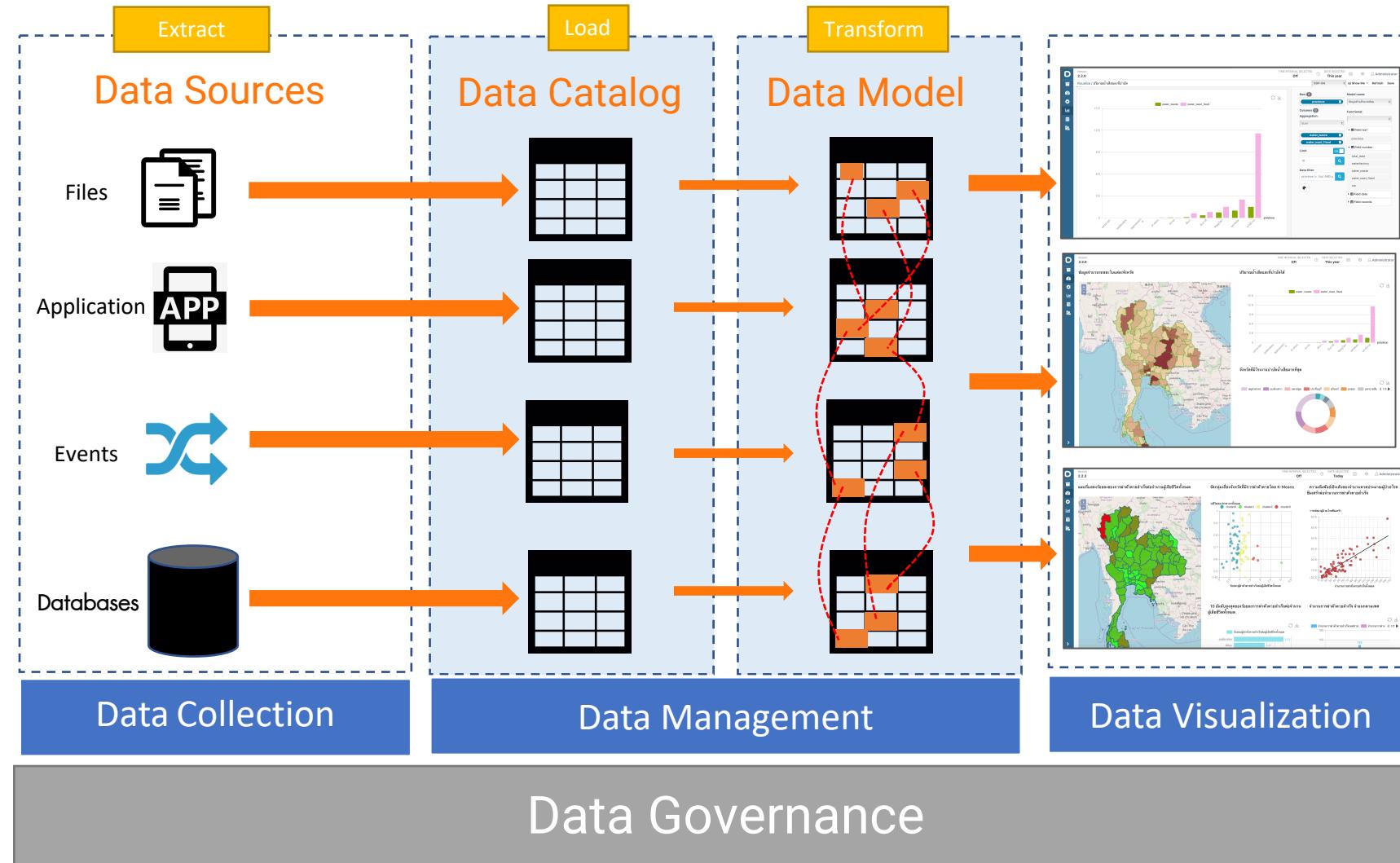
กราฟ ที่ 2 จำนวนผู้เสียหายในประเทศไทย

กราฟ ที่ 3 จำนวนผู้เสียหายในประเทศไทย

Data Quality Monitor

- ความถูกต้อง
- ความครบถ้วน
- ความเป็นปัจจุบัน

Softnix Data Platform Processes



Data Catalog



Version
2.3.0

TIME INTERVAL SELECTED
Off

DATE SELECTED
Today

Detail model / crime

Data Model Data Source Data Profile

Detail of model

Model name: crime
Project: Search
Model description:
Category:
Data access levels: public
Spatial coverage:
Update frequency:
Contact name:
Contact email:
Created by: Administrator

คุณสมบัติของข้อมูล

Detail of data store

Data store name: Hadoop
Data store description:
Host: 172.17.78.74
Port: 8090
Catalog: hive
Username: admin
Created by: Administrator

ระบบจัดเก็บ
ข้อมูล

Detail of selected field

Search filter date by: None

Field name	Type
address	varchar
birthdate	varchar
complain_date	varchar
crime_case	varchar
crime_number	varchar
name	varchar
id	varchar
surname	varchar
fullname	varchar
nation	varchar

คำอธิบายข้อมูล

Description

ที่อยู่

วันเดือนปีเกิด

วันเดือนปีที่ต้องคืน

หมายเลขอรบประชาชน

นามสกุล

ชื่อนามสกุล

สัญชาติ

Data Quality Management

Version 2.3.0

TIME INTERVAL SELECTED Off DATE SELECTED Today

Detail model / crime

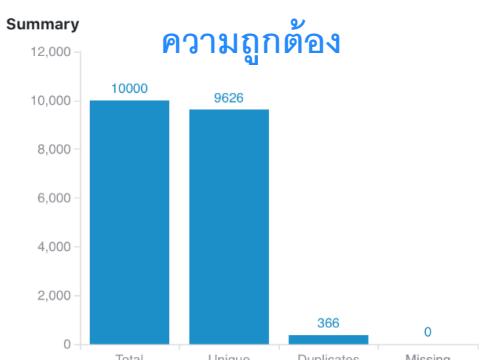
Data Model Data Source Data Profile

Fields

- address
- birthdate
- complain_date
- crime_case
- crime_number
- name
- id
- surname
- fullname
- nation
- org
- title
- want_d
- warrant_no

Summary

ความถูกต้อง



Relative Statistics

Type	Percentage
Duplicates	3.80 %
Unique	96.20 %
Nulls	0.00 %

Top Value

Value	Count
พ.๐๗๗๗/๙๕๕๗	3
พ.๑๖๖๔/๙๕๕๗	3
พ.๔๐๘๗/๙๕๕๗	3
พ.๖๐๐๗/๙๕๕๗	3
พ.๒๔๒๔/๙๕๕๗	3
พ.๙๔๔๔/๙๕๕๗	3
พ.๒๖๖๗/๙๕๕๗	3
พ.๔๕๗๔/๙๕๕๗	3
พ.๕๓๐๔/๙๕๕๗	2
พ.๔๔๔๔/๙๕๕๗	2

Detail drill down by chart

address	birthdate	complain_date	crime_case
36/722 อาพ雍 บ้านเก่าขี้ด ศรีนิพัทธ์ 32230	24/07/2509	10/06/2557	ชูเชิงว่าจะก่อการร้าย หรือครอบครองสิ่งของ他人โดยไม่ได้รับอนุญาต
80/56 ศรีสังข์ บ้านเก่าขี้ด นราธิวาส 30230	06/04/2494	29/12/2557	เสพสุรยาเสนาคนองคงสติไม่ได้ในที่สาธารณะ
79/96 ห้องล่ามเจ้าเมืองเชียงใหม่ นครศรีธรรมราช 80190	26/07/2493	20/01/2554	ทะเลาะวิวาทในที่สาธารณะ
195/35 บ้านหนองบัว กองหางดง เชียงใหม่ 50230	23/01/2503	16/04/2554	พกพาอาวุธในที่สาธารณะ
284/10 โสนล้อ บ้านหนองบัวทอง บ้านทุ่ง 11110	17/11/2503	23/04/2560	ขับรถโดยประมาท
145/62 ทุ่งต้อม บ้านเก่าสันป่าตอง เชียงใหม่ 50120	18/12/2509	13/07/2560	ความผิดฐานหลักทรัพย์โดยไม่จ่ายหนี้
94/542 บ้านแพะ บ้านเก่าเสนา พะนังครตีกือยะ 13110	15/05/2505	27/03/2553	ต่อสู้ด้วยความเสื่อมถငุ์
44/993 บ้านกรวด บ้านเก่าบ้านกรวด บุรีรัมย์ 31180	28/07/2502	24/05/2553	ความผิดฐานหลักทรัพย์อื่นโดยไม่จ่ายหนี้
12/268 หมู่ที่ 8 บ้านหนองมหาด อุตรดิตถ์ 41280	29/08/2511	07/01/2551	พกพาอาวุธในที่สาธารณะ
52/676 ทับช้าง บ้านเก่าสอดด้า จันทบุรี 22180	31/01/2509	07/08/2550	ความผิดฐานบุกรุก

crime_number

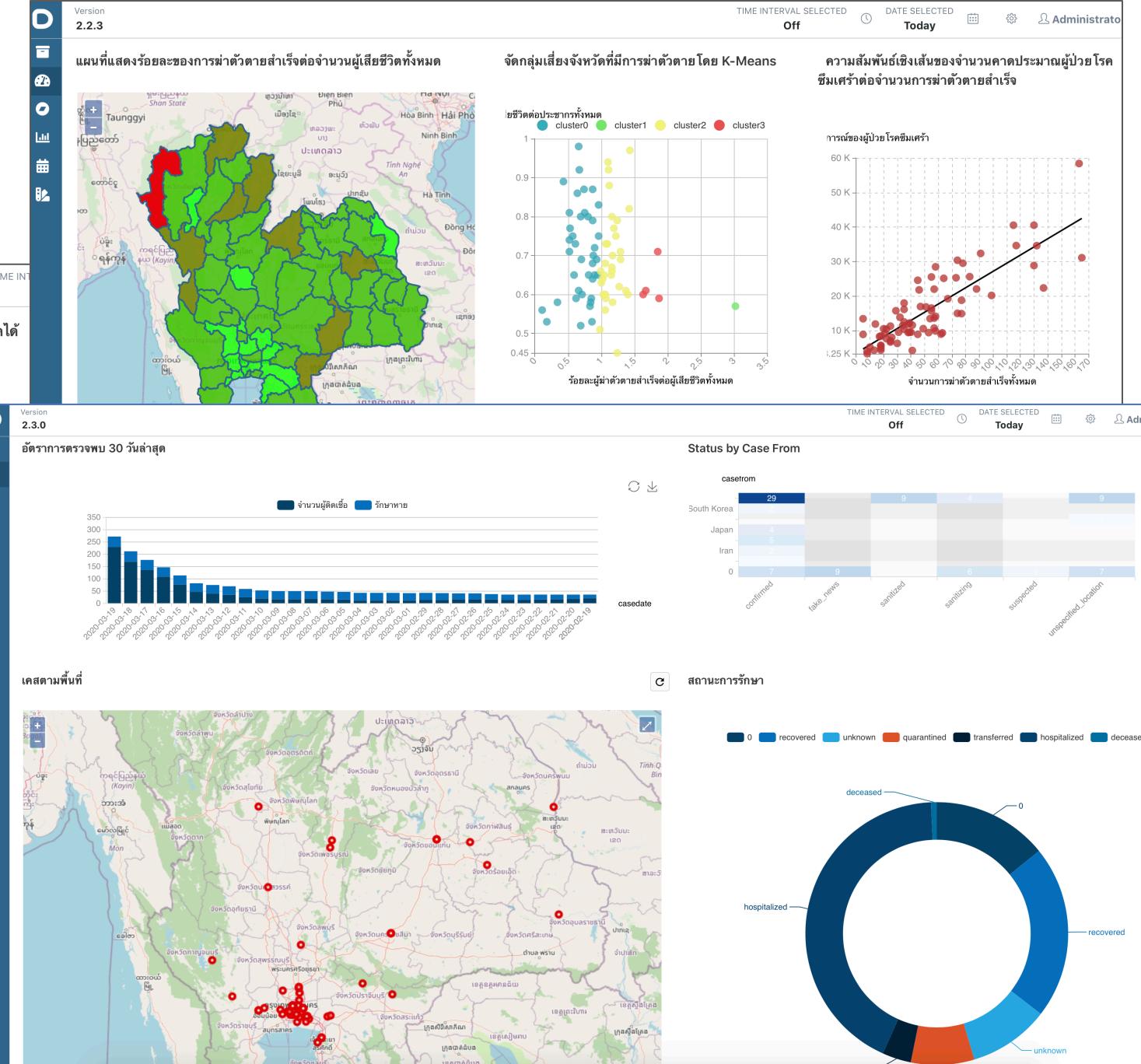
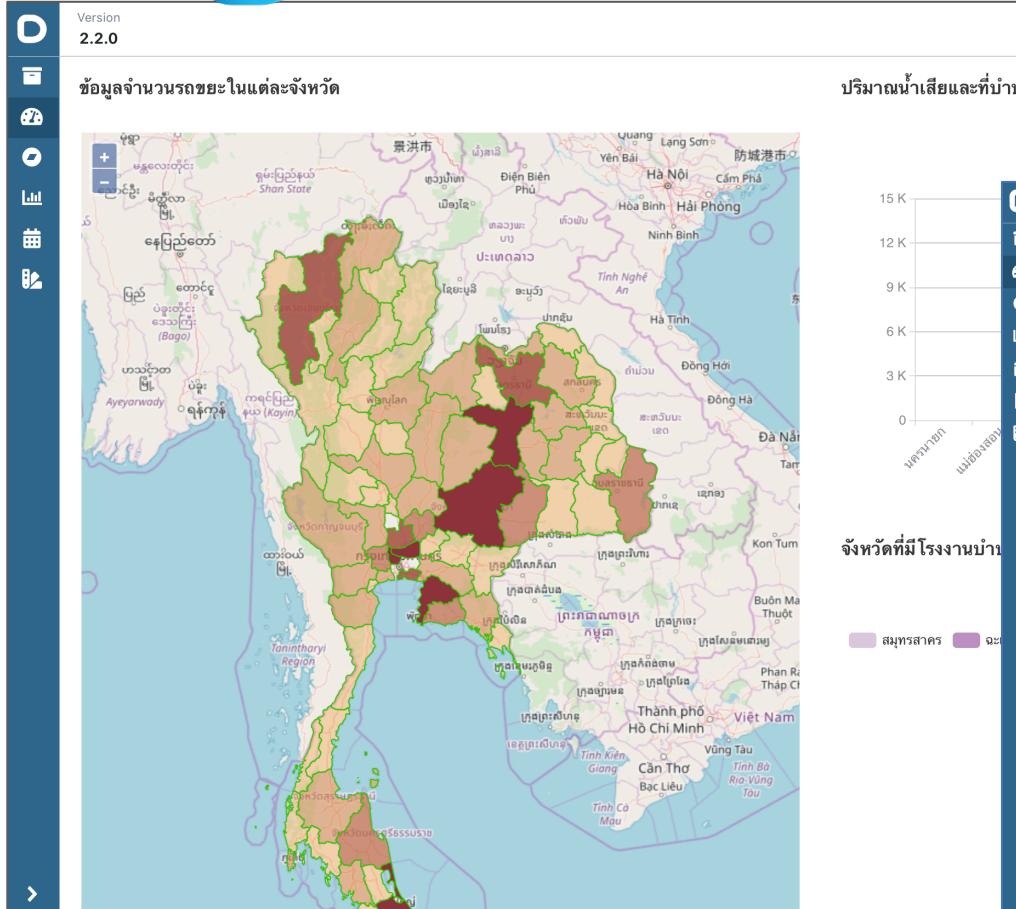
ราย พ.๔๗๗๗/๙๕๕๗ สุพ. วิธี ฟ.๔๗๗๗/๙๕๕๗

ความชำรุด

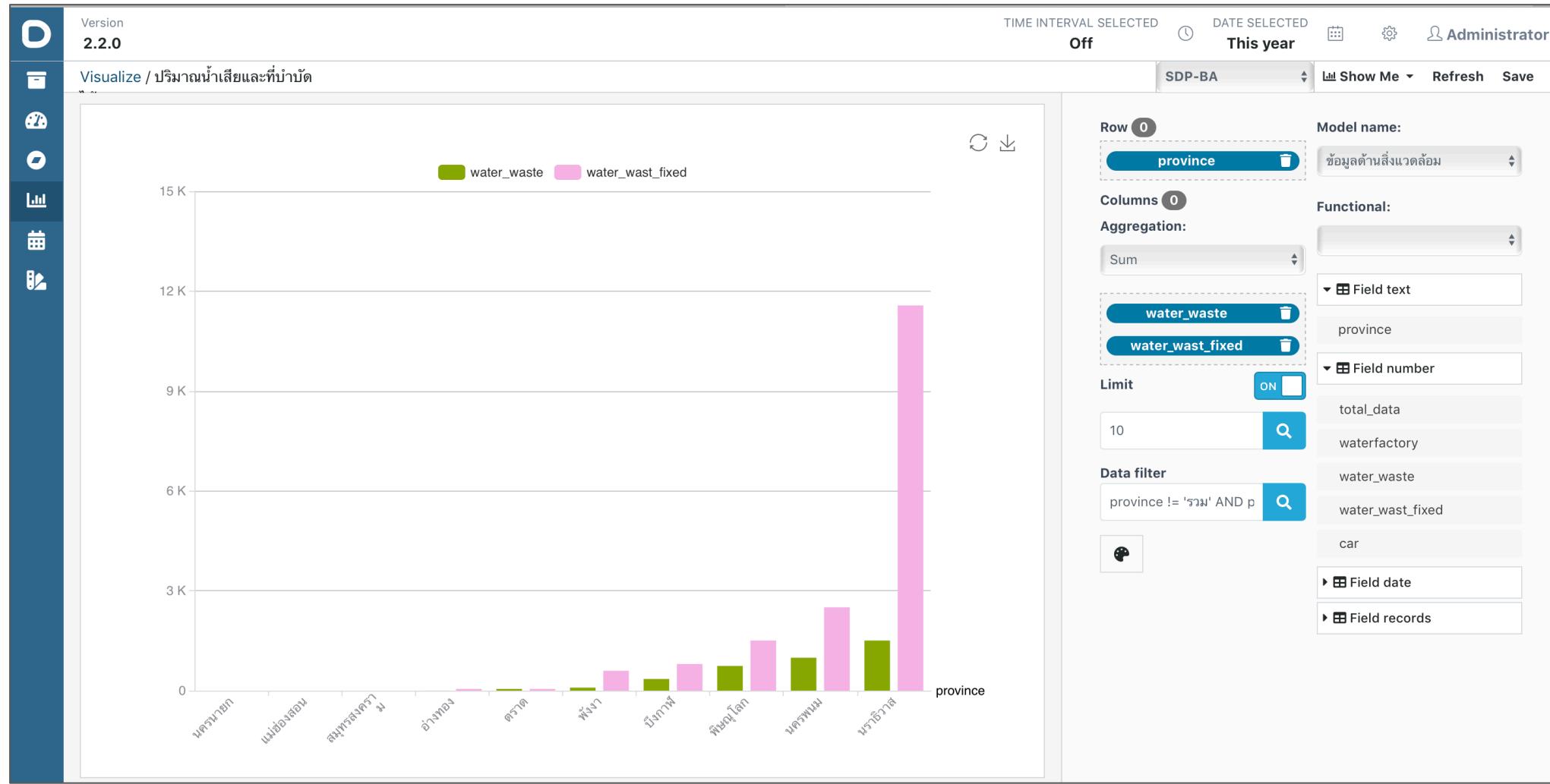
Data Visualization



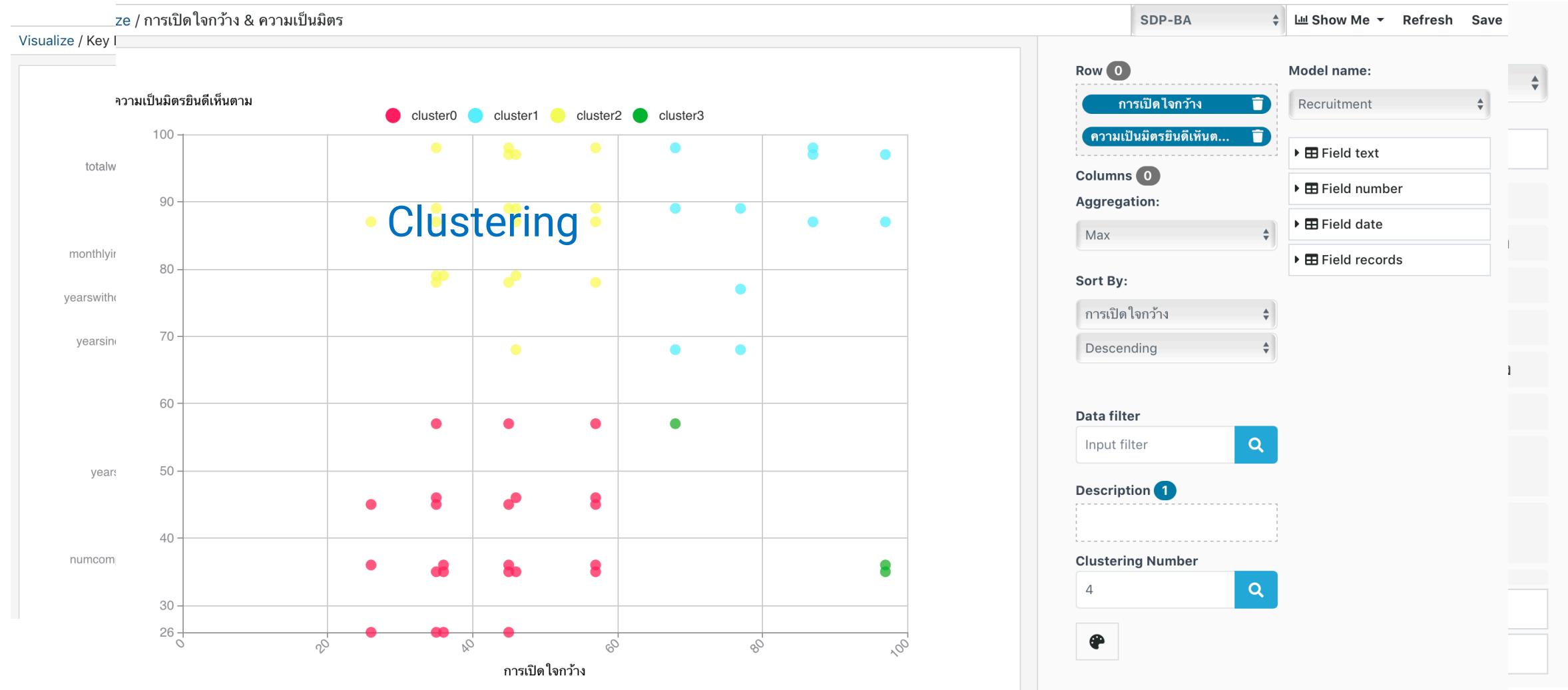
Softnix **DATA PLATFORM**



Self Services



Build-in Analytics



Geo-Location Analytics



Version 2.3.0

Visualize / เคสตามพื้นที่

TIME INTERVAL SELECTED: Off DATE SELECTED: Today Administra

lat : 14.805827
lng : 100.65233
สถานะช่วง : unspecified_location
เพศ : unknown
สถานที่ : จังหวัดพะรี
สถานการรักษา : hospitalized
เดินทางจาก : 0
คำชี้แจง : (คำชี้แจง) สำนักงานสาธารณสุขจังหวัดพะรี แจ้งพบผู้ติดเชื้อไวรัสโควิด-19 จำนวน 1 ราย
แหล่งที่มา : <https://www.facebook.com/1895451883861717/posts/3634201333320088/>

Optional Map

Layer : Marker Zoom : 7

Circle Radius : 4 Stroke Size : 4 Stroke Color : Red

Fill Color : Event Mode : On Click View : Default

C Apply

Latitude: Model name: lat COVID19 v2

Longitude: lng

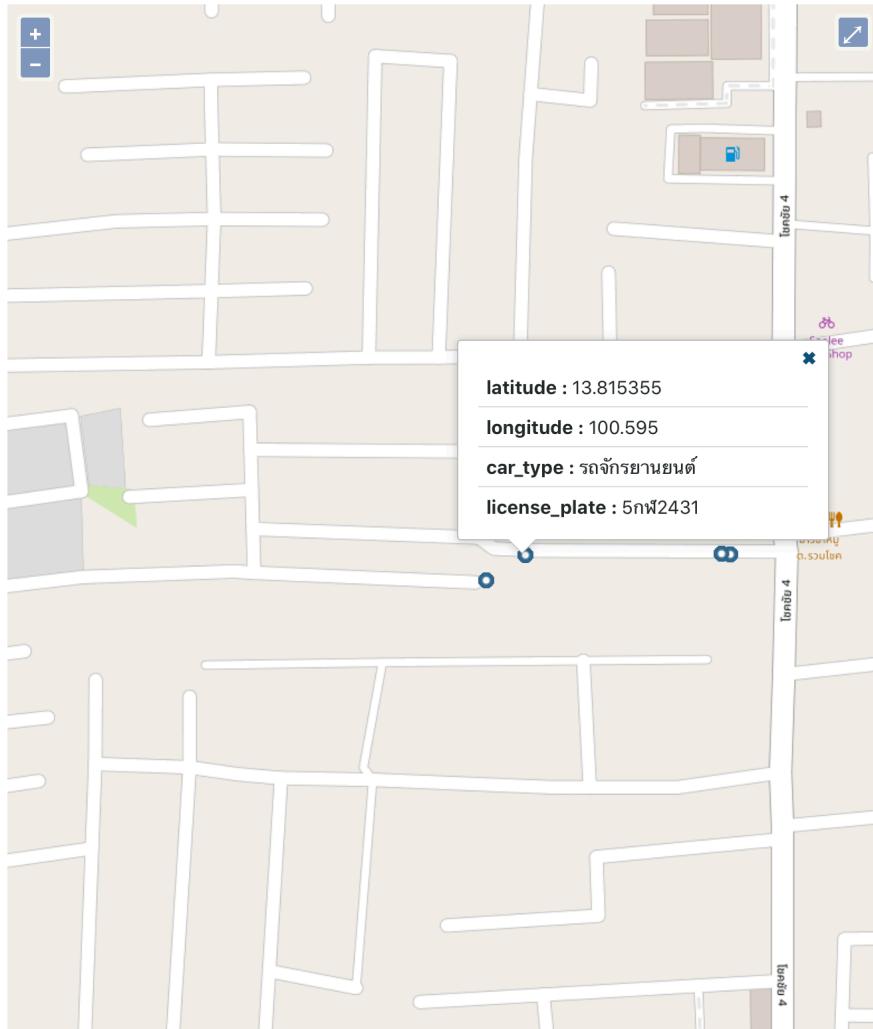
Photo:

Row 0

- สถานะช่วง
- เพศ
- สถานที่
- สถานการรักษา
- เดินทางจาก
- คำชี้แจง
- แหล่งที่มา

Columns 5 Aggregation:

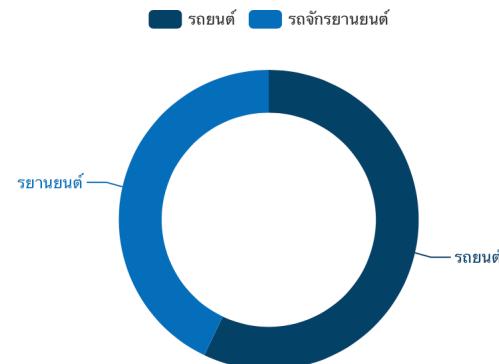
VehicleMap



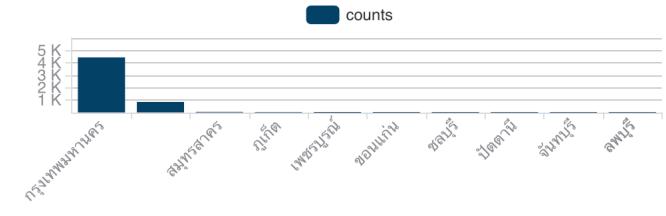
Total Car Number

5,765

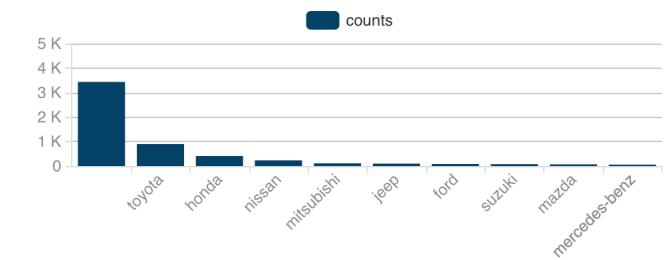
Car type Pie



Province Bar

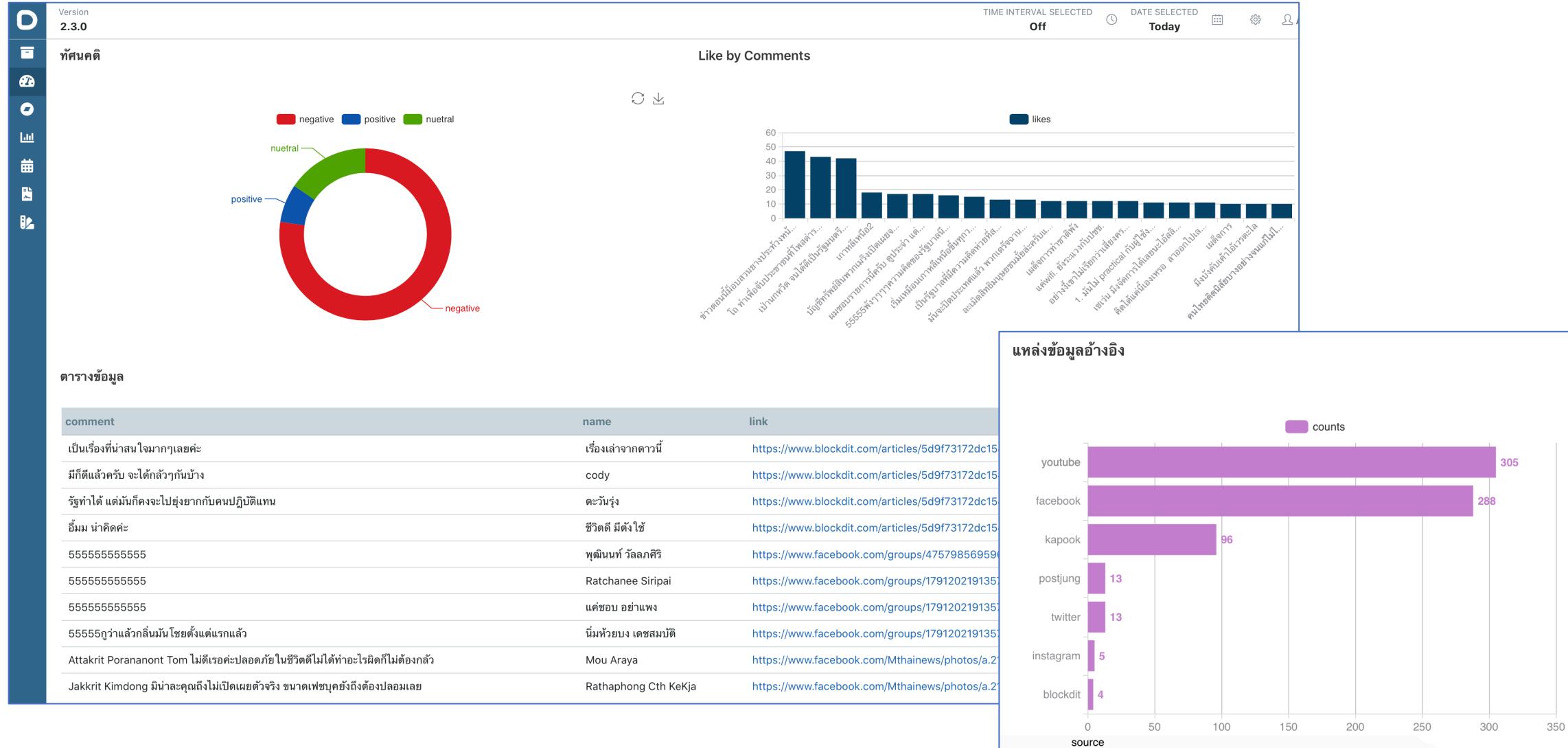


Car Brand Bar



Heat Map car type model

Social Monitor





Workshop 1

- Create Data Catalog
 - Setup Data Schema
 - Upload Data
 - Create Data Catalog
 - Update Data
 - Management Data Catalog

Primitive type	Description	Example
TINYINT	It has 1 byte, from -128 to 127. The postfix is Y. It is used as a small range of numbers.	10Y
SMALLINT	It has 2 bytes, from -32,768 to 32,767. The postfix is S. It is used as a regular descriptive number.	10S
INT	It has 4 bytes, from -2,147,483,648 to 2,147,483,647.	10
BIGINT	It has 8 bytes, from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807. The postfix is L.	100L
FLOAT	This is a 4 byte single-precision floating-point number, from 1.40129846432481707e ⁻⁴⁵ to 3.40282346638528860e ⁺³⁸ (positive or negative). Scientific notation is not yet supported. It stores very close approximations of numeric values.	1.2345679

Data Type

DOUBLE	This is an 8 byte double-precision floating-point number, from $4.94065645841246544e^{-324d}$ to $1.79769313486231570e^{+308d}$ (positive or negative). Scientific notation is not yet supported. It stores very close approximations of numeric values.	1.2345678901234567
BINARY	This was introduced in Hive 0.8.0 and only supports CAST to STRING and vice versa.	1011
BOOLEAN	This is a TRUE or FALSE value.	TRUE
STRING	This includes characters expressed with either single quotes ('') or double quotes (""). Hive uses C-style escaping within the strings. The max size is around 2 G.	'Books' or "Books"
CHAR	This is available starting with Hive 0.13.0. Most UDF will work for this type after Hive 0.14.0. The maximum length is fixed at 255.	'US' or "US"
VARCHAR	This is available starting with Hive 0.12.0. Most UDF will work for this type after Hive 0.14.0. The maximum length is fixed at 65,355. If a string value being converted/assigned to a varchar value exceeds the length specified, the string is silently truncated.	'Books' or "Books"

Data Type

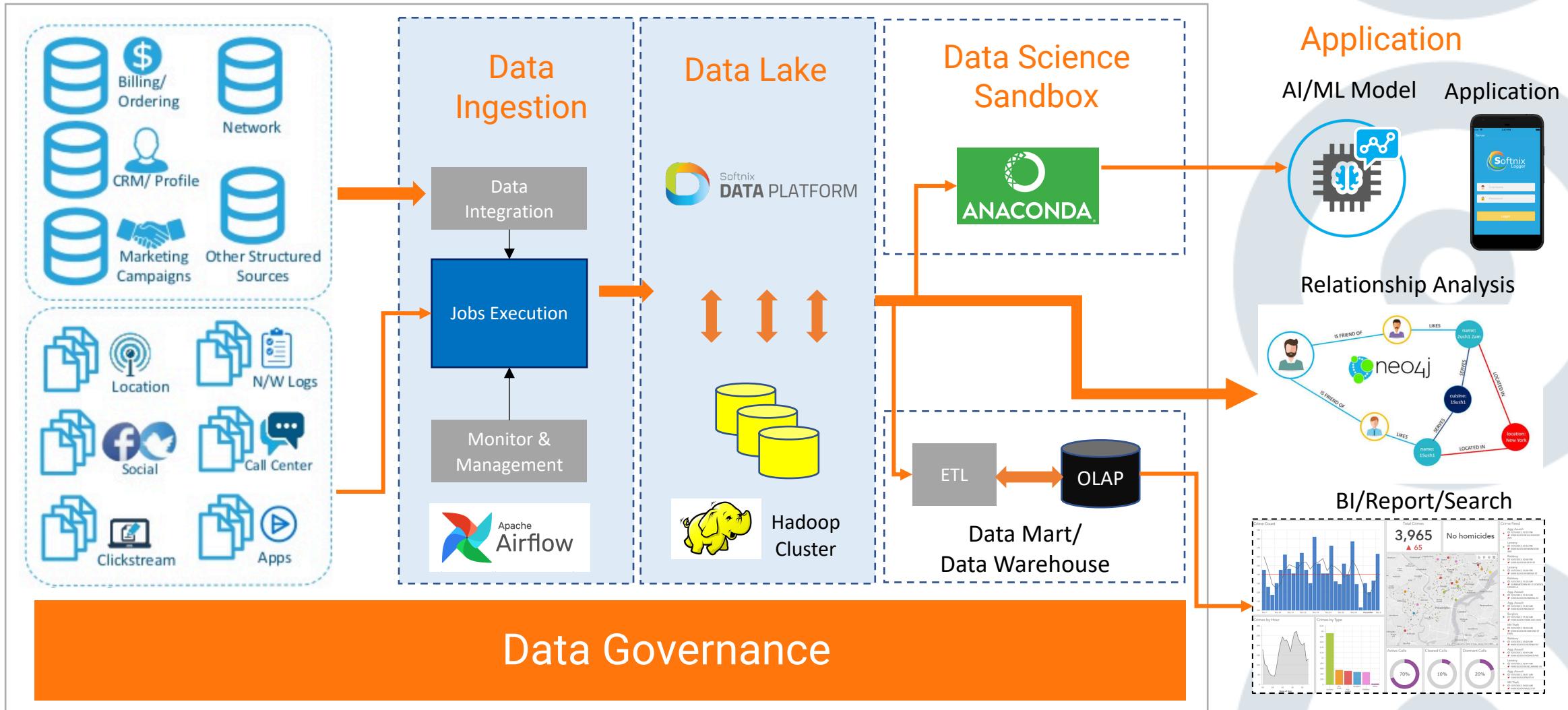
DATE	<p>This describes a specific year, month, and day in the format of YYYY-MM-DD. It is available starting with Hive 0.12.0. The range of dates is from 0000-01-01 to 9999-12-31.</p>	2013-01-01
TIMESTAMP	<p>This describes a specific year, month, day, hour, minute, second, and millisecond in the format of YYYY-MM-DD HH:MM:SS[.fff...]. It is available starting with Hive 0.8.0.</p>	2013-01-01 12:00:01.345

Data Type

Workshop 2



- Create Data Dashboard
 - Create Visualize
 - Create Dashboard
 - Management Dashboard





Hadoop

What is Apache Hadoop?



- Open source software framework designed for storage and processing of large scale data on clusters of commodity hardware
- Created by Doug Cutting and Mike Carafella in 2005.
- Cutting named the program after his son's toy elephant.

Core Hadoop Concepts

Applications are written in a high-level programming language

- No network programming or temporal dependency

Nodes should communicate as little as possible

- A “shared nothing” architecture

Data is spread among the machines in advance

- Perform computation where the data is already stored as often as possible

The Hadoop Ecosystem

Hadoop Common

- Contains Libraries and other modules

HDFS

- Hadoop Distributed File System

Hadoop YARN

- Yet Another Resource Negotiator

Hadoop MapReduce

- A programming model for large scale data processing



HDFS

Hadoop Distributed File System

Responsible for storing large data on the cluster, especially for low-cost commodity hardware

HDFS works best with a smaller number of large files

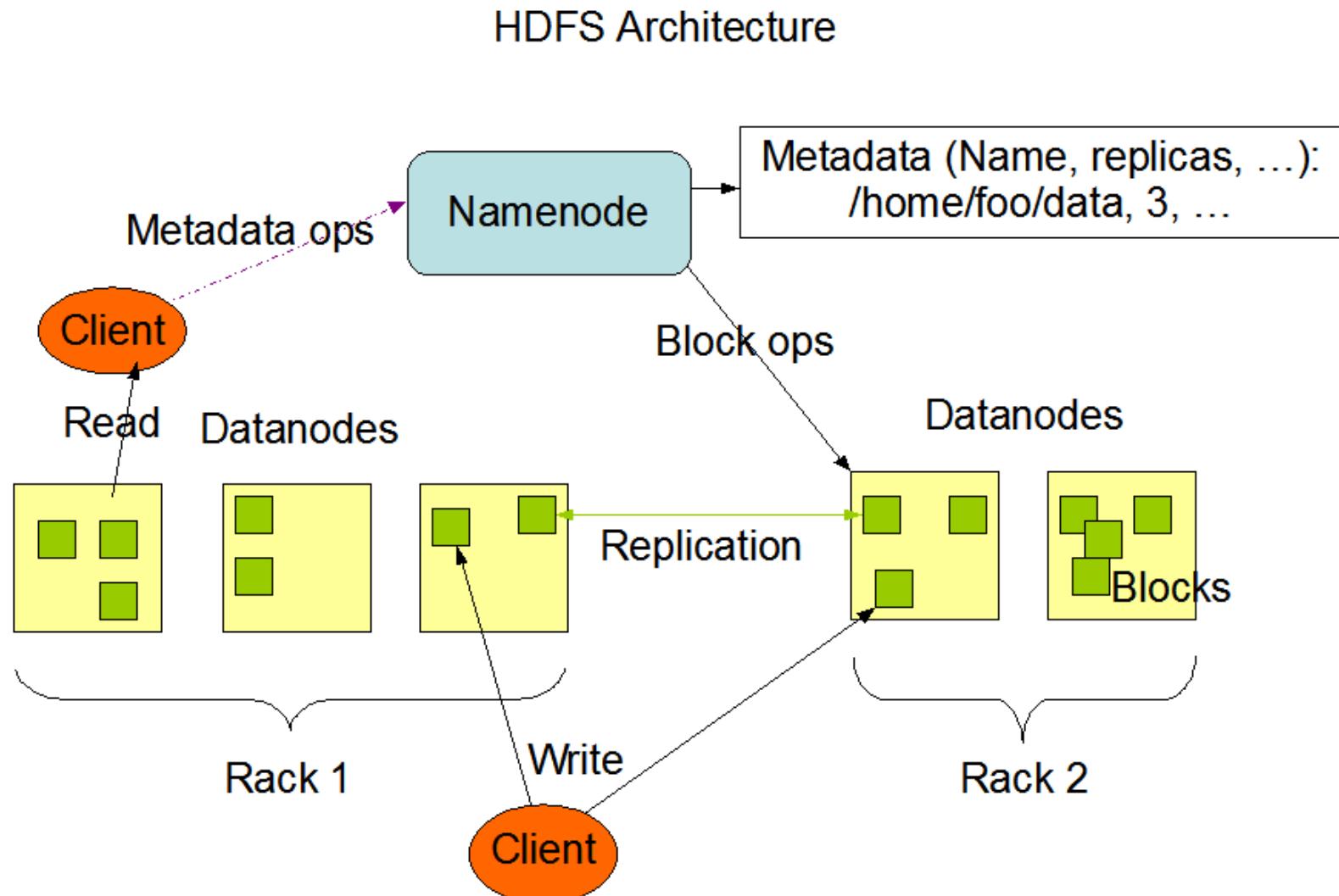
- Optimized for streaming reads of large files and not random reads
- Files in HDFS are “write-once”

Data files are split into blocks and distributed across the

- nodes in the cluster

Each block is replicated multiple times

HDFS Architecture

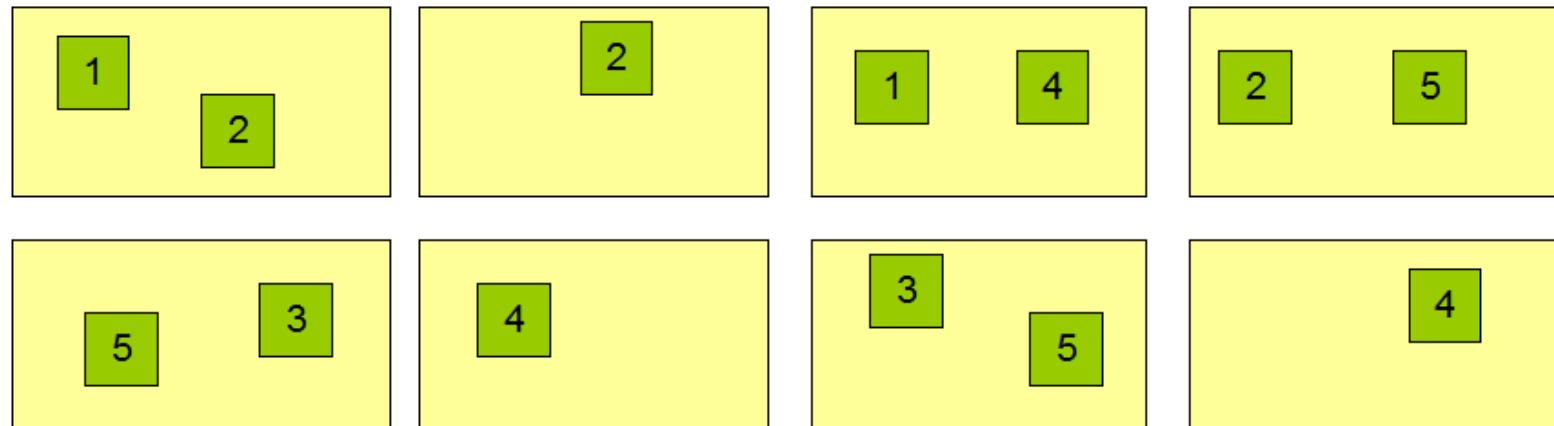


Data Replication and Fault Tolerance

Block Replication

Namenode (Filename, numReplicas, block-ids, ...)
/users/sameerp/data/part-0, r:2, {1,3}, ...
/users/sameerp/data/part-1, r:3, {2,4,5}, ...

Datanodes



Map Reduce

The Mapper

- Reads data as key/value pairs
- Outputs zero or more key/value pairs
- map: (K1,V1) ? list(K2,V2)

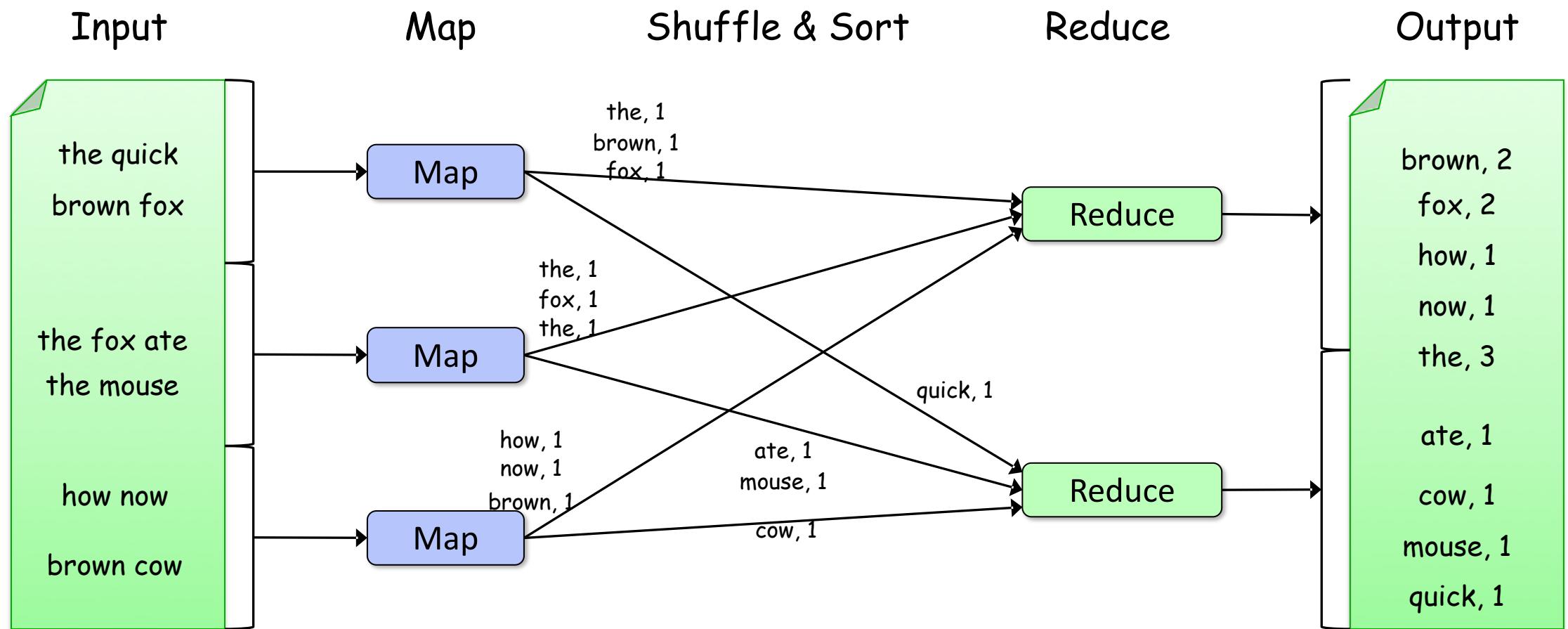
Shuffle and Sort

- Output from the mapper is sorted by key
- All values with the same key are guaranteed to go to the same machine

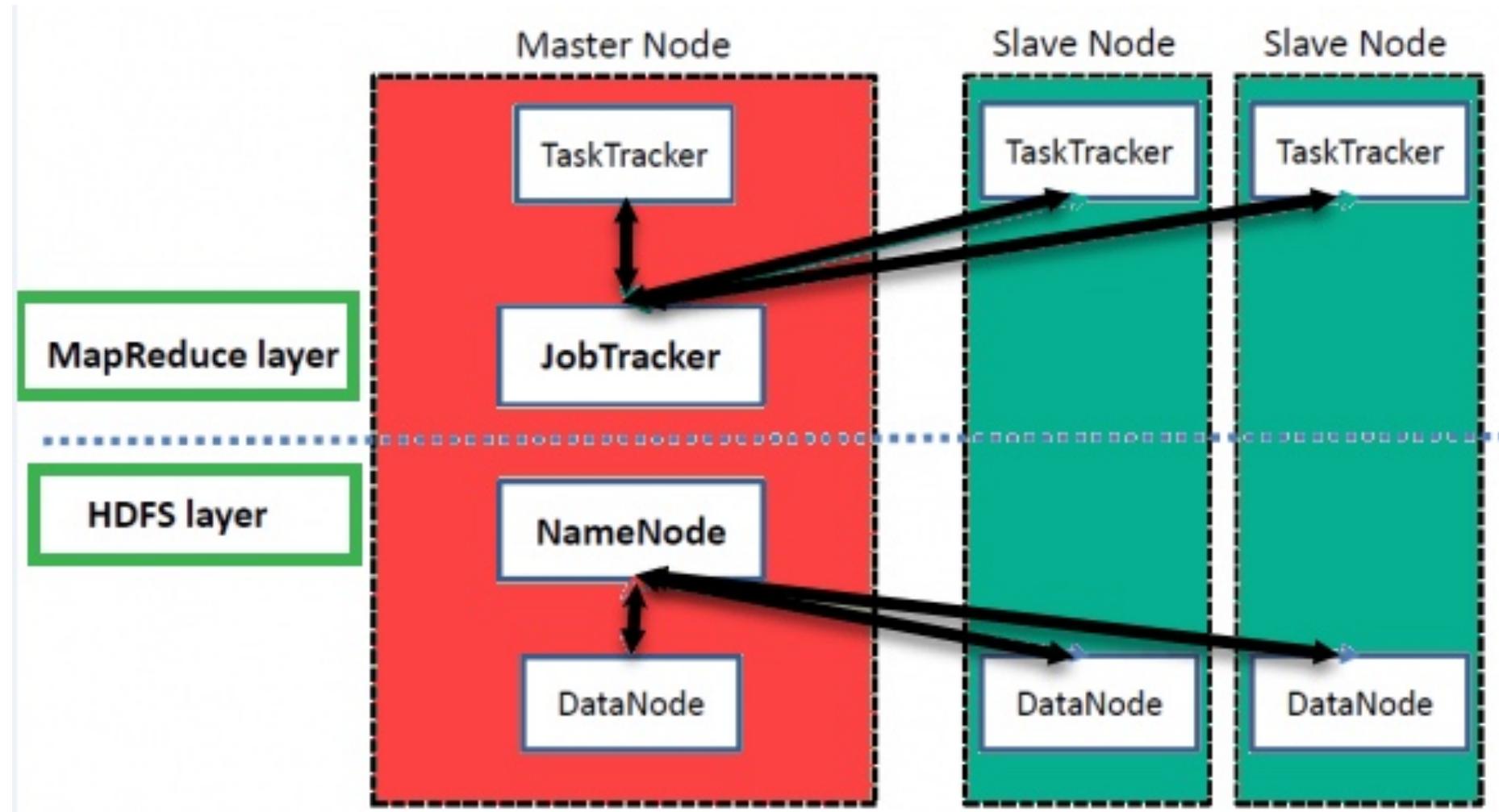
The Reducer

- Gets a list of all values associated with a key as input
 - Starts after all maps finish
 - Reducers operating on different keys can be executed simultaneously
- The reducer outputs zero or more final key/value pairs
 - Usually just one output per input key
- reduce: (K2,list(V2)) ? list(K3,V3)

Word Count Map Reduce



Architecture of Hadoop

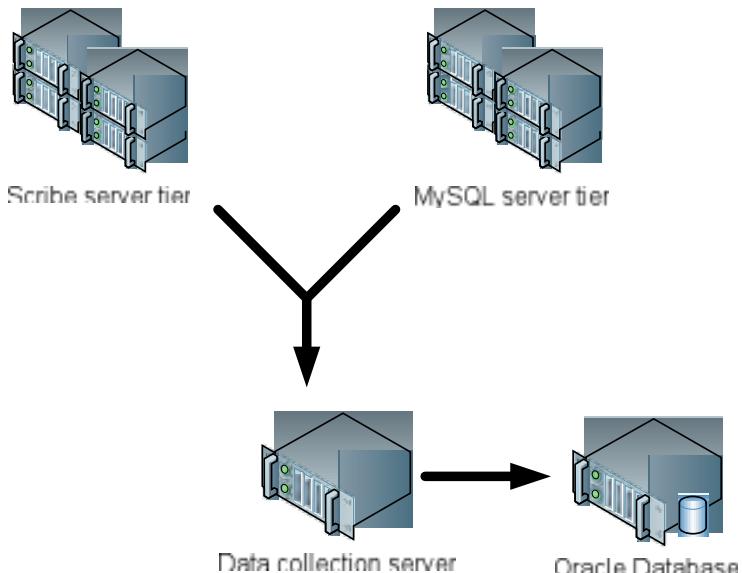




HIVE

Background

- Started at Facebook
- Data was collected by nightly cron jobs into Oracle DB
- “ETL” via hand-coded python
- Grew from 10s of GBs (2006) to 1 TB/day new data (2007), now 10x that.



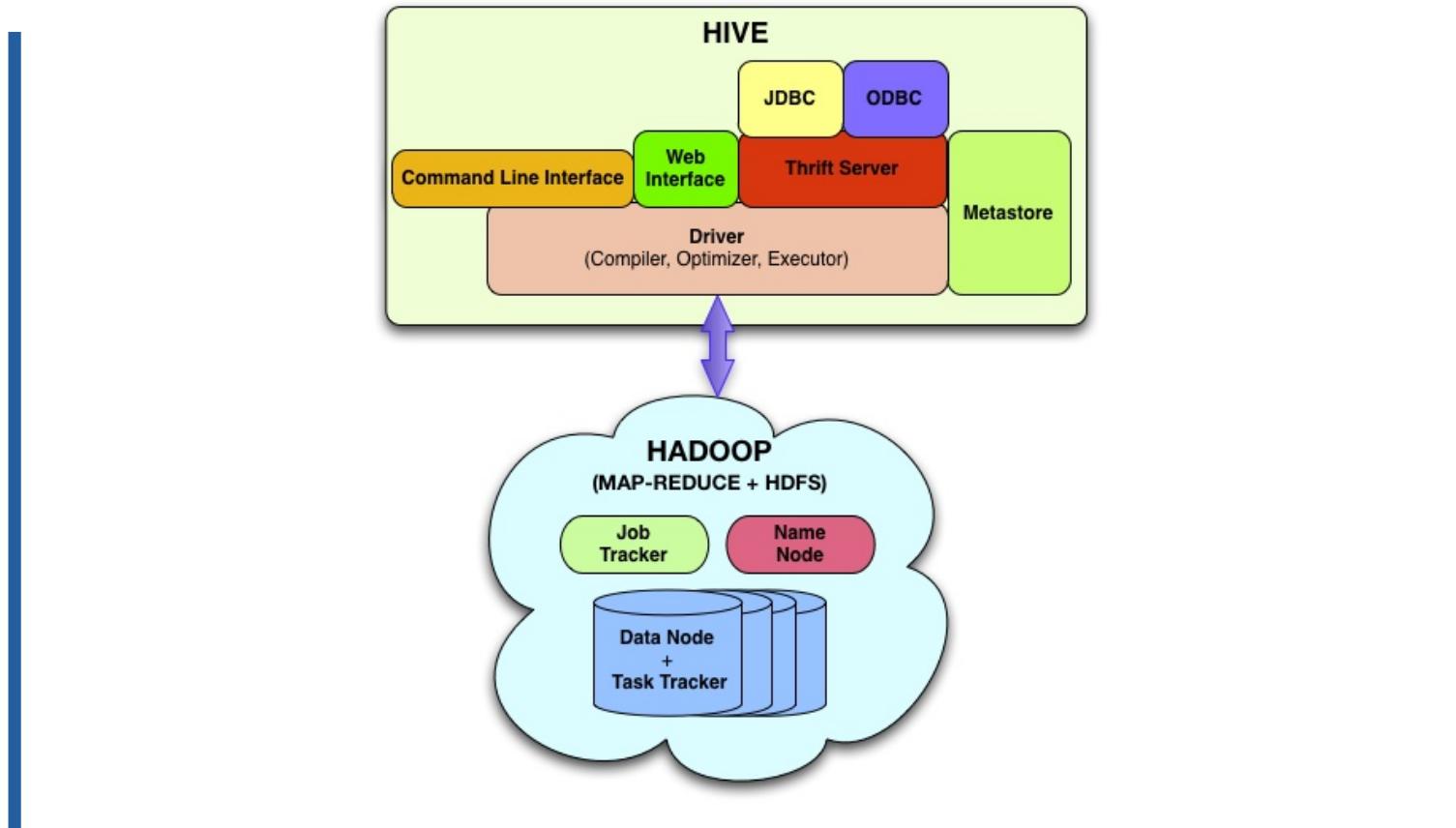
Hadoop as Enterprise Data Warehouse

- Scribe and MySQL data loaded into Hadoop HDFS
- Hadoop MapReduce jobs to process data
- Missing components:
 - Command-line interface for “end users”
 - Ad-hoc query support
 - ... without writing full MapReduce jobs
 - Schema information

Hive Applications

- Log processing
- Text mining
- Document indexing
- Customer-facing business intelligence (e.g., Google Analytics)
- Predictive modeling, hypothesis testing

Hive Architecture



Data Model

- **Tables**
 - Typed columns (int, float, string, date, boolean)
 - Also, array/map/struct for JSON-like data
- **Partitions**
 - e.g., to range-partition tables by date
- **Buckets**
 - Hash partitions within ranges (useful for sampling, join optimization)

Column Data Types

```
CREATE TABLE t (
    s STRING,
    f FLOAT,
    a ARRAY<MAP<STRING, STRUCT<p1:INT,
    p2:INT>>);

SELECT s, f, a[0]['foobar'].p2 FROM t;
```

Metastore

- Database: namespace containing a set of tables
- Holds Table/Partition definitions (column types, mappings to HDFS directories)
- Statistics
- Implemented with DataNucleus ORM. Runs on Derby, MySQL, and many other relational databases

Physical Layout

- Warehouse directory in HDFS
 - e.g., `/user/hive/warehouse`
- Table row data stored in subdirectories of warehouse
- Partitions form subdirectories of table directories
- Actual data stored in flat files
 - Control char-delimited text, or SequenceFiles
 - With custom SerDe, can use arbitrary format

Installing Hive

From a Release Tarball:

```
$ wget http://archive.apache.org/dist/
hadoop/hive/hive-0.5.0/hive-0.5.0-
bin.tar.gz
$ tar xvzf hive-0.5.0-bin.tar.gz
$ cd hive-0.5.0-bin
$ export HIVE_HOME=$PWD
$ export PATH=$HIVE_HOME/bin:$PATH
```

Logging

- Hive uses log4j
- Log4j configuration located in \$HIVE_HOME/conf/hive-log4j.properties
- Logs are stored in /tmp/\${user.name}/hive.log

Starting the Hive CLI

- Start a terminal and run
`$ hive`
- Should see a prompt like:

`hive>`

Hive CLI Commands

- List tables:
 - `hive> show tables;`
- Describe a table:
 - `hive> describe <tablename>;`
- More information:
 - `hive> describe extended <tablename>;`

Selecting data

```
hive> SELECT * FROM <tablename> LIMIT 10;
```

```
hive> SELECT * FROM <tablename>  
      WHERE freq > 100 SORT BY freq ASC  
      LIMIT 10;
```

Manipulating Tables

- DDL operations
 - SHOW TABLES
 - CREATE TABLE
 - ALTER TABLE
 - DROP TABLE

Creating Tables in Hive

- Most straightforward:

```
CREATE TABLE foo(id INT, msg STRING);
```

- Assumes default table layout
 - Text files; fields terminated with ^A, lines terminated with \n

Changing Row Format

- Arbitrary field, record separators are possible.
e.g., CSV format:

```
CREATE TABLE foo(id INT, msg STRING)
DELIMITED FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n';
```

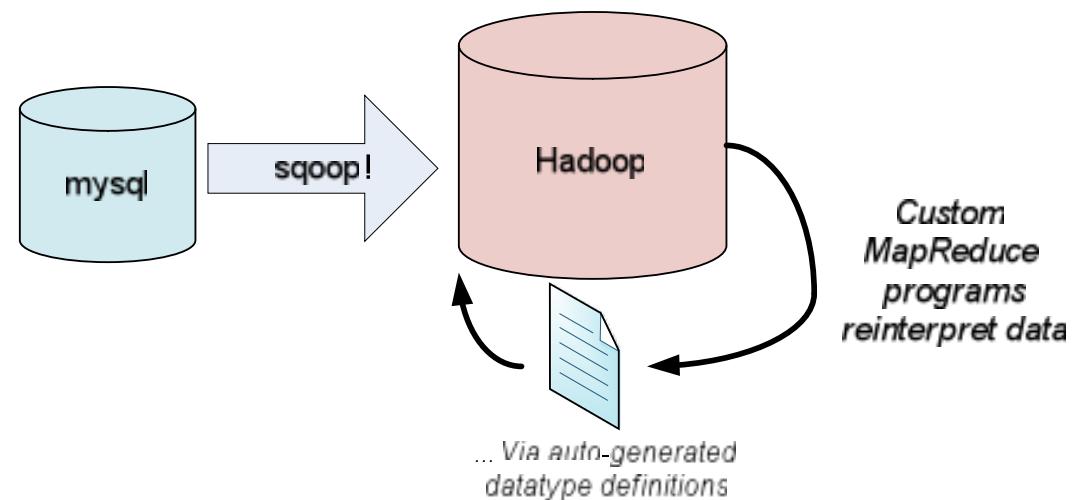
Partitioning Data

- One or more partition columns may be specified:

```
CREATE TABLE foo (id INT, msg STRING)  
PARTITIONED BY (dt STRING);
```

- Creates a subdirectory for each value of the partition column, e.g.:
`/user/hive/warehouse/foo/dt=2009-03-20/`
- Queries with partition columns in WHERE clause will scan through only a subset of the data

Sqoop = SQL-to-Hadoop



Sqoop: Features

- JDBC-based interface (MySQL, Oracle, PostgreSQL, etc...)
- Automatic datatype generation
 - Reads column info from table and generates Java classes
 - Can be used in further MapReduce processing passes
- Uses MapReduce to read tables from database
 - Can select individual table (or subset of columns)
 - Can read all tables in database
- Supports most JDBC standard types and null values

Example input

```
mysql> use corp;
Database changed

mysql> describe employees;
+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+
| id    | int(11) | NO   | PRI | NULL    | auto_increment |
| firstname | varchar(32) | YES  |     | NULL    |                |
| lastname | varchar(32) | YES  |     | NULL    |                |
| jobtitle | varchar(64) | YES  |     | NULL    |                |
| start_date | date   | YES  |     | NULL    |                |
| dept_id  | int(11) | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+
```

Loading into HDFS

```
$ sqoop --connect jdbc:mysql://db.foo.com/corp \
--table employees
```

- Imports “employees” table into HDFS directory

Hive Integration

```
$ sqoop --connect jdbc:mysql://db.foo.com/
  corp --hive-import --table employees
```

- Auto-generates CREATE TABLE / LOAD DATA INPATH statements for Hive
- After data is imported to HDFS, auto-executes Hive script
- Follow-up step: Loading into partitions

Hive Resources

Documentation

- wiki.apache.org/hadoop/Hive

Mailing Lists

- hive-user@hadoop.apache.org

IRC

- ##hive on Freenode



presto A decorative graphic consisting of a cluster of small, semi-transparent dots in shades of blue, cyan, and black, arranged in a roughly triangular or star-like pattern to the right of the word 'presto'.



What is Apache Presto?

- Apache Presto is a distributed parallel query execution engine, optimized for low latency and interactive query analysis. Presto runs queries easily and scales without down time even from gigabytes to petabytes.
- A single Presto query can process data from multiple sources like HDFS, MySQL, Cassandra, Hive and many more data sources. Presto is built in Java and easy to integrate with other data infrastructure components. Presto is powerful, and leading companies like Airbnb, DropBox, Groupon, Netflix are adopting it.

Presto – Features



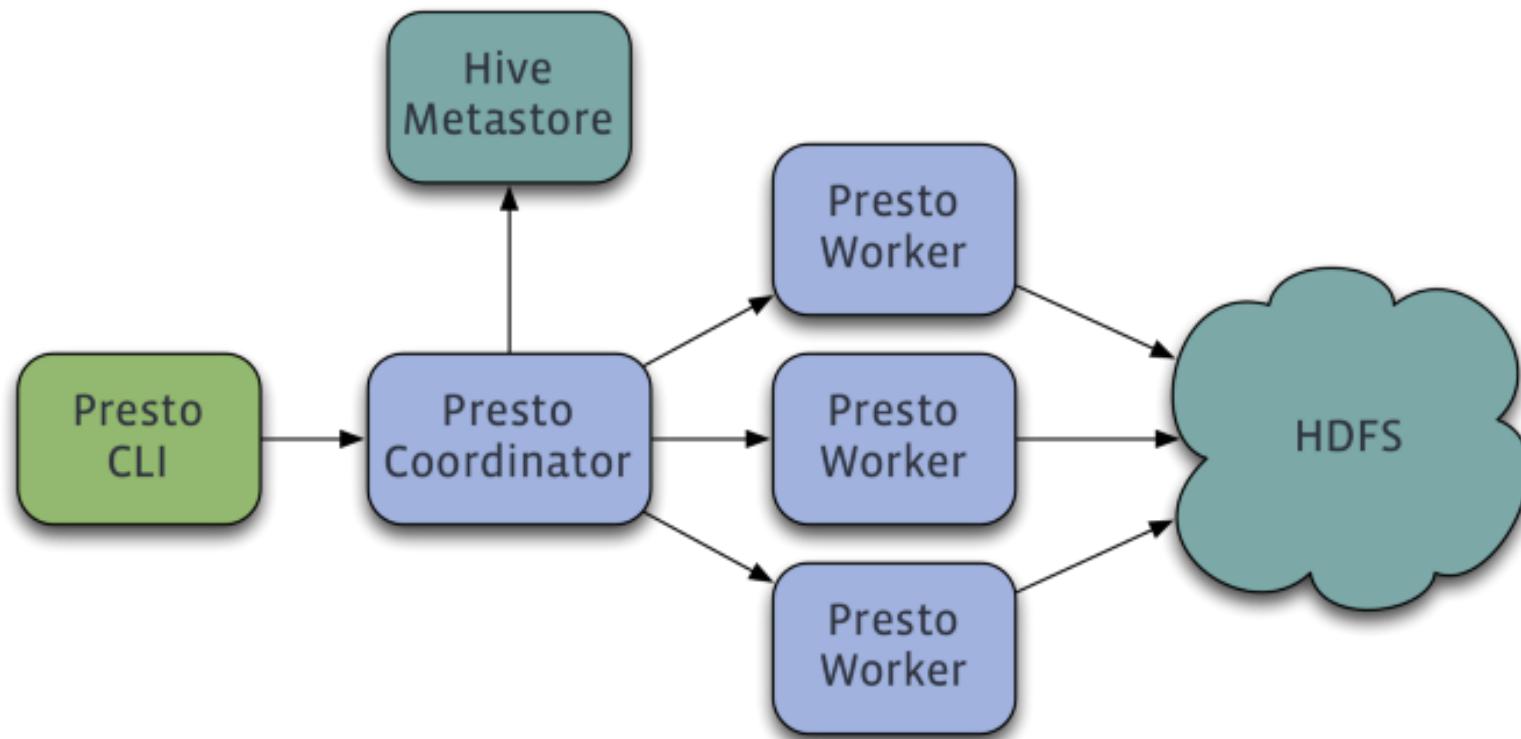
- ▶ Simple and extensible architecture.
- ▶ Pluggable connectors - Presto supports pluggable connector to provide metadata and data for queries.
- ▶ Pipelined executions -Avoids unnecessary I/O latency overhead.
- ▶ User-defined functions -Analysts can create custom user-defined functions to migrate easily.
- ▶ Vectorized columnar processing.

Presto - Benefits

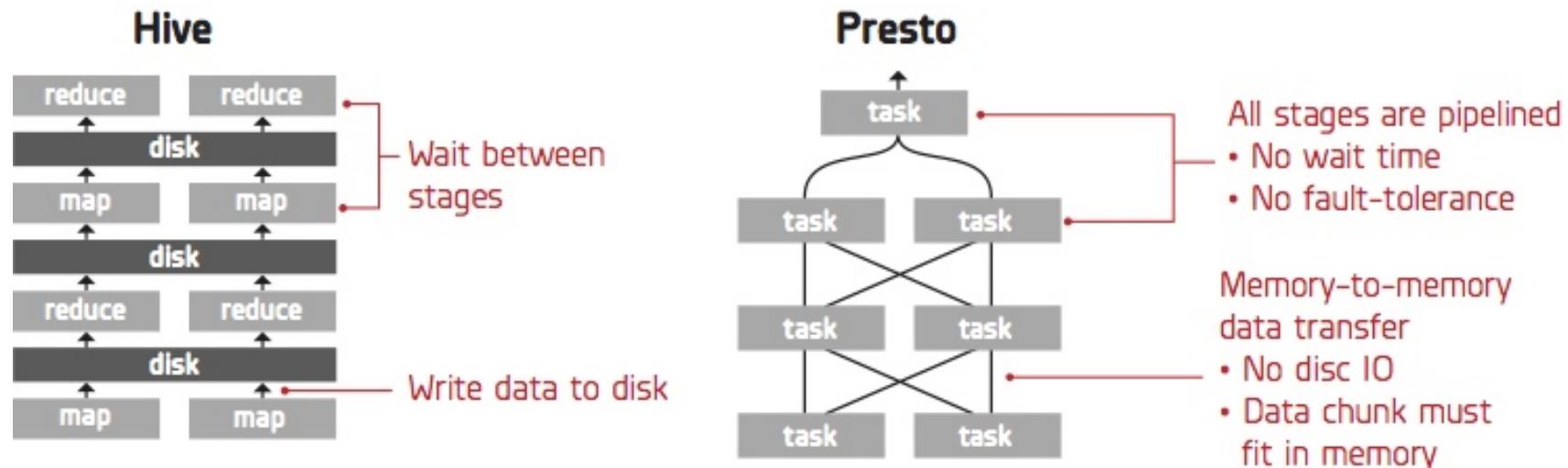


- ▶ Specialized SQL operations
- ▶ Easy to install and debug
- ▶ Simple storage abstraction
- ▶ Quickly scales petabytes data with low latency

Presto – Architecture



Hive vs Presto



Basic Python

CO Python Crash Course.ipynb ☆
PRO File Edit View Insert Runtime Tools Help Changes will not be saved

+ Code + Text Copy to Drive

Python Crash Course

This notebook will introduce some fundamental concepts of Python.

VERY IMPORTANT: you should first visit:

- This welcome: <https://colab.research.google.com/notebooks/welcome.ipynb>
- And this overview: https://colab.research.google.com/notebooks/basic_features_overview.ipynb

to get a quick overview of working with Google Colaboratory Notebooks.

```
[ ] 1 foo = 'This is a string'  
2 help(foo.upper)
```

```
▶ 1 def bar(x):  
2   'This is my help string'  
3   pass  
4 help(bar)
```

Basic data types

Python is a modern programming language that supports a range of basic, atomic data types.

Numbers

The integer numbers (e.g. 2, 4, 20) have type int, the ones with a fractional part (e.g. 5.0, 1.6) have type float. Expression syntax is straightforward: the operators +, -, *, / work just like in most other languages; parentheses () can be used for grouping. The equal sign (=) is used to assign a value to a variable.



Web Scraping

- **Data extraction is the act or process of retrieving data out of (usually unstructured or poorly structured) data sources for further data processing or data storage (data migration).**



- HTTP programming
- DOM parsing
- Text pattern matching (Regular expression)
- Etc.





HTTP Request & Response

HTTP/1.1 200 OK

Date: Mon, 23 May 2005 22:38:34 GMT

Content-Type: text/html; charset=UTF-8

Content-Encoding: UTF-8

Content-Length: 138

Last-Modified: Wed, 08 Jan 2003 23:11:55 GMT

Server: Apache/1.3.3.7 (Unix) (Red-Hat/Linux)

ETag: "3f80f-1b6-3e1cb03b"

Accept-Ranges: bytes

Connection: close

<html>

<head>

 <title>An Example Page</title>

</head>

<body> Hello World, this is a very simple HTML document. </body>

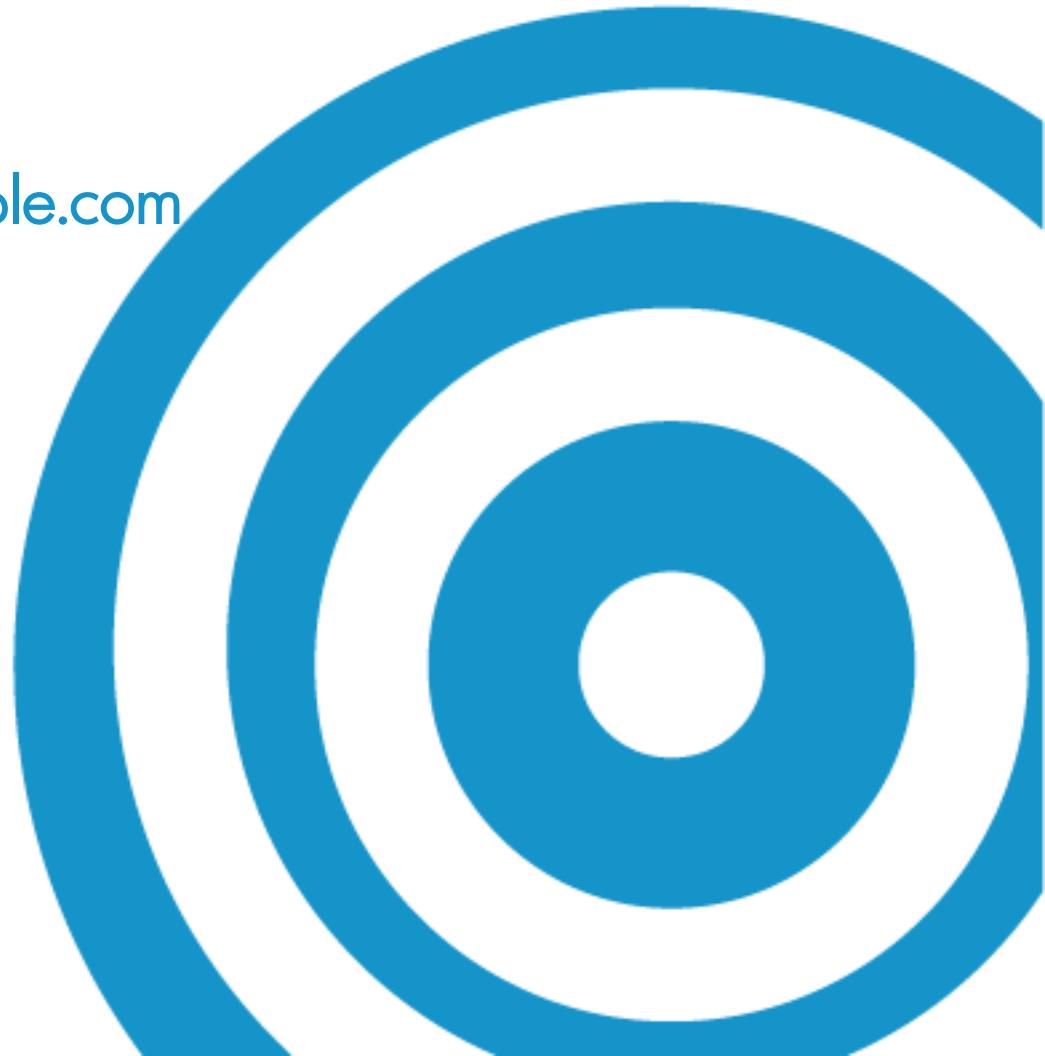
</html>

GET /index.html

HTTP/1.1

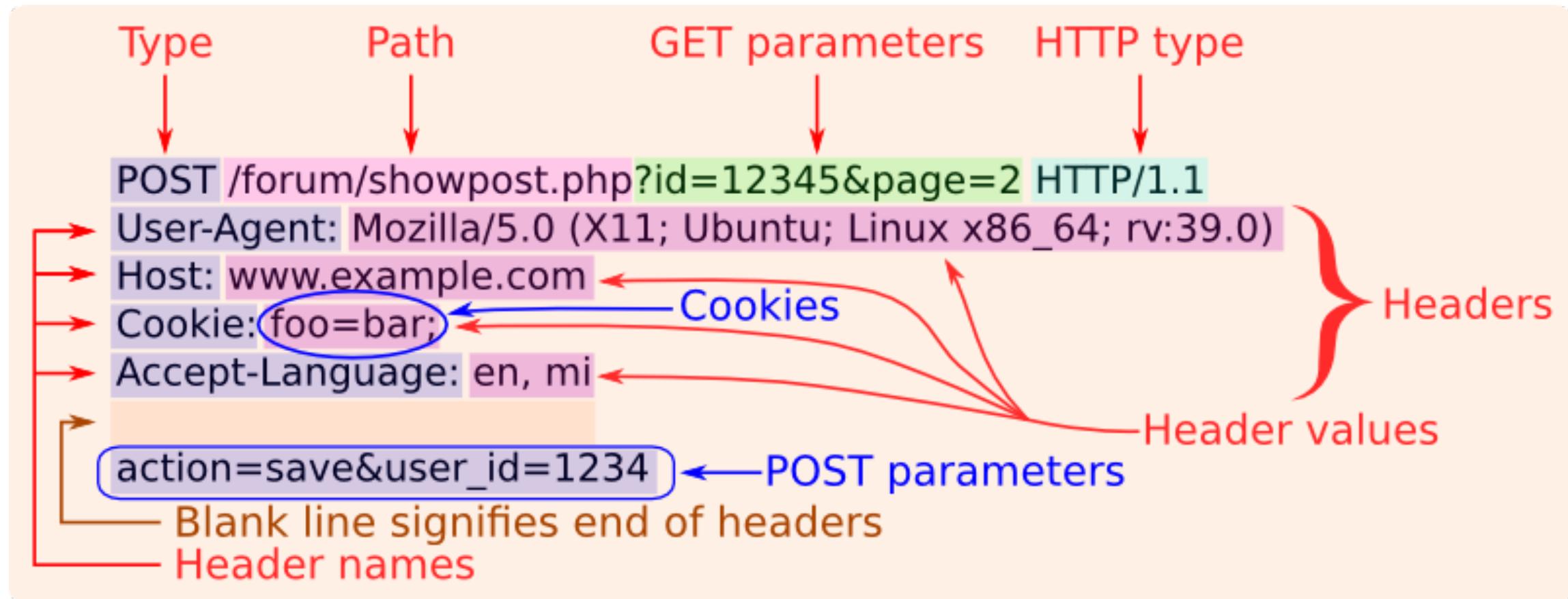
Host:

www.example.com



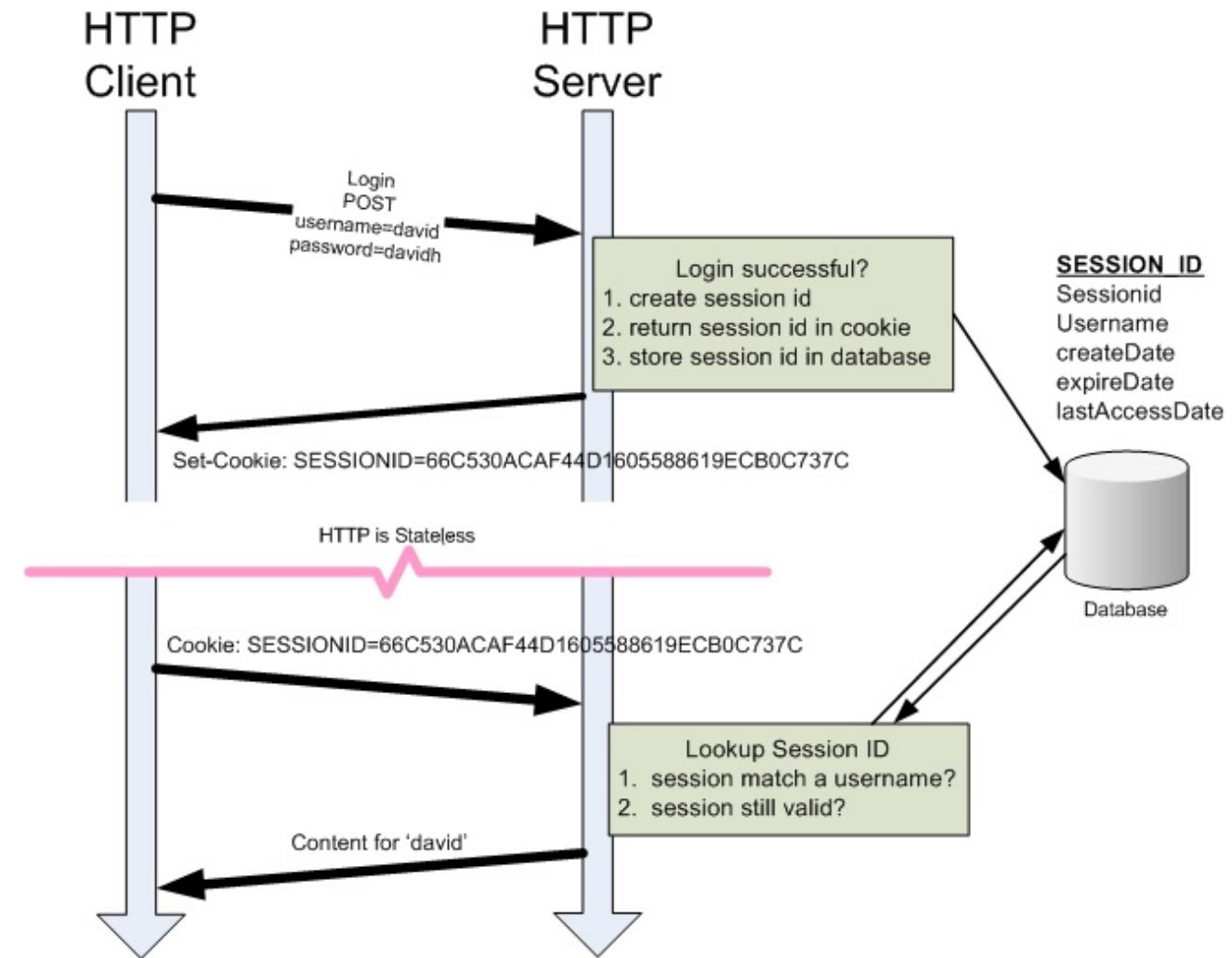


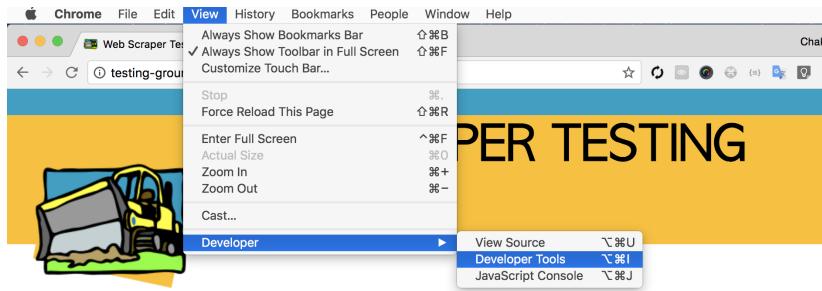
HTTP Component





Cookie & Session





LOGIN

Often in order to reach the desired information you need to be logged in to the website. Most of today's websites use so-called form-based authentication which implies sending user credentials using POST method, authenticating it on the server and storing user's session in a cookie.

This simple test shows scraper's ability to:

1. Send user credentials via POST method

The screenshot shows the Network tab of the Chrome DevTools. A single request is listed: a POST to `/login?mode=welcome`. The request details show the method as GET, status code 200 OK, and a response body containing session cookie information. The response headers include `Set-Cookie: tsess=TEST_DRIVE_SESSION`.

Developertools (Browser)

The screenshot shows the Postman application interface. The "Builder" tab is active. A test script is displayed in the main pane, showing a sequence of steps for logging in. The script includes comments explaining the process: sending user credentials via POST, receiving a session cookie, and processing an HTTP redirect. It also describes how to enter credentials and what to look for in the response to verify success or failure.

Postman

[Beautiful Soup 4.9.0 documentation »](#)

Table of Contents

Beautiful Soup Documentation

- [Getting help](#)
- [Quick Start](#)
- [Installing Beautiful Soup](#)
 - [Problems after installation](#)
 - [Installing a parser](#)
- [Making the soup](#)
- [Kinds of objects](#)
 - [`Tag`](#)
 - [Name](#)
 - [Attributes](#)
 - [Multi-valued attributes](#)
 - [`NavigableString`](#)
 - [`BeautifulSoup`](#)
 - [Comments and other special strings](#)
- [Navigating the tree](#)
 - [Going down](#)
 - [Navigating using tag names](#)
 - [.contents and .children](#)
 - [.descendants](#)
 - [.string](#)
 - [.strings and stripped_strings](#)
 - [Going up](#)
 - [.parent](#)
 - [.parents](#)
 - [Going sideways](#)
 - [.next_sibling and .previous_sibling](#)
 - [.next_siblings and .previous_siblings](#)
 - [Going back and forth](#)
 - [.next_element and .previous_element](#)
 - [.next_elements and .previous_elements](#)
- [Searching the tree](#)
 - [Kinds of filters](#)

Beautiful Soup Documentation

Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work.



These instructions illustrate all major features of Beautiful Soup 4, with examples. I show you what the library is good for, how it works, how to use it, how to make it do what you want, and what to do when it violates your expectations.

This document covers Beautiful Soup version 4.9.1. The examples in this documentation should work the same way in Python 2.7 and Python 3.2.

You might be looking for the documentation for Beautiful Soup 3. If so, you should know that Beautiful Soup 3 is no longer being developed and that support for it will be dropped on or after December 31, 2020. If you want to learn about the differences between Beautiful Soup 3 and Beautiful Soup 4, see [Porting code to BS4](#).

This documentation has been translated into other languages by Beautiful Soup users:

- 这篇文档当然还有中文版。
- このページは日本語で利用できます(外部リンク)
- 이 문서는 한국어 번역도 가능합니다.
- Este documento também está disponível em Português do Brasil.
- Эта документация доступна на русском языке.

Getting help

If you have questions about Beautiful Soup, or run into problems, send mail to the discussion group. If your problem involves parsing an HTML document, be sure to mention what the `diagnose()` function says about that document.

Quick Start



Web Scraping Demo

jupyter ราคาอ้อย scrap Last Checkpoint: 10/20/2019 (unsaved changes)

Logout Trusted Python 3

```
File Edit View Insert Cell Kernel Widgets Help
+ Run C Code
```

```
import re
import collections
import numpy as np
from requests import get
from bs4 import BeautifulSoup
from urllib.parse import quote, unquote
import deepcut
import pickle
from keras.models import Sequential, Model
from keras.utils import np_utils
from keras.layers import Embedding, Reshape, Activation, Input, Dense, GRU, LSTM, Dropout
from pythainlp.tokenize import word_tokenize

Using TensorFlow backend.
```

```
In [3]: header = []
for j in range(0,20,10):
    #print(j)
    url = "https://search.news.com/search/result?start="+str(j)+"&q=%E0%B8%A3%E0%B8%B2%E0%B8%84%E0%B8%B2%E0%B8%AD%E0%B9%80"
    r = get(url)
    content = BeautifulSoup(r.text).find_all('div', class_="detail")
    for i in content:
        header.append(i.find('a').text)
```

```
In [4]: len(header)
Out[4]: 20
```

```
In [5]: header
Out[5]: ['เศรษฐกิจ-ธุรกิจ',
          'สังคม',
          'ส่วนลดการบินชุดใหญ่ ไม่อั้นราคาน้ำดื่ม',
          'สังคม',
          'จีเคาระราคาอ้อยสูงถึงค่าผลิตอันสูงที่สุดในโลก',
          'ยินดีสหกรณ์ไทย ยกเว้นค่าธรรมเนียม',
          'ชาวยิวในรัสเซียรับราคาอ้อยหัวงั้นตันละ 900 บาท/ตัน',
          'จีรัชพลศุภาราคาอ้อย',
          'หมวดเงินสือพัน',
          'ครม. กู้เงินราคาอ้อยฯ จำนวน 881 ล้านบาท',
          'ราคาร้อย 1,050 บาท/ตัน',
          'หมวดเงินสือพัน',
          'นัดเคาะราคาอ้อย พุ่งตันละ 1,000 บาท',
          'ชาวยิวในรัสเซียหันตัน 2559/60 พัน',
          'ครม. อนุมัติซื้ออ้อยหันตัน 2 ราคาแล้ว']
```



Apache
Airflow

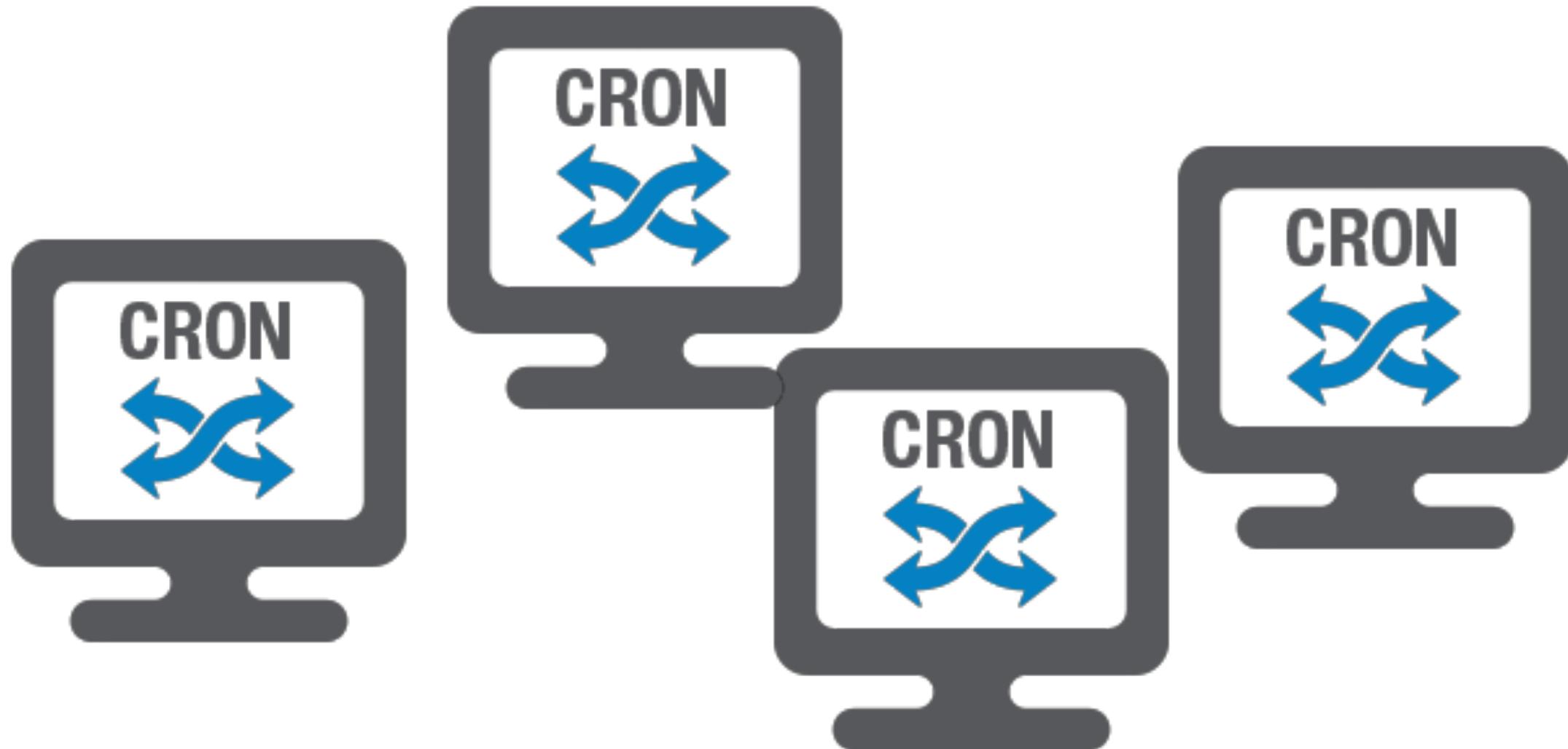
Apache
AirFlow



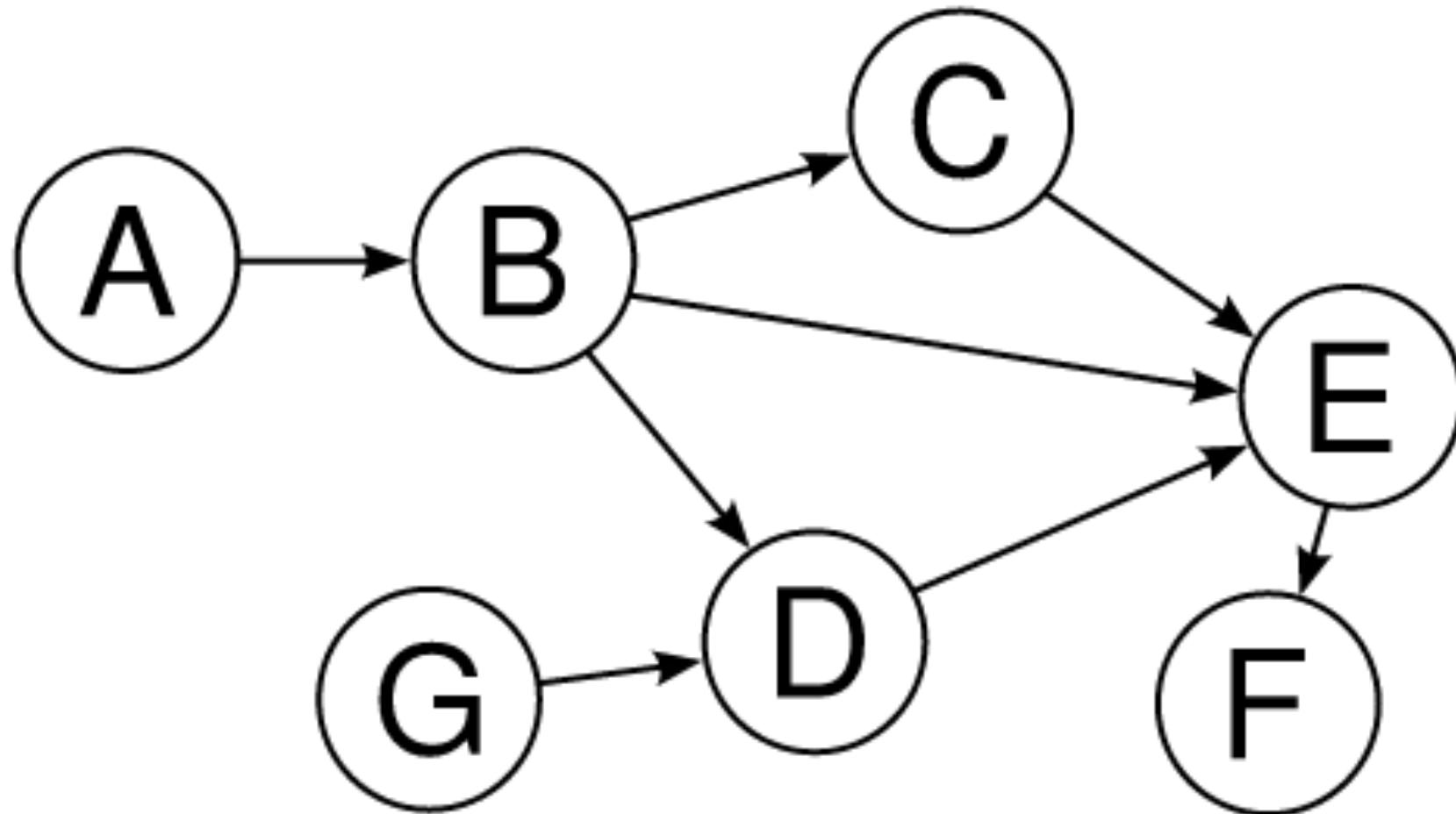


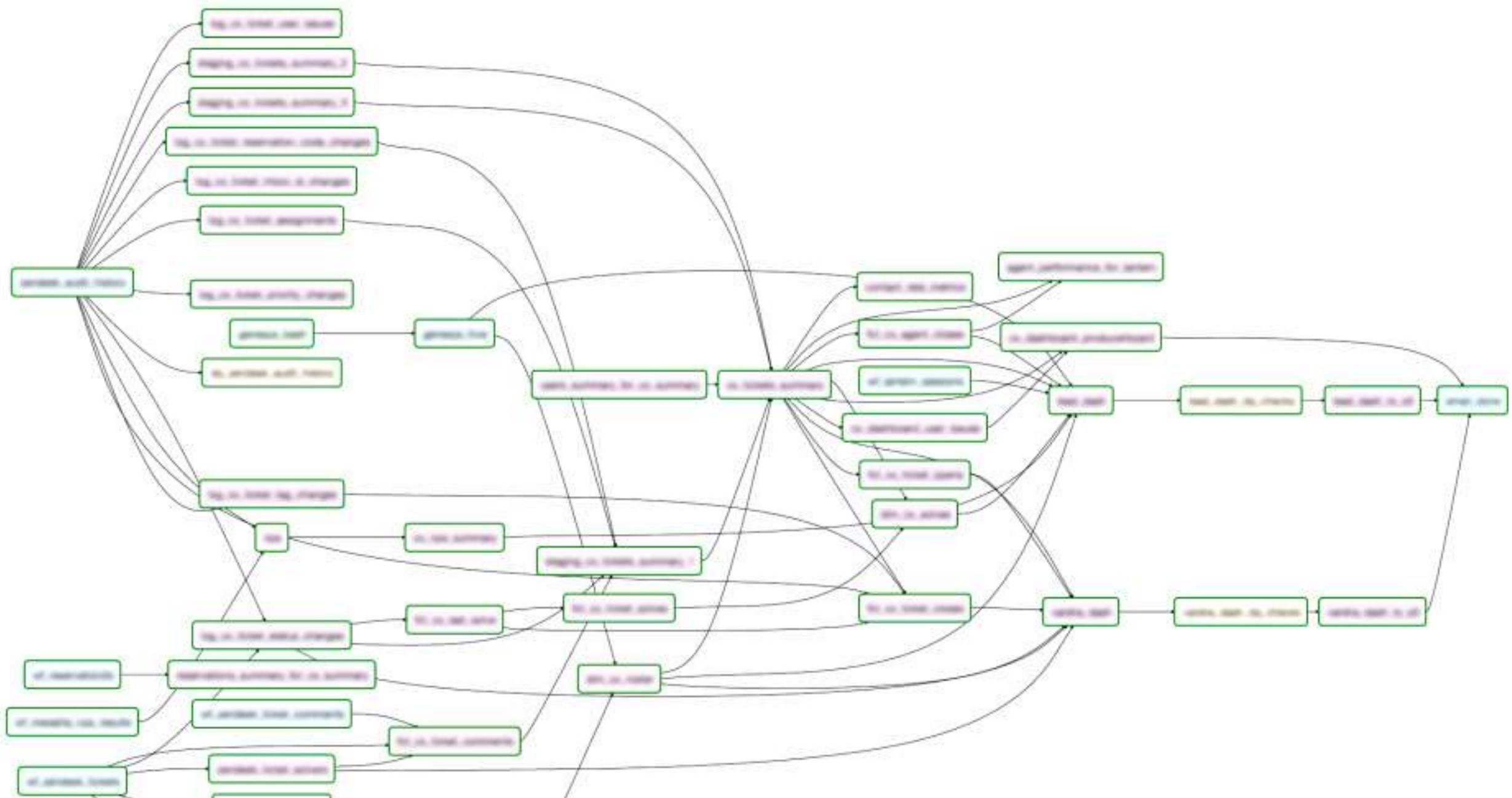
Why Data Flow Engine?

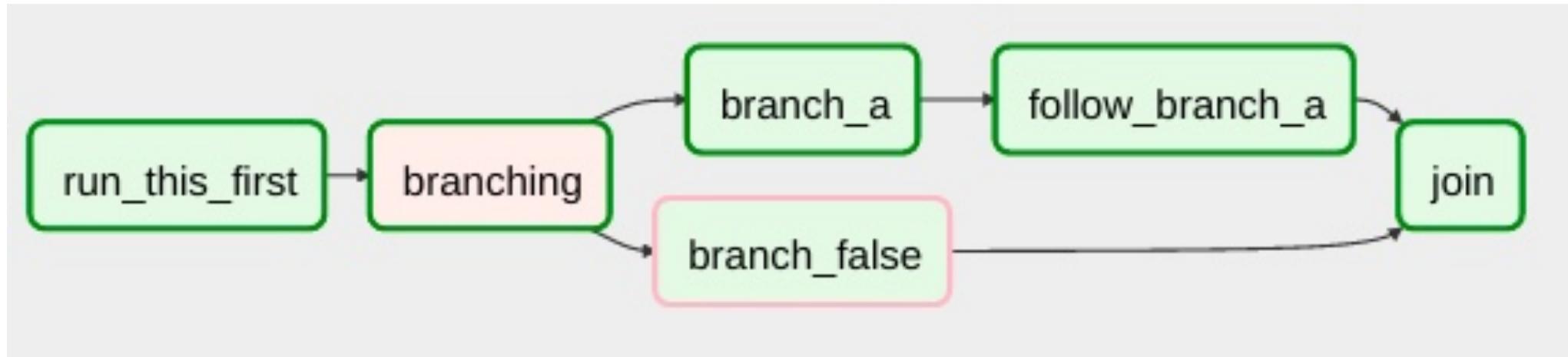
CronJob



Direct Acyclic Graph (DAG)







Data relationships

- Data availability
 - if the data is not there, trigger the process to generate the data.
- Data dependency
 - Some data relies on other data to generate.



Operability

- Job failed and resume
- Job monitor
- Backfill



Airflow





DAG structure as code

```
default_args = {
    'email': ['airflow@airflow.com'],
    'email_on_failure': False,
    'email_on_retry': False,
    'retries': 1,
    'retry_delay': timedelta(minutes=5),
}

dag = DAG('tutorial', default_args=default_args)

# t1, t2 and t3 are examples of tasks created by instantiating operators
t1 = BashOperator(task_id='print_date', bash_command='date', dag=dag)
t2 = BashOperator(task_id='sleep', bash_command='sleep 5', retries=3, dag=dag)

templated_command = """
    {% for i in range(5) %}
        echo "{{ ds }}"
        echo "{{ macros.ds_add(ds, 7)}}"
        echo "{{ params.my_param }}"
    {% endfor %}
"""

t3 = BashOperator(
    task_id='templated',
    bash_command=templated_command,
    params={'my_param': 'Parameter I passed in'},
    dag=dag)

t2.set_upstream(t1)
t3.set_upstream(t1)
```

```
default_args = {
    'owner': 'airflow',
    'depends_on_past': False,
    'start_date': datetime(2015, 6, 1),
    'email': ['airflow@airflow.com'],
    'email_on_failure': False,
    'email_on_retry': False,
    'retries': 1,
    'retry_delay': timedelta(minutes=5),
    # 'queue': 'bash_queue',
    # 'pool': 'backfill',
    # 'priority_weight': 10,
    # 'end_date': datetime(2016, 1, 1),
}
```

```
templated_command = """
    {% for i in range(5) %}
        echo "{{ ds }}"
        echo "{{ macros.ds_add(ds, 7) }}"
        echo "{{ params.my_param }}"
    {% endfor %}
"""

```

```
    {{ ds }} => now YYYY-MM-DD
    {{ yesterday_ds }} => yesterday YYYY-MM-DD
    {{ tomorrow_ds }} => tomorrow YYYY-MM-DD
    ...

```

```
dag = DAG('tutorial', default_args=default_args)

# t1, t2 and t3 are examples of tasks created by instantiating operators
t1 = BashOperator(
    task_id='print_date',
    bash_command='date',
    dag=dag)

t2 = BashOperator(
    task_id='sleep',
    bash_command='sleep 5',
    retries=3,
    dag=dag)

t3 = BashOperator(
    task_id='templated',
    bash_command=templated_command,
    params={'my_param': 'Parameter I passed in'},
    dag=dag)

t2.set_upstream(t1)
t3.set_upstream(t1)
```

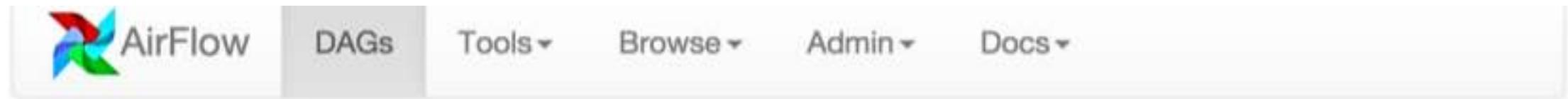


Airflow Command

- python tutorial.py
- airflow list_dags
- airflow list_taskstutorial
- airflow list_taskstutorial --tree
- airflow test tutorial print_date2015-06-01
- airflow test tutorial sleep 2015-06-01
- airflow run tutorial templated2015-06-01
- Backfill: airflow backfill tutorial -s 2015-06-07 -e 2015-06-10
 - Run again
 - Run another date range

Site: <http://localhost:8080/admin/>

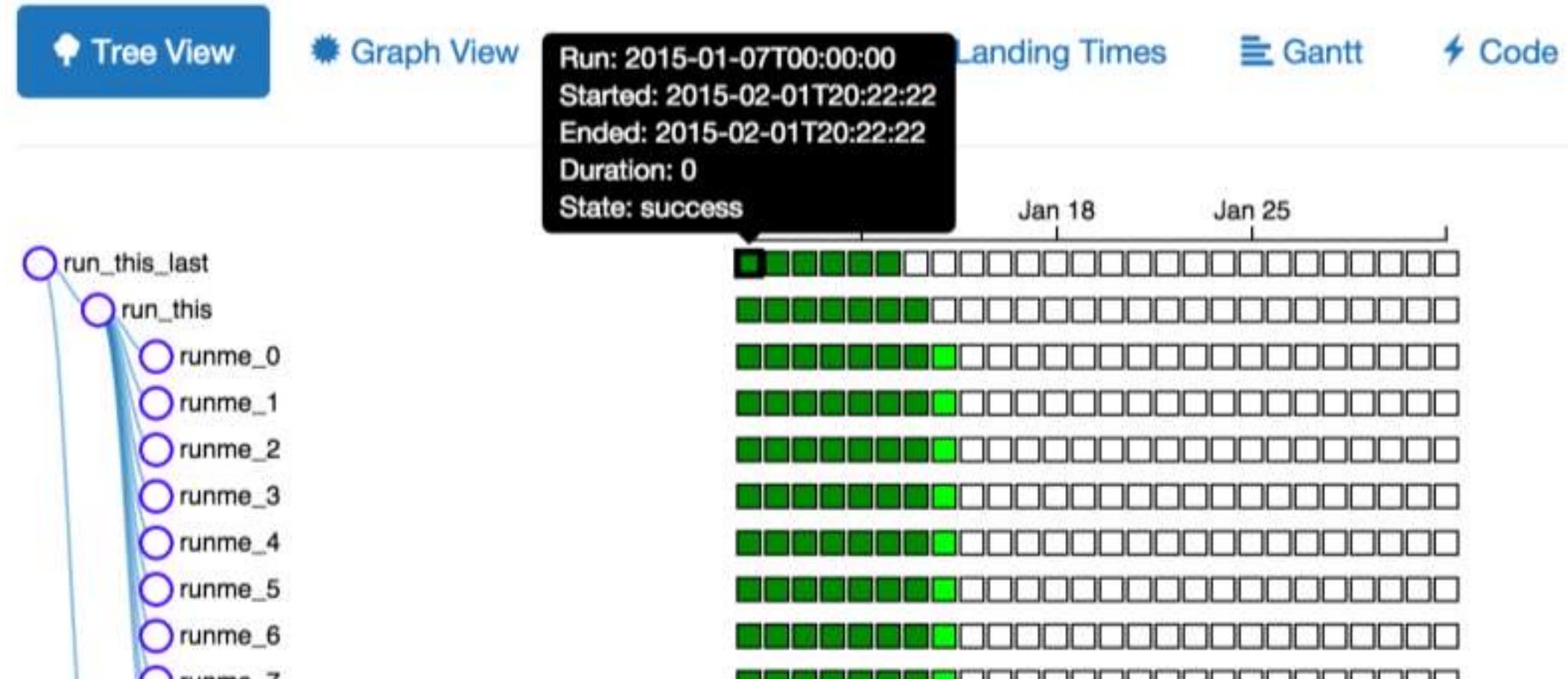
UI – DAG view



DAGs

DAG	Filepath	Owner	Task by State	Links
example1	example_dags/example1.py	airflow	80 1 0	      
example2	example_dags/example2.py	airflow	128 10 0	      
example3	example_dags/example3.py	airflow	138 5 0	      

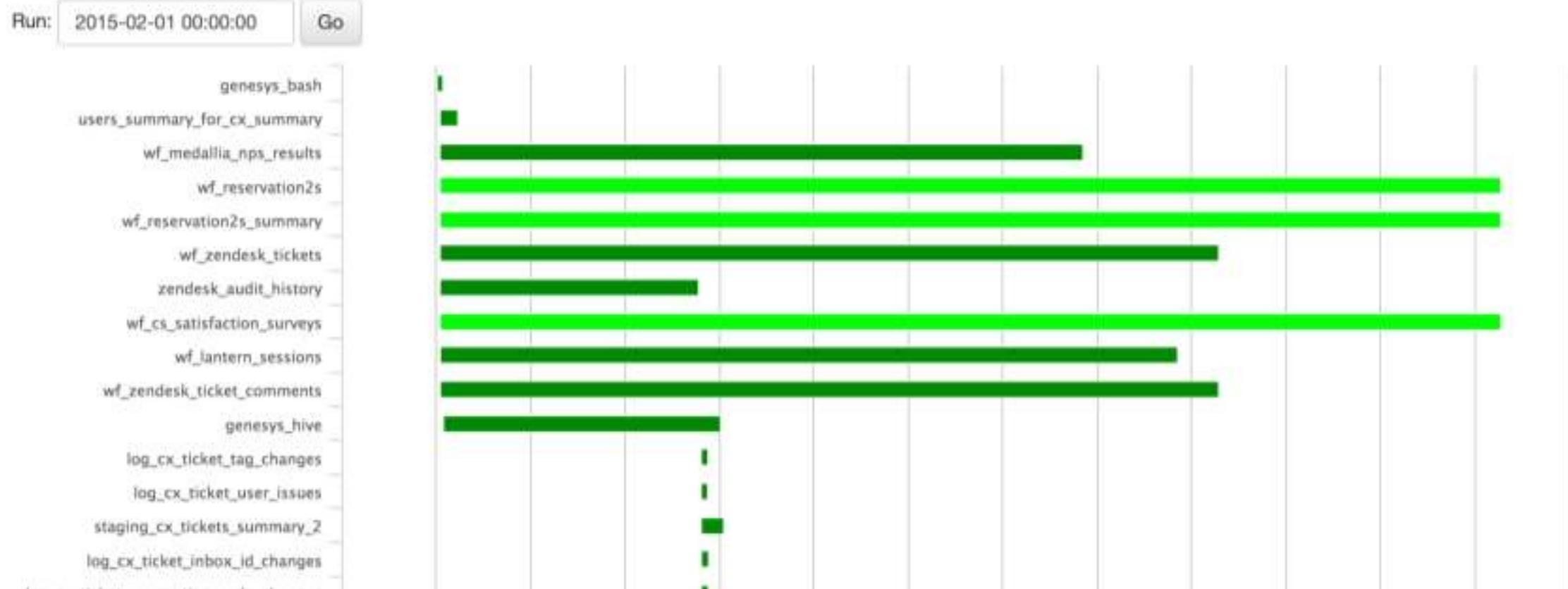
UI – Tree View



UI -Graph View

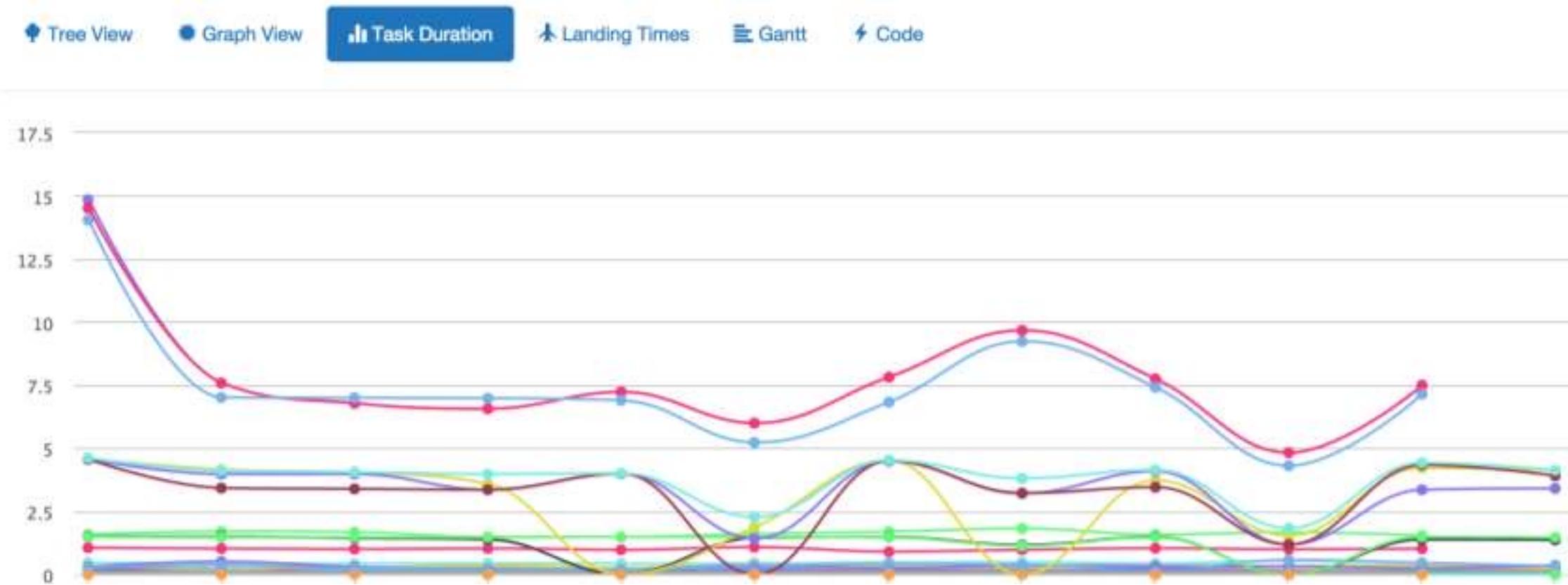


UI -Gantt

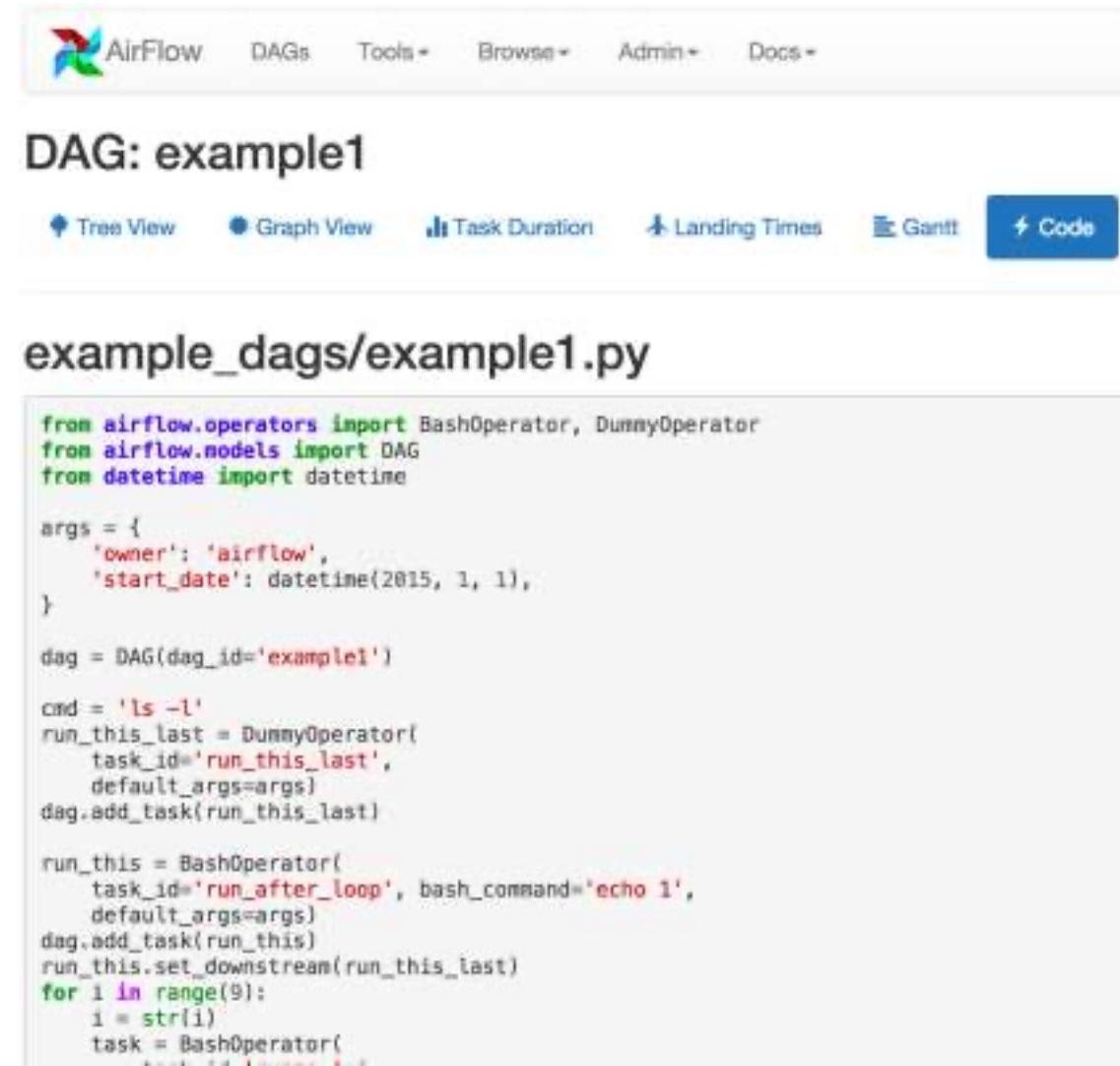


UI – Task duration

DAG: core_cx



UI –Code



The screenshot shows the Airflow web interface for a DAG named 'example1'. The top navigation bar includes links for AirFlow, DAGs, Tools, Browse, Admin, and Docs. Below the navigation, there are tabs for Tree View, Graph View, Task Duration, Landing Times, Gantt, and Code, with 'Code' being the active tab. The main content area displays the Python code for 'example1.py'.

```
from airflow.operators import BashOperator, DummyOperator
from airflow.models import DAG
from datetime import datetime

args = {
    'owner': 'airflow',
    'start_date': datetime(2015, 1, 1),
}

dag = DAG(dag_id='example1')

cmd = 'ls -l'
run_this_last = DummyOperator(
    task_id='run_this_last',
    default_args=args)
dag.add_task(run_this_last)

run_this = BashOperator(
    task_id='run_after_loop', bash_command='echo 1',
    default_args=args)
dag.add_task(run_this)
run_this.set_downstream(run_this_last)
for i in range(9):
    i = str(i)
    task = BashOperator(
        task_id='run-%s' % i,
        bash_command='ls > /tmp/%s.txt',
        default_args=args)
    dag.add_task(task)
    if i < 8:
        task.set_downstream(run_this)
```



Airflow -Pros

- Dynamic generating path
- Have both Time scheduler and Command line trigger
- Has Master/Worker model (automatically distribute tasks)
- Scale if you have many tasks in a chain.
 - ▣ But not be so useful to most of our tasks. Maybe useful for full dump.
- Fancy UI
 - ▣ Dependencies of tasks
 - ▣ Task success/failure
 - ▣ Scheduled tasks status
- Has utility lib to wait_datafor S3, Hadoop...

Airflow -Cons

- Additional DB/Redis or Rabbitmq for Celery
 - HA design: Use RDBMS/redis-cache in AWS
- Require python 2.7 and many other libraries.
- Not dependent on data. Just task dependency.
(Not big cons)
 - Write check data file code.