Amundsen: A Data Discovery Platform from Lyft

April 17th 2019

Jin Hyuk Chang | @jinhyukchang | Engineer, Lyft
Tao Feng | @feng-tao | Engineer, Lyft

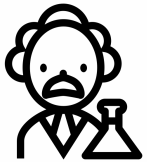
Agenda

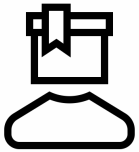
- Data at Lyft
- Challenges with Data Discovery
- Data Discovery at Lyft
- Demo
- Architecture
- Summary

Data platform users


Data Modelers


Analysts


Data Scientists


Product Managers

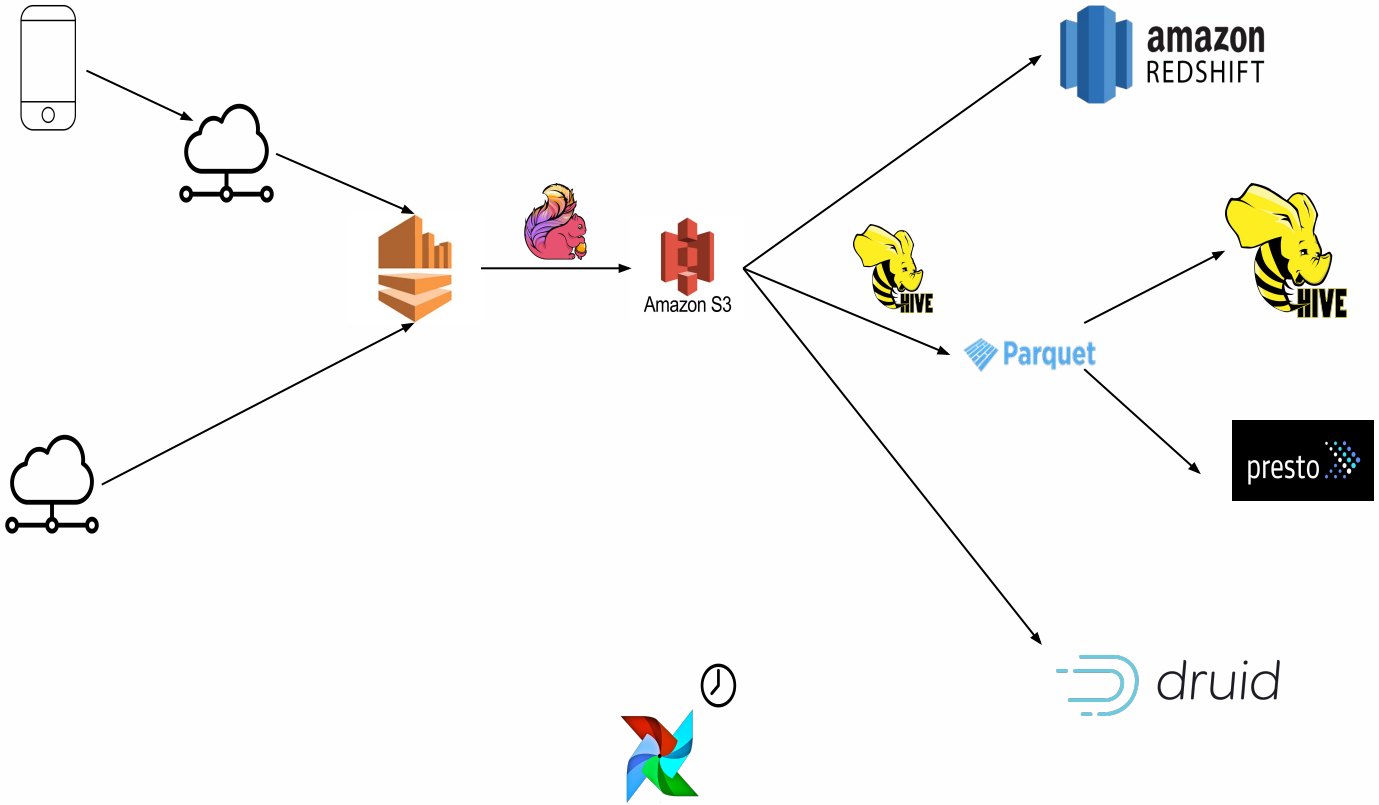

General Managers


Engineers


Experimenters

Data Platform

Core Infra high level architecture



Custom apps

Data Discovery

Hi! I am a n00b Data Scientist!

- My first project is to analyze and predict Data council Attendance
- Where is the data?
- What does it mean?

Status quo

- Option 1: Phone a friend!
- Option 2: Github search

Understand the context

- What does this field mean?
  - Does attendance data include employees?
  - Does it include revenue?
- Let me dig in and understand

Explore

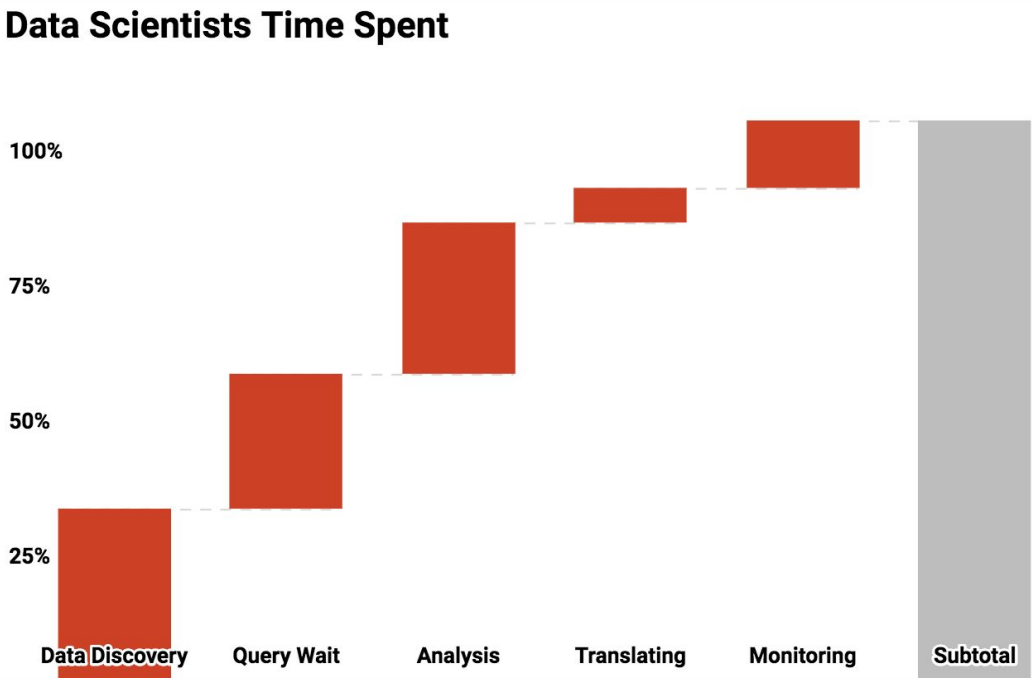
```
SELECT
*
FROM
default.my_table
WHERE ds='2018-01-01'
LIMIT 100;
```

Exploring with SELECT * is EVIL

1. Lack of productivity for data scientists
2. Increased load on the databases

Data Scientists spend upto 1/3rd time in Data Discovery…

- Data discovery
  - Lack of understanding of what data exists, where, who owns it, who uses it, and how to request access.

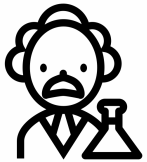


Audience for data discovery

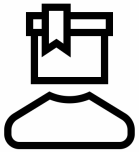
Data Discovery - User personas

Data Modelers

Analysts



Data Scientists



Product Managers

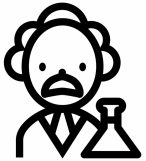
General Managers

Engineers

Experimenters

Data Platform

3 Data Scientist personas

 Power user	 Noob user	 Manager
● All info in their head ● Get interrupted a lot due to questions	● Lost ● Ask “power users” a lot of questions	● Dependencies landing on time ● Communicating with stakeholders

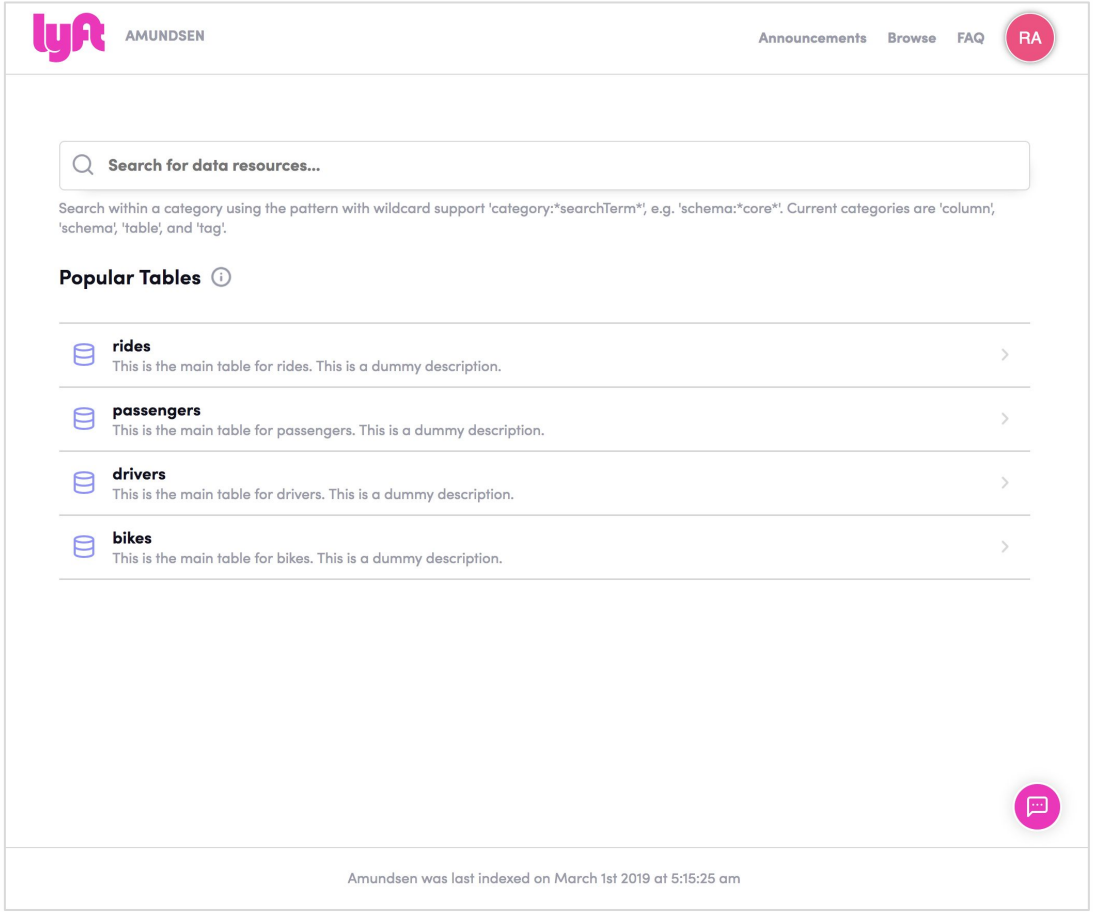
Data Discovery answers 3 kinds of questions

Search based	**Lineage based**	**Network based**
Where is the table/dashboard for X? What does it contain?	I am changing a data model, who are the owner and most common users?	I want to follow a power user in my team.
Does this analysis already exist?	This table's delivery was delayed today, I want to notify everyone downstream.	I want to bookmark tables of interest and get a feed of data delay, schema change, incidents.

Meet Amundsen

First person to discover the South Pole - Norwegian explorer, Roald Amundsen

Landing page optimized for search



Search results ranked on relevance and query activity

passenger

Search within a category using the pattern with wildcard support 'category:*searchTerm*', e.g. 'schema:*core*'. Current categories are 'column', 'schema', 'table', and 'tag'.

1-2 of 2 results

passenger
This is the main table for passenger . This is a dummy description.

passenger_ride_cancellations
Passenger ride cancels. This is a dummy description.

How does search work?

Relevance - search for “apple” on Google

Low relevance

High relevance

Popularity - search for “apple” on Google

Low popularity

High popularity

Striking the balance

Relevance	Popularity
● Names, Descriptions, Tags, [owners, frequent users]	● Querying activity ● Dashboarding ● Different weights for automated vs adhoc querying

Back to mocks...

Search results ranked on relevance and query activity

passenger

Search within a category using the pattern with wildcard support 'category:*searchTerm*', e.g. 'schema:*core*'. Current categories are 'column', 'schema', 'table', and 'tag'.

1-2 of 2 results

passenger
This is the main table for passenger . This is a dummy description.

passenger_ride_cancellations
Passenger ride cancels. This is a dummy description.

Detailed description and metadata about data resources

AMUNDSEN

Announcements Browse FAQ RA

Rides

May 25, 2012 – Mar 03, 2019

The source for all ride related data.

Columns

users `string`
Dummy description. You can click here to edit.

desk_count `int`
Dummy description. You can click here to edit.

passenger `string`
Add Description

ride_id `string`
Add Description

driver_os `string`
Add Description

driver_os_version `string`
Dummy description. You can click here to edit.

driver_app_version `string`
Add Description

OWNED BY
test@lyft.com
default-user@lyft.com
Add

FREQUENT USERS

GENERATED BY
rides/rides

SOURCE CODE
rides.rides

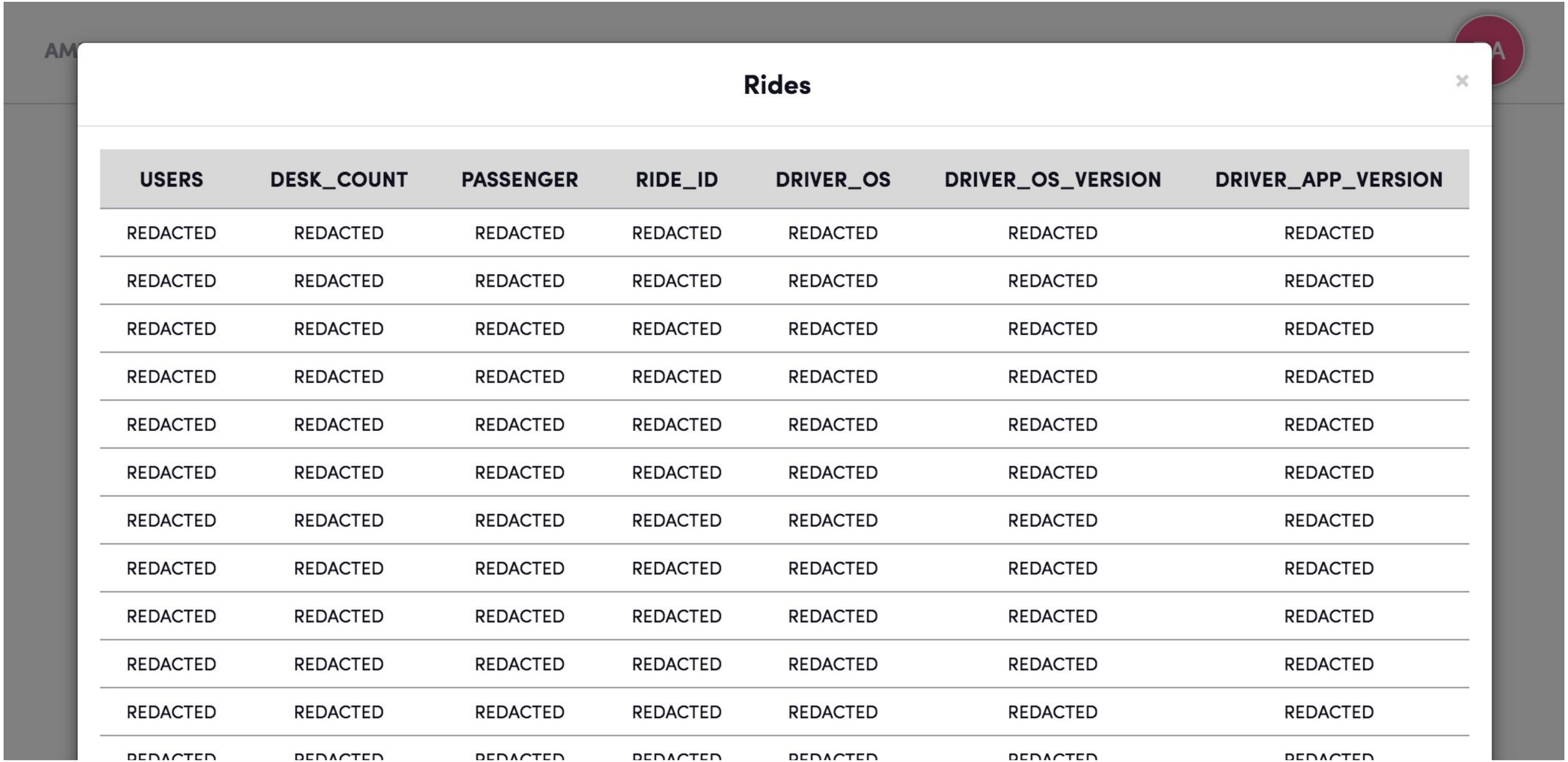
TABLE LINEAGE (BETA)
rides.rides

TABLE PROFILE (BETA)
Preview Data
Explore with SQL

TAGS
driver passenger events

Amundsen was last indexed on March 1st 2019 at 5:15:25 am

Data Preview within the tool



Rides

USERS	DESK_COUNT	PASSENGER	RIDE_ID	DRIVER_OS	DRIVER_OS_VERSION	DRIVER_APP_VERSION
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED
REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED	REDACTED

Computed stats about column metadata

desk_count `int`

Dummy description. You can click here to edit.

How is this data generated?

These stats are based on data collected for this column on 3/1/2019

passenger `string`

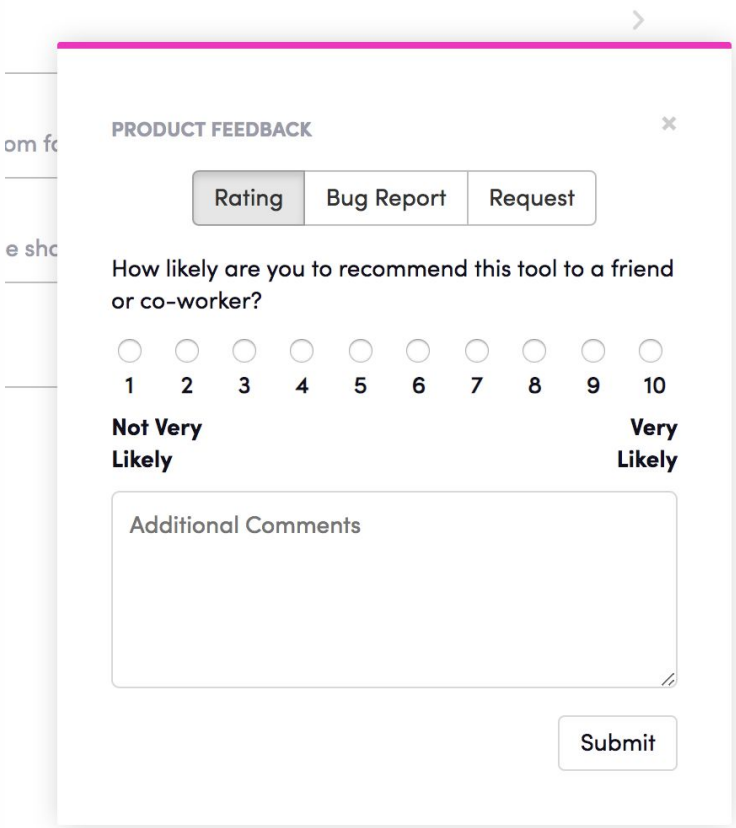
Add Description

COUNT	COUNT_NULL	COUNT_DISTINCT	LEN_MAX
123456	321	123456	24

LEN_MIN	LEN_AVG	LEN_SUM
24	24	1234567

Disclaimer: these stats are arbitrary.

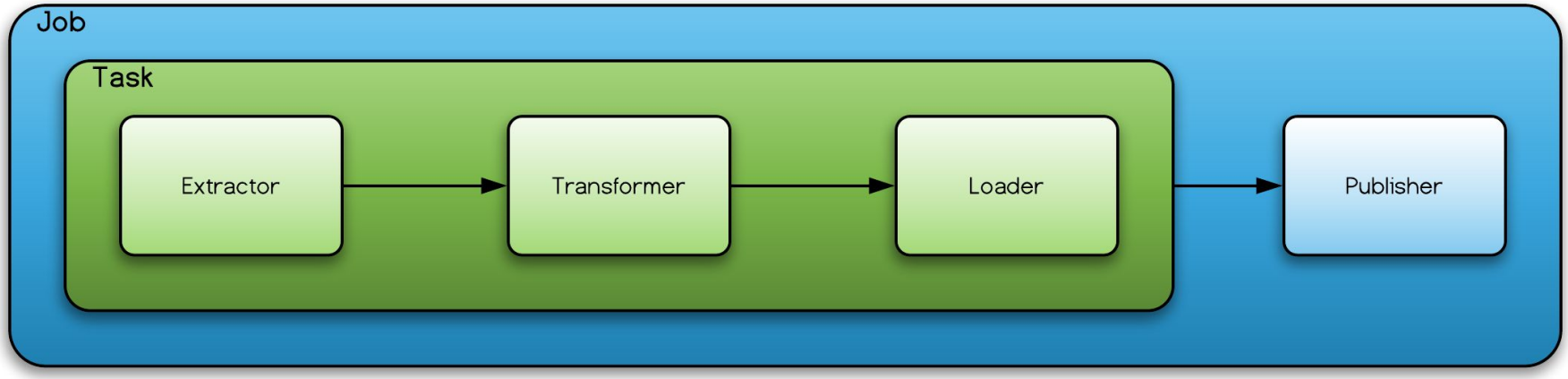
Built-in user feedback



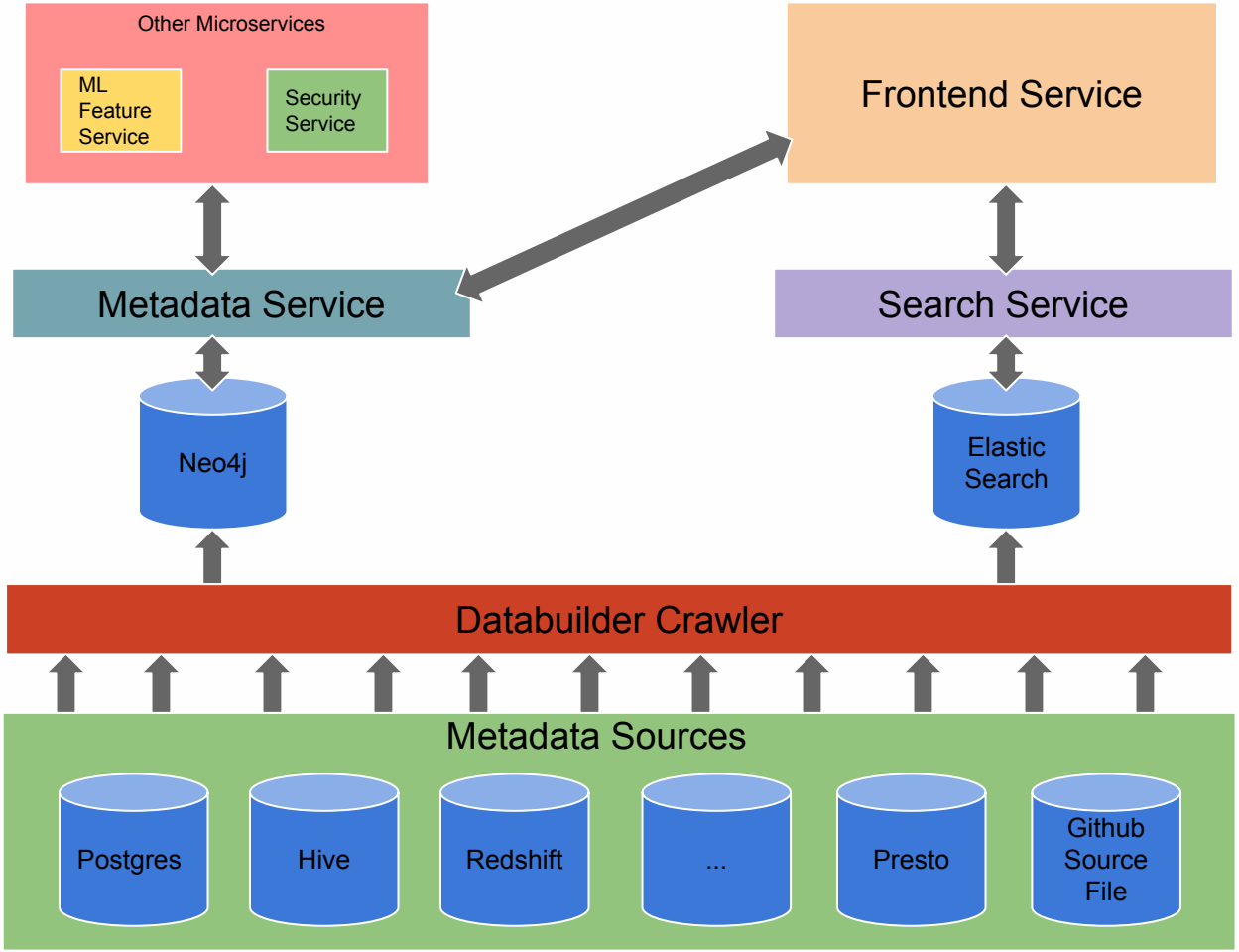
Demo

Open source in mind

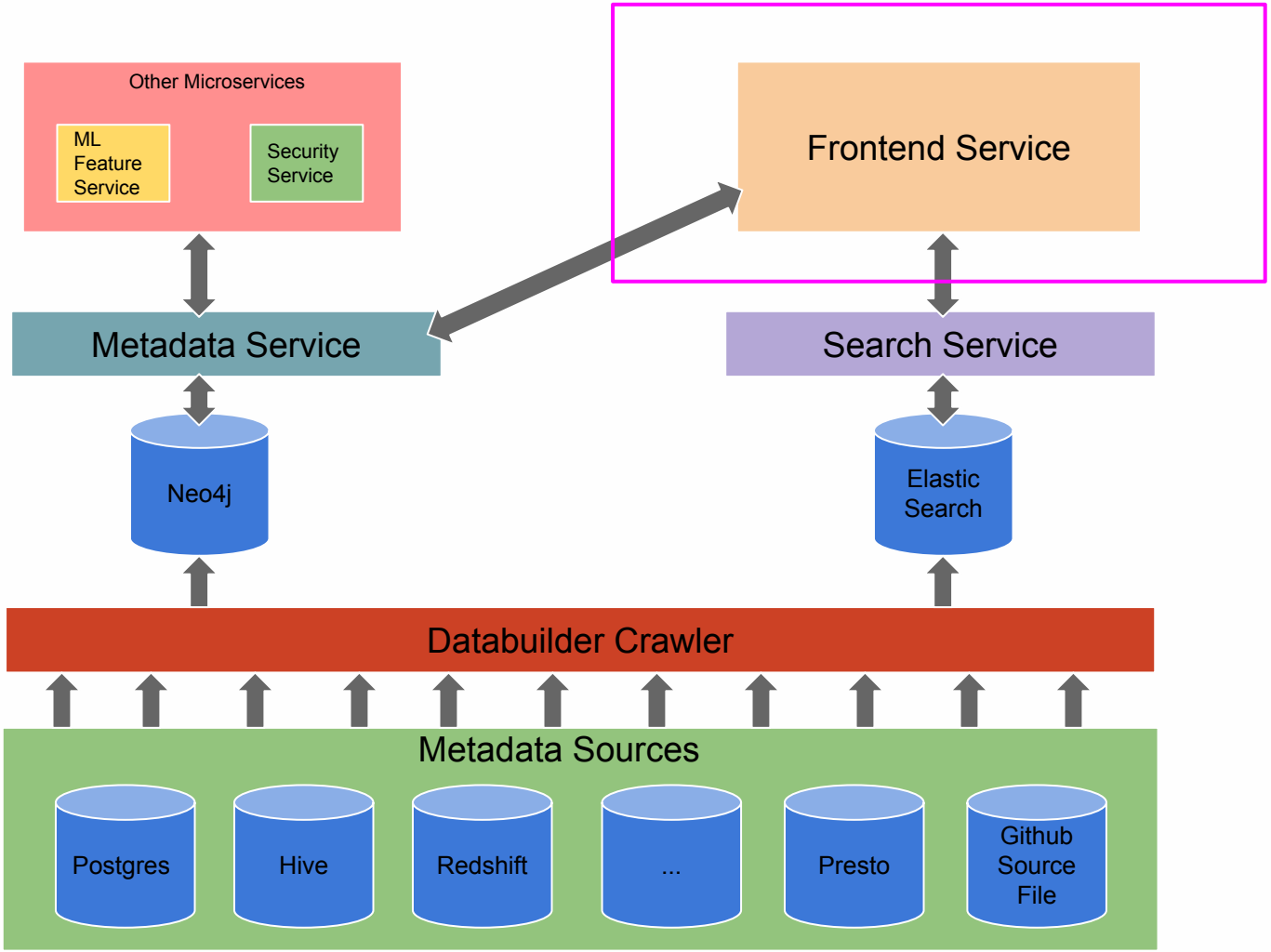
- Pluggable code to each micro-services via Python entry point, etc
- Pluggable API endpoint via Blueprint
- Build your ingestion pipeline like a Lego brick



Amundsen's architecture


Other Microservices
ML Feature Service
Security Service
Frontend Service
Metadata Service
Search Service
Neo4j
Elastic Search
Databuilder Crawler
Metadata Sources
Postgres
Hive
Redshift
...
Presto
Github Source File

1. Frontend Service


Other Microservices
ML Feature Service
Security Service
Frontend Service
Metadata Service
Search Service
Neo4j
Elastic Search
Databuilder Crawler
Metadata Sources
Postgres
Hive
Redshift
...
Presto
Github Source File

Amundsen table detail page

AMUNDSEN

Announcements Browse FAQ RA

Rides

May 25, 2012 – Mar 03, 2019

The source for all ride related data.

Columns

users `string`
Dummy description. You can click here to edit.

desk_count `int`
Dummy description. You can click here to edit.

passenger `string`
Add Description

ride_id `string`
Add Description

driver_os `string`
Add Description

driver_os_version `string`
Dummy description. You can click here to edit.

driver_app_version `string`
Add Description

OWNED BY
- test@lyft.com
- default-user@lyft.com
- Add

FREQUENT USERS

GENERATED BY
rides/rides

SOURCE CODE
rides.rides

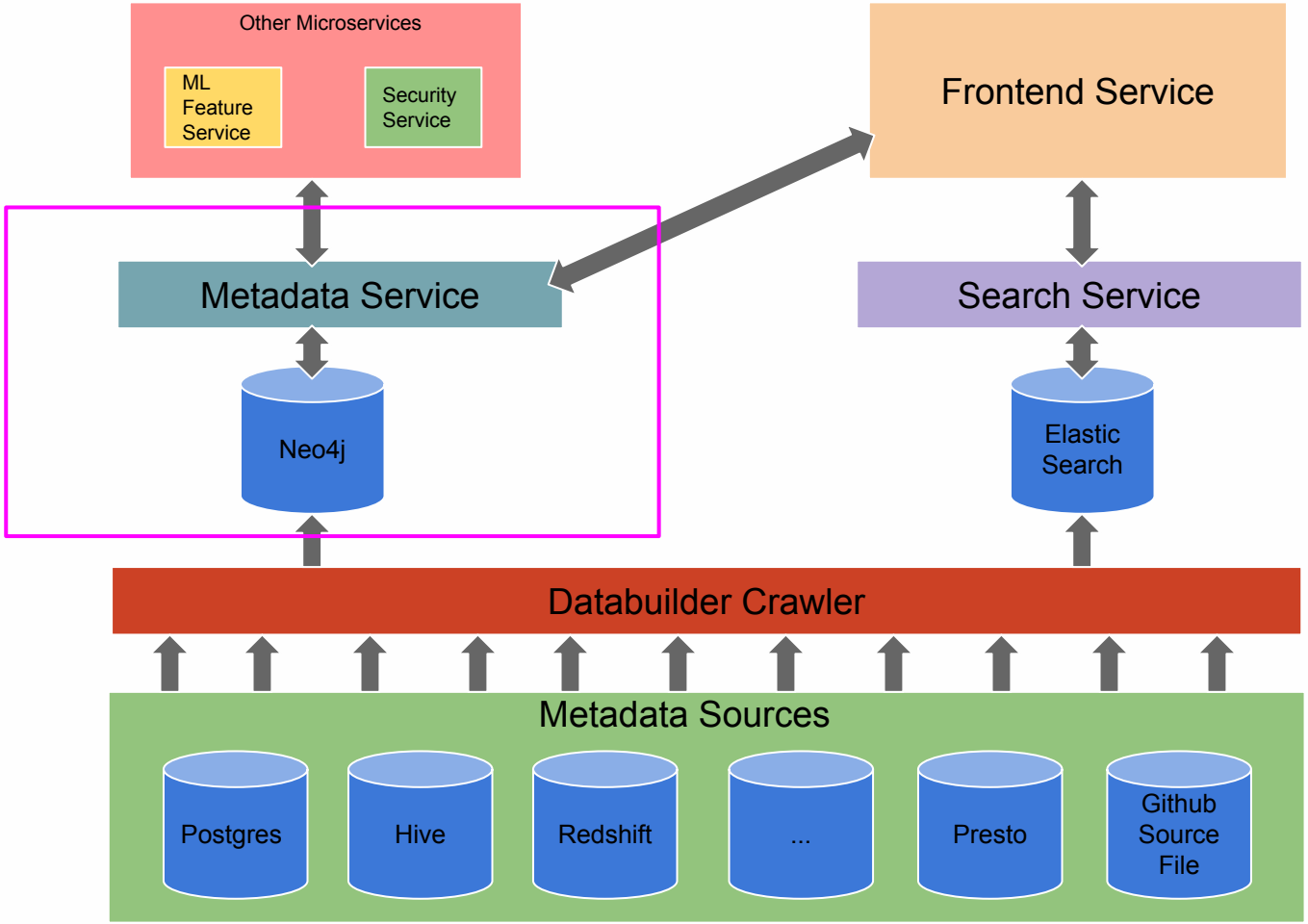
TABLE LINEAGE (BETA)
rides.rides

TABLE PROFILE (BETA)
Preview Data
Explore with SQL

TAGS
driver passenger events

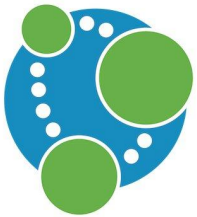
Amundsen was last indexed on March 1st 2019 at 5:15:25 am

2. Metadata Service


Other Microservices
ML Feature Service
Security Service
Frontend Service
Metadata Service
Search Service
Neo4j
Elastic Search
Databuilder Crawler
Metadata Sources
Postgres
Hive
Redshift
...
Presto
Github Source File

2. Metadata Service

- A thin proxy layer to interact with graph database
  - Currently Neo4j is the default option for graph backend engine
  - Work with the community to support Apache Atlas



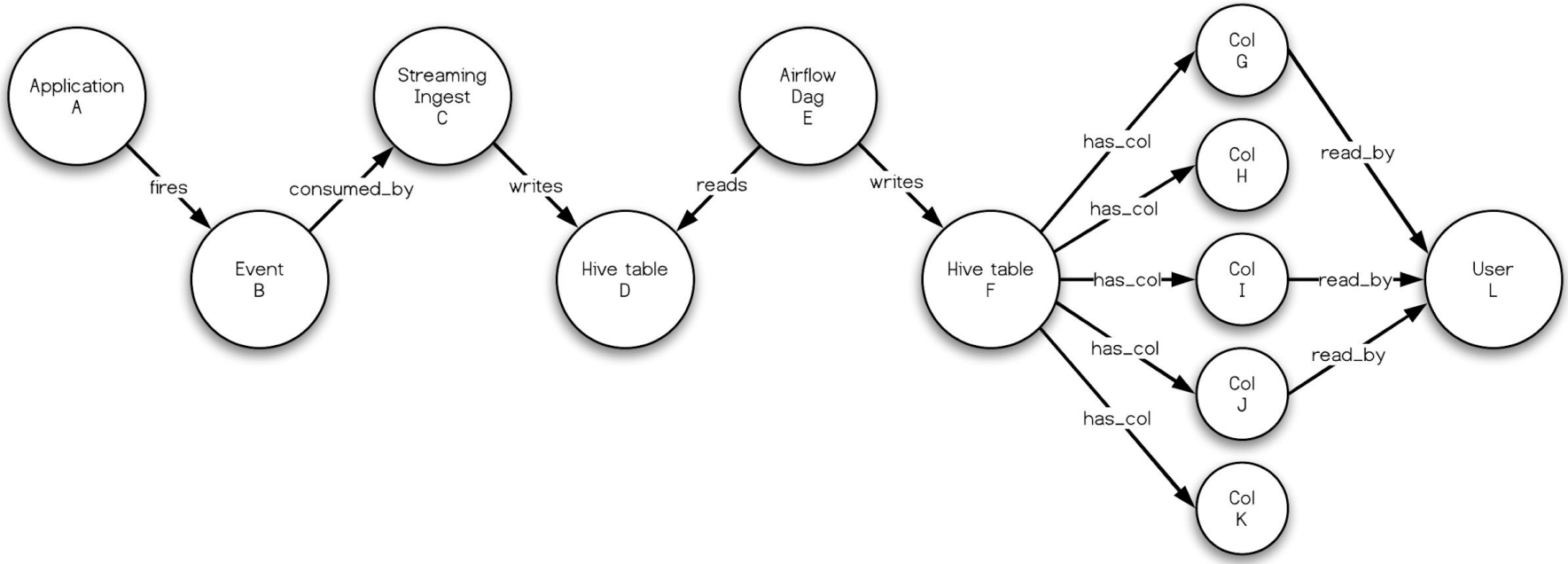
Apache Atlas

- Support Rest API for other services pushing / pulling metadata directly

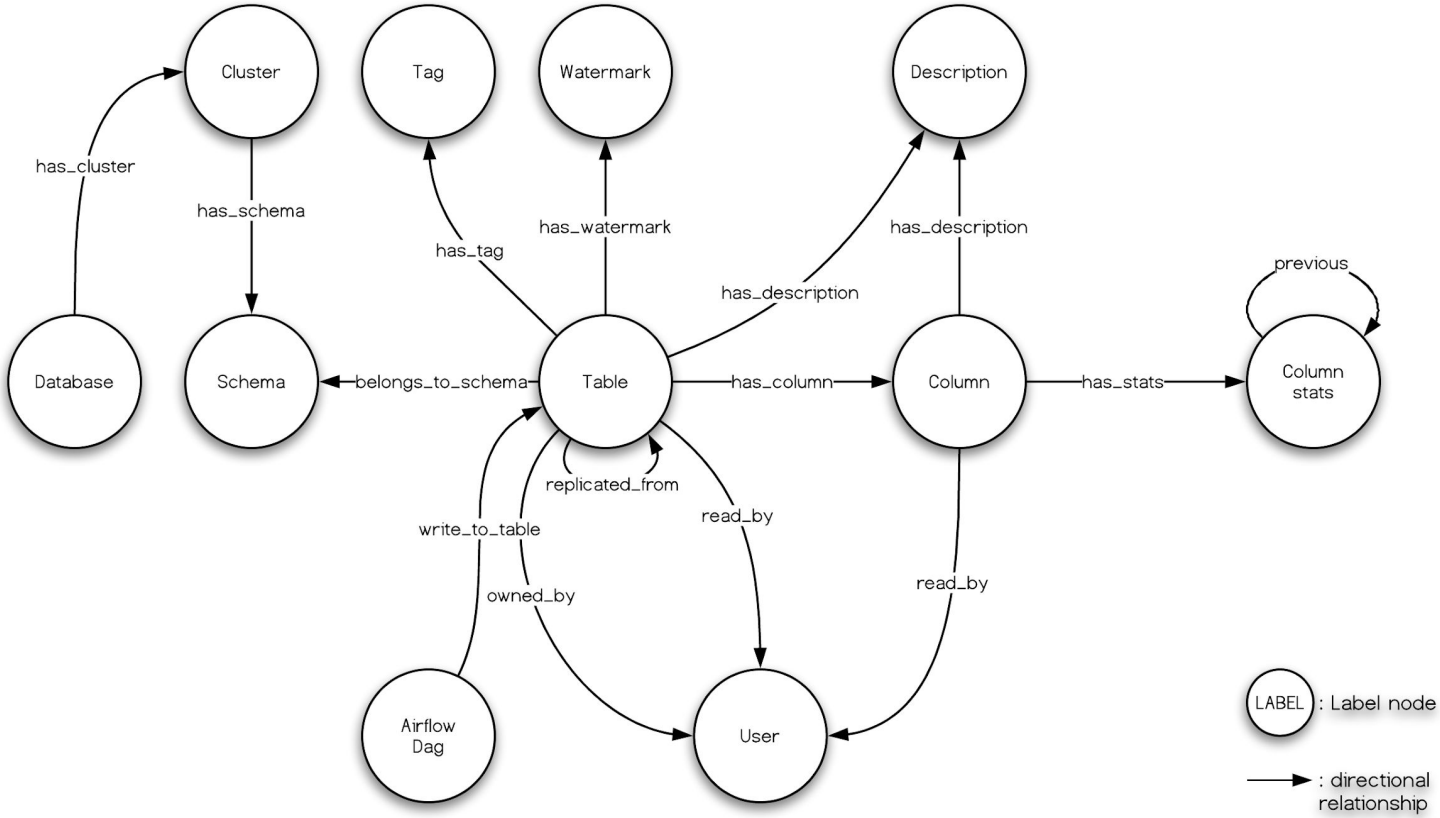
Trade Off #1

Why choose Graph database

Why Graph database?



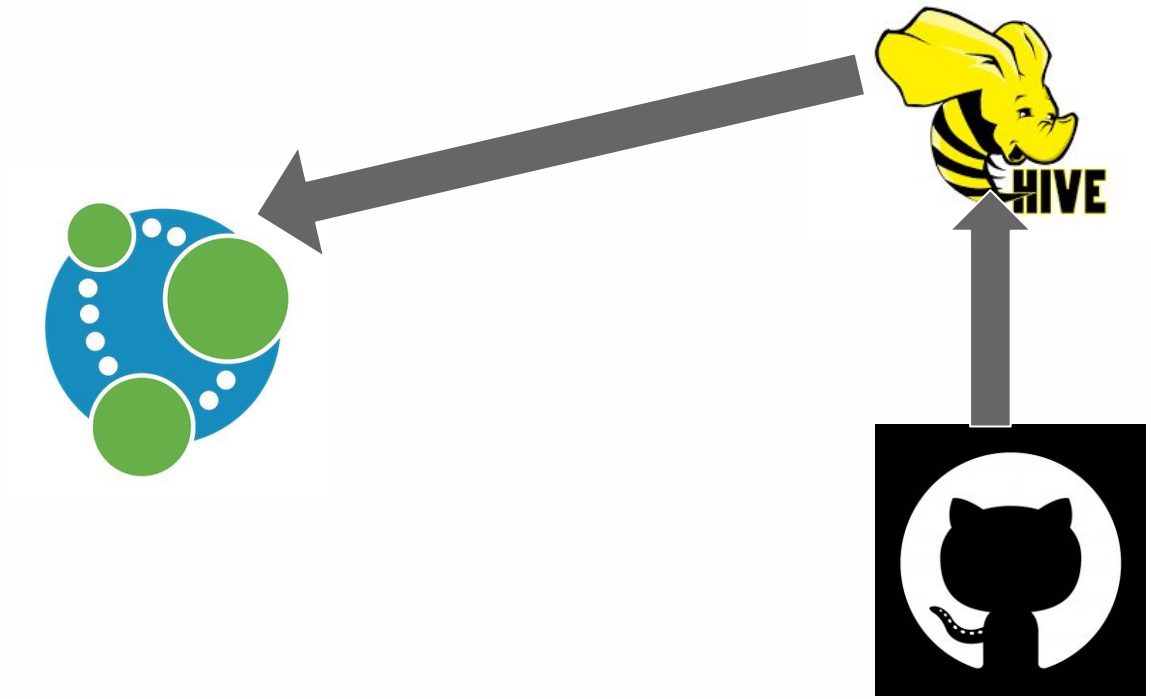
Why Graph database?



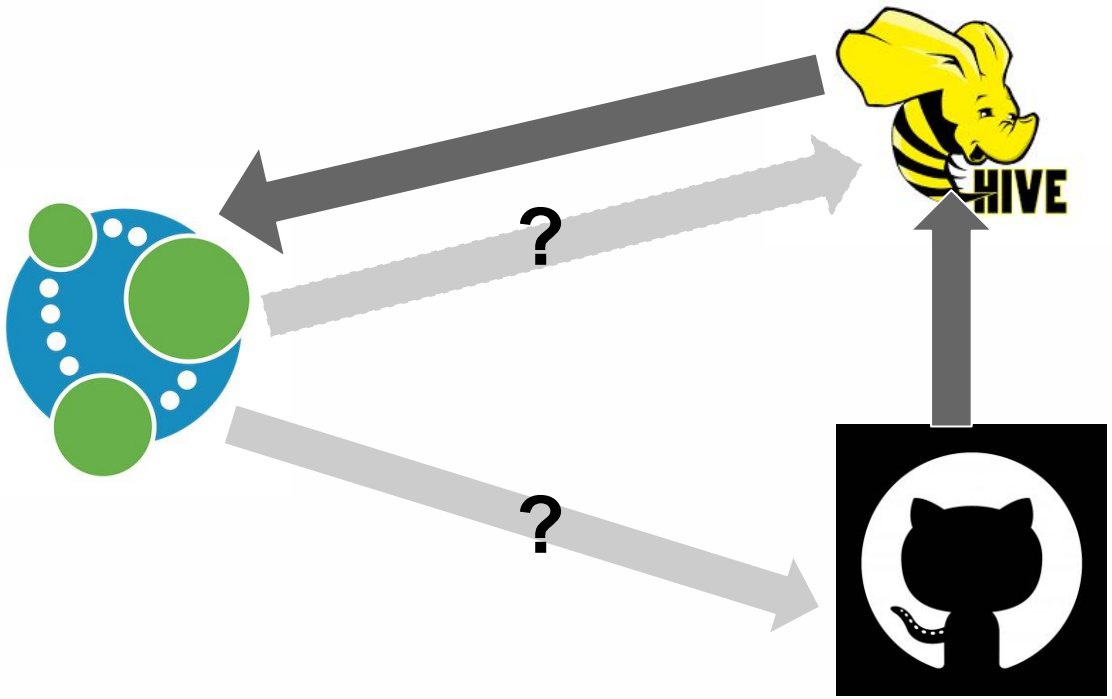
Trade Off #2

Why not propagate the metadata back to source

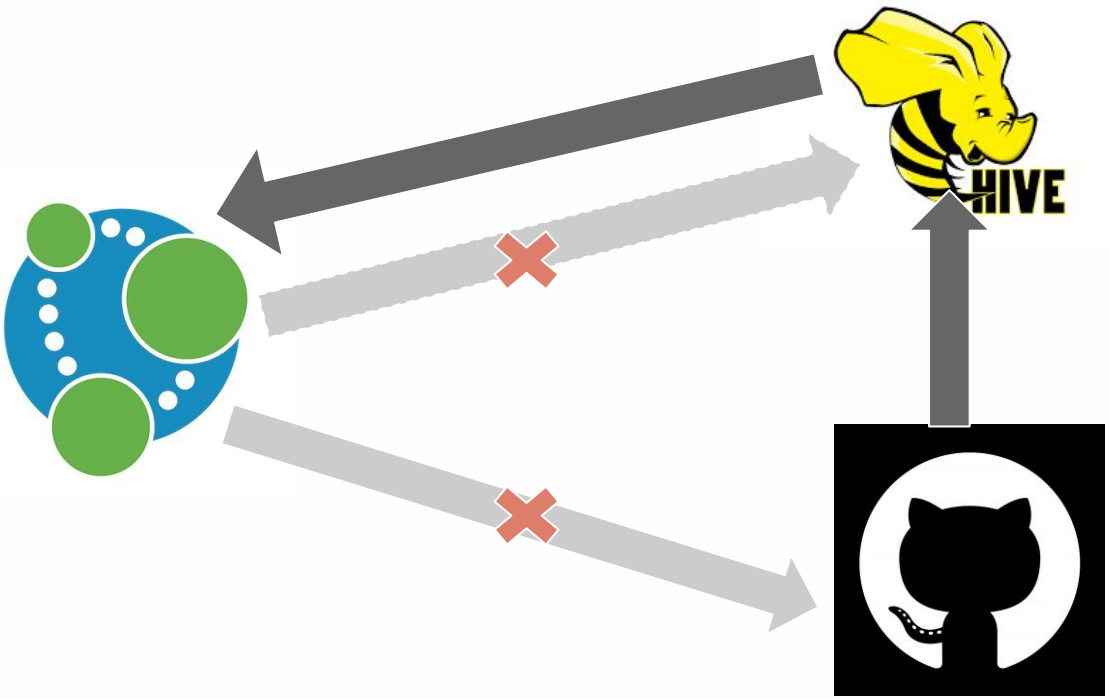
Why not propagate the metadata back to source



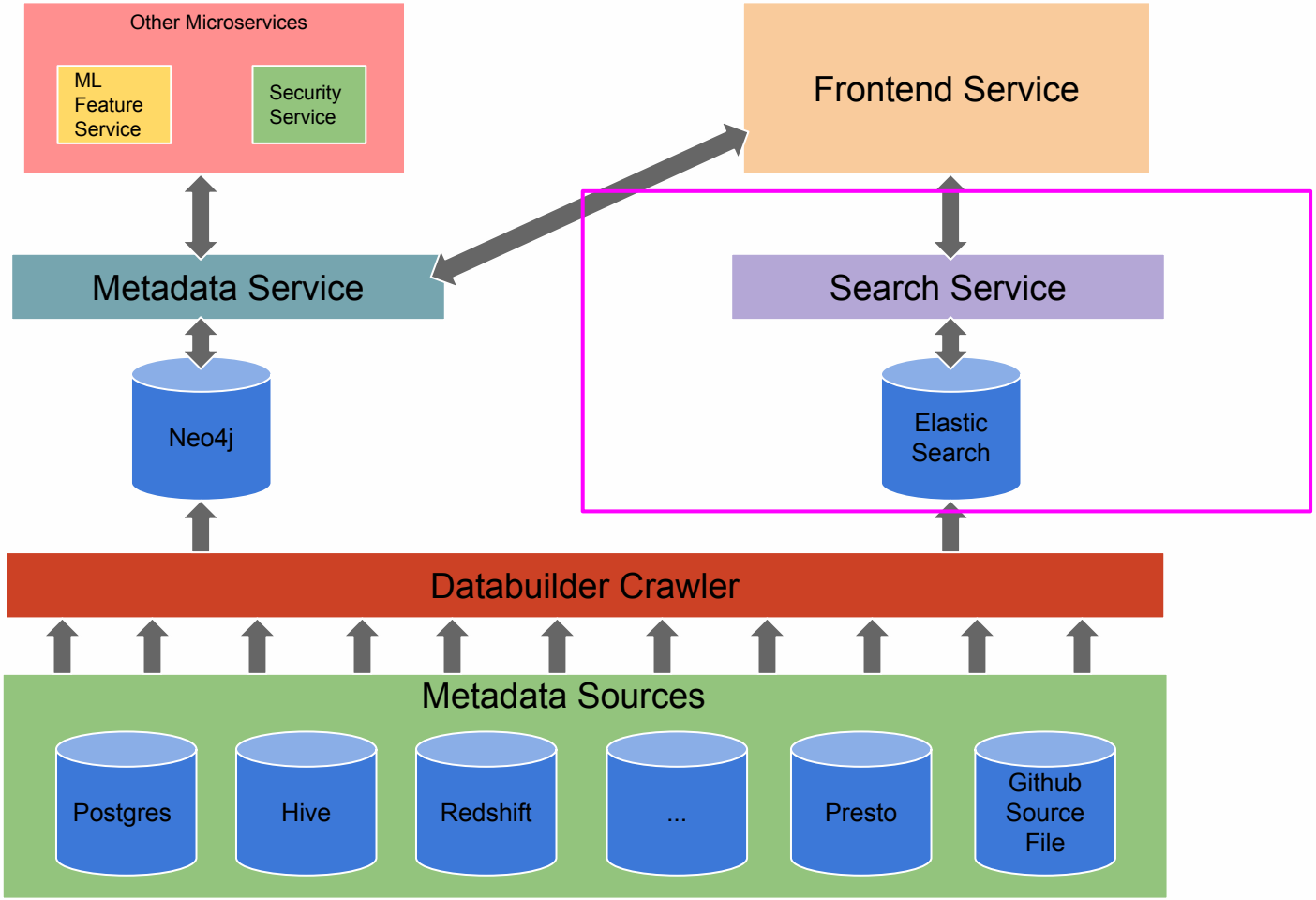
Why not propagate the metadata back to source



Why not propagate the metadata back to source



3. Search Service


Other Microservices
ML Feature Service
Security Service
Frontend Service
Metadata Service
Search Service
Neo4j
Elastic Search
Databuilder Crawler
Metadata Sources
Postgres
Hive
Redshift
...
Presto
Github Source File

3. Search Service

- A thin proxy layer to interact with the search backend
  - Currently it supports Elasticsearch as the search backend.

- Support different search patterns
  - **Normal** Search: match records based on relevancy
  - **Category** Search: match records first based on data type, then relevancy
  - **Wildcard** Search

Challenge #1

How to make the search result more relevant?

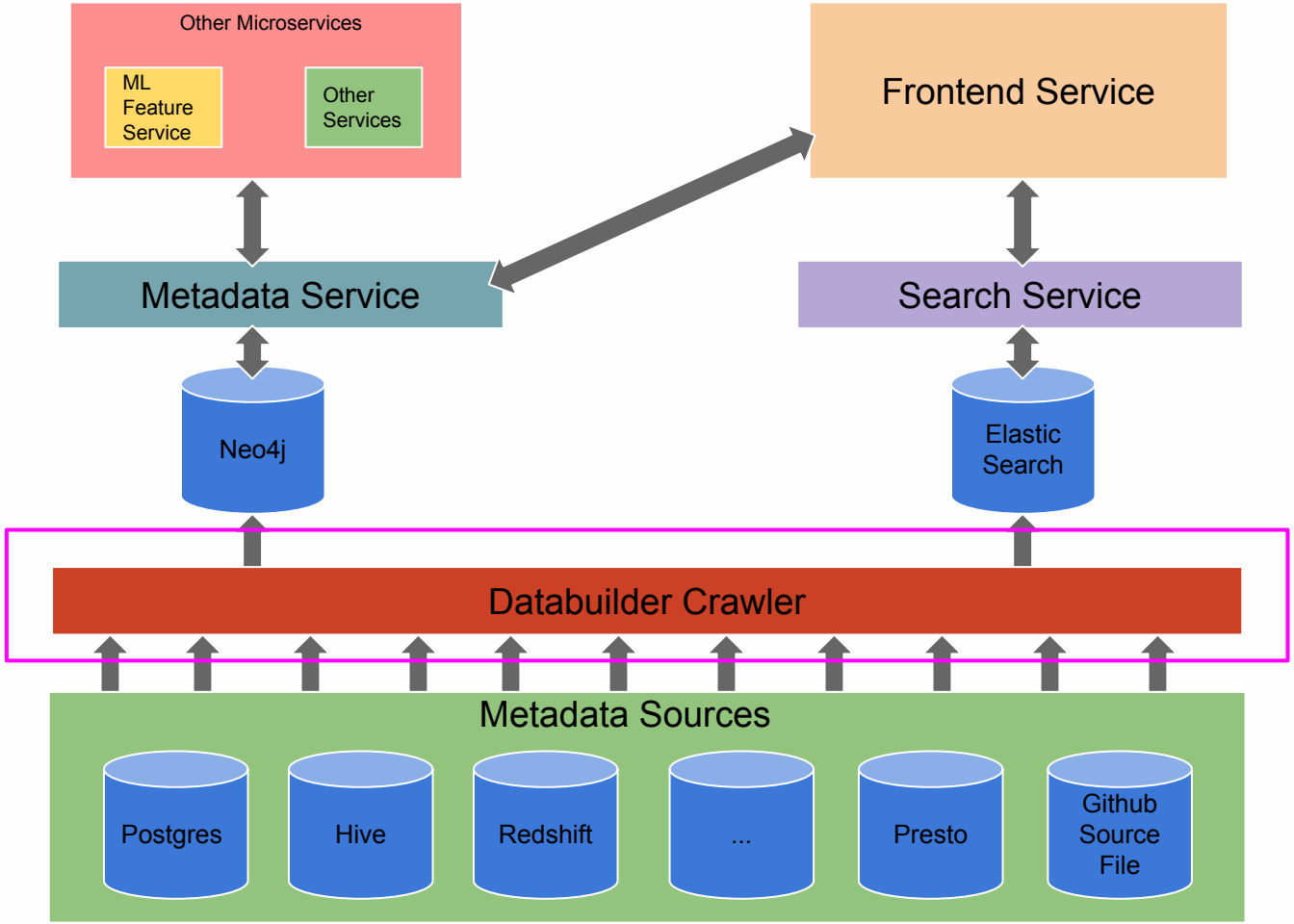
How to make the search result more relevant?

- Define a search quality metric
  - Click-Through-Rate (CTR) over top 5 results

- Search behaviour instrumentation is key

- Couple of improvements:
  - Boost the **exact table** ranking
  - Support **wildcard** search (e.g. `event_*`)
  - Support **category** search (e.g. `column: is_line_ride`)

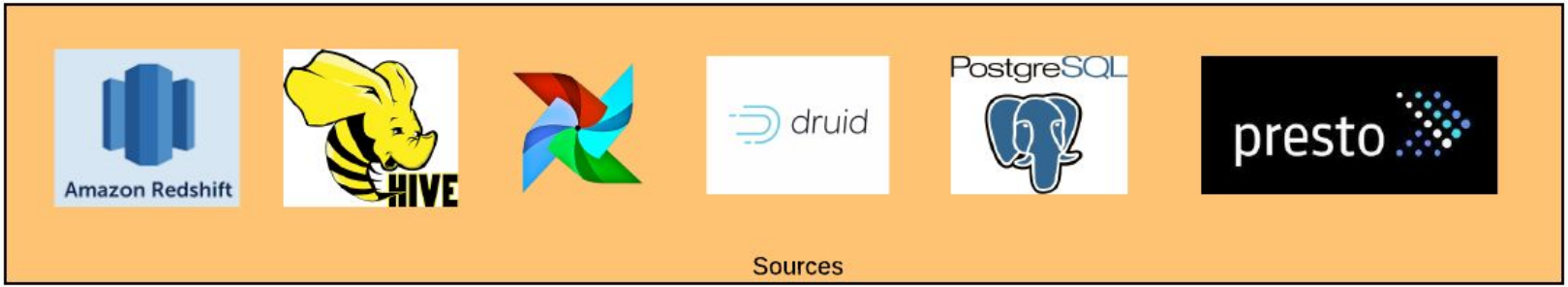
4. Data Builder


Other Microservices
ML Feature Service
Other Services
Frontend Service
Metadata Service
Search Service
Neo4j
Elastic Search
Databuilder Crawler
Metadata Sources
Postgres
Hive
Redshift
...
Presto
Github Source File

Challenge #1

Various forms of metadata

Metadata Sources @ Lyft



Metadata - Challenges

- **No Standardization**: No single data model that fits for all data resources
  - A data resource could be a table, an Airflow DAG or a dashboard

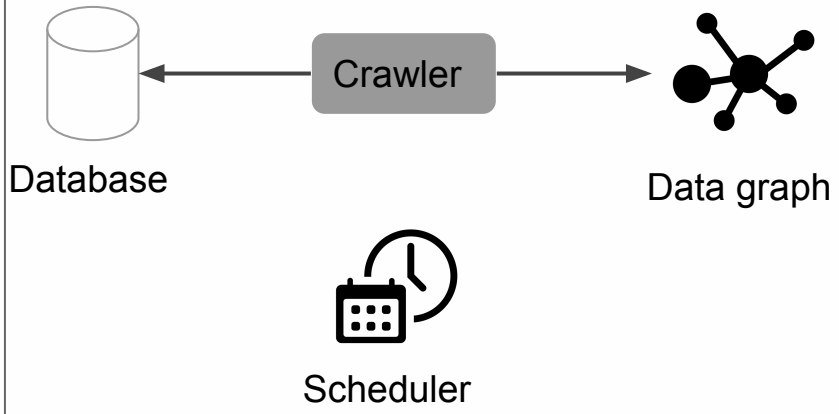
- **Different Extraction**: Each data set metadata is stored and fetched differently
  - Hive Table: Stored in Hive metastore
  - RDBMS(postgres etc): Fetched through DBAPI interface
  - Github source code: Fetched through git hook
  - Mode dashboard: Fetched through Mode API
  - ...

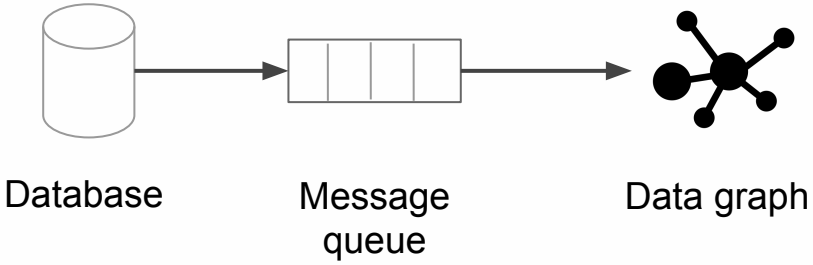
Challenge #2

Pull model vs Push model

Pull model vs. Push model

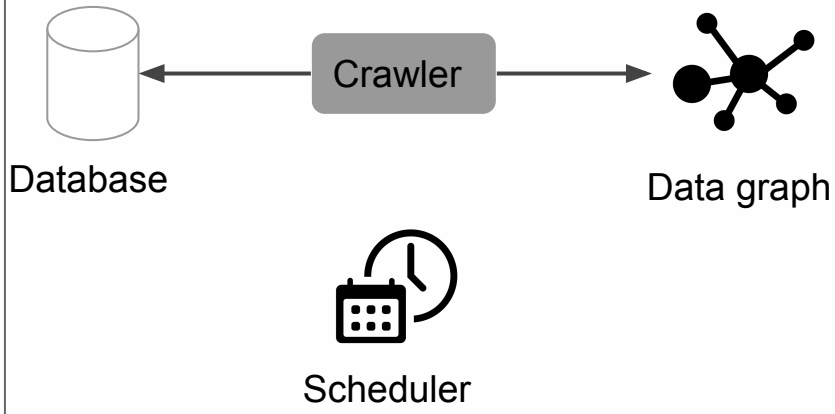
Pull Model	Push Model
• Periodically update the index by pulling from the system (e.g. database) via crawlers.	• The system (e.g. database) pushes metadata to a message bus which downstream subscribes to.

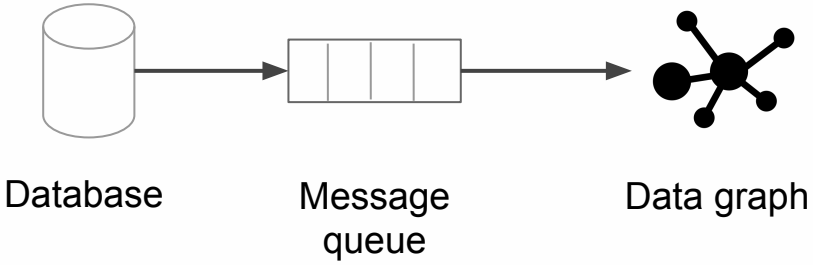




Pull model vs. push model

Pull Model	Push Model
• Onus of integration lays on data graph • No interface to prescribe, hard to maintain crawlers	• Onus of integration lies on database • Message format serves as the interface • Allows for near-real time indexing

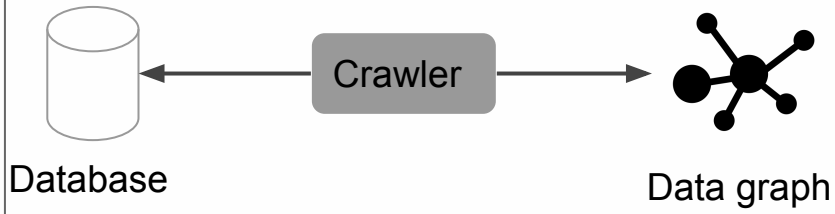




Pull model vs. push model

Pull Model

- Onus of integration lays on data graph
- No interface to prescribe, hard to maintain crawlers

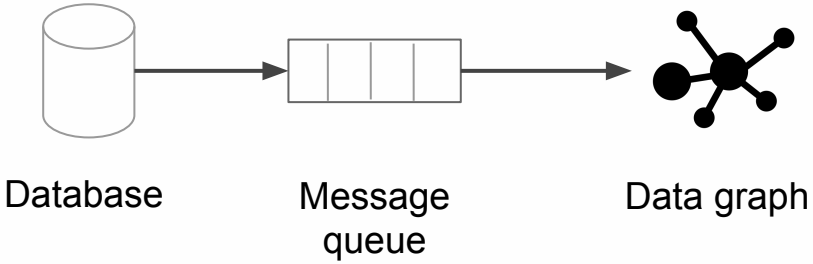


Preferred if

- Waiting for indexing is ok
- Working with “strapped” teams
- There’s already an interface

Push Model

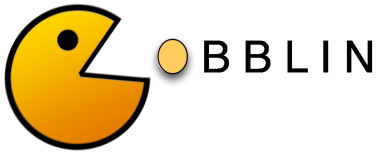
- Onus of integration lies on database
- Message format serves as the interface
- Allows for near-real time indexing

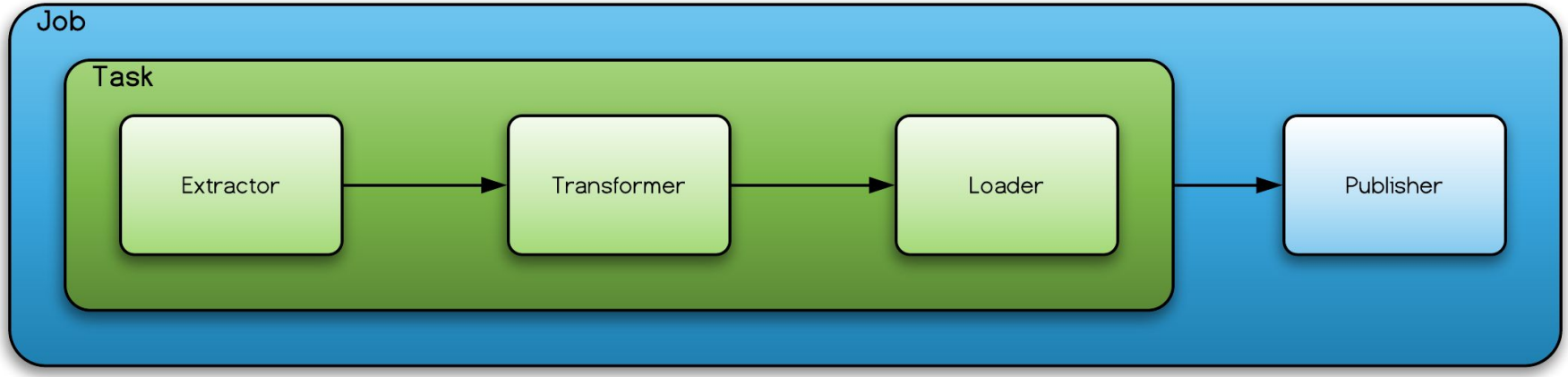


Preferred if

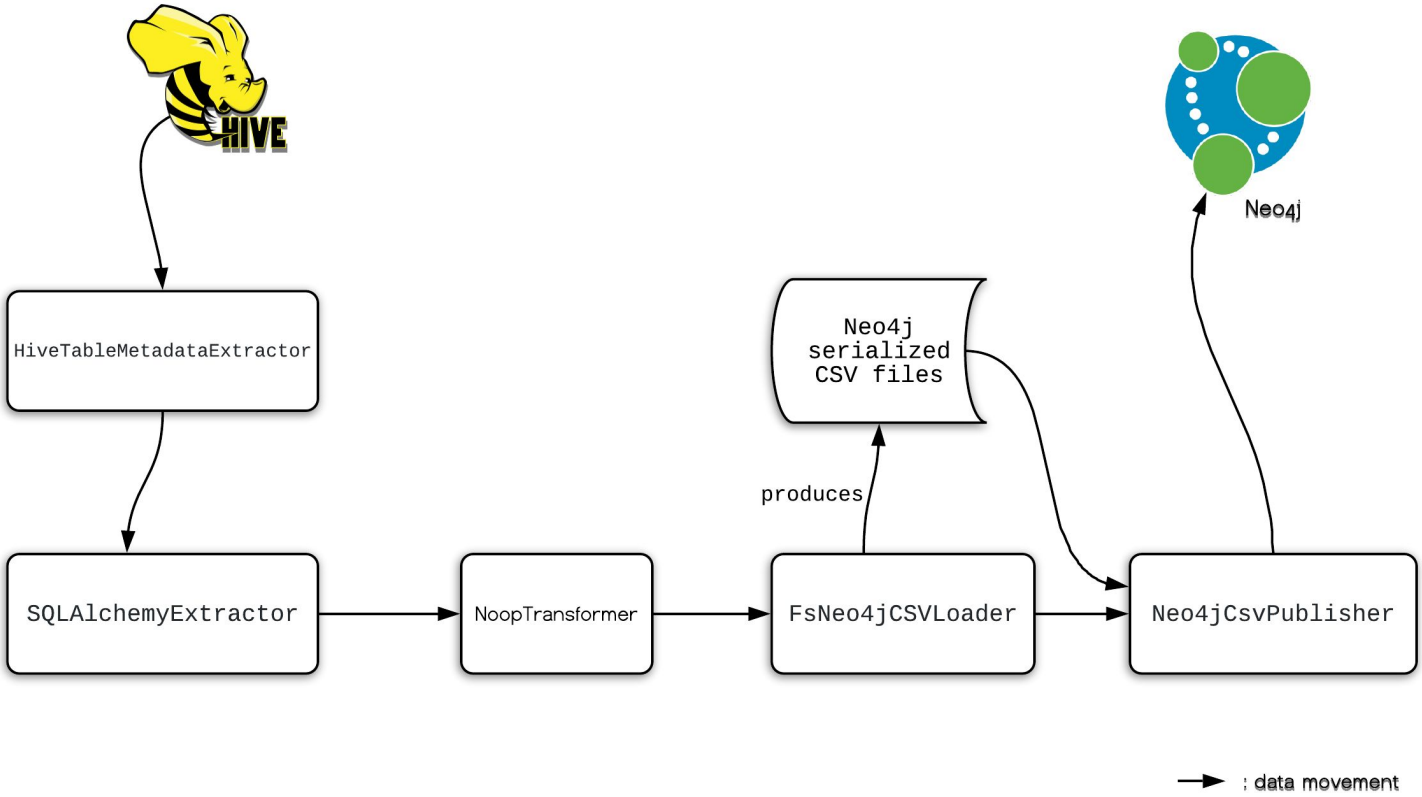
- Near-real time indexing is important
- Clean interface doesn’t exist
- Other tools like Wherehows are moving towards Push Model

4. Databuilder

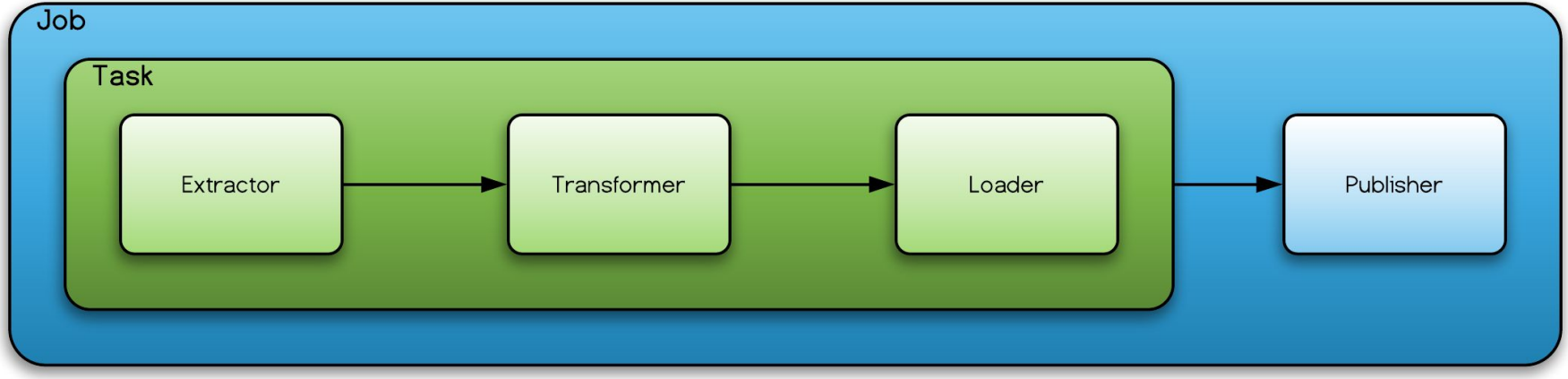




Databuilder in action



How are we building data? Databuilder



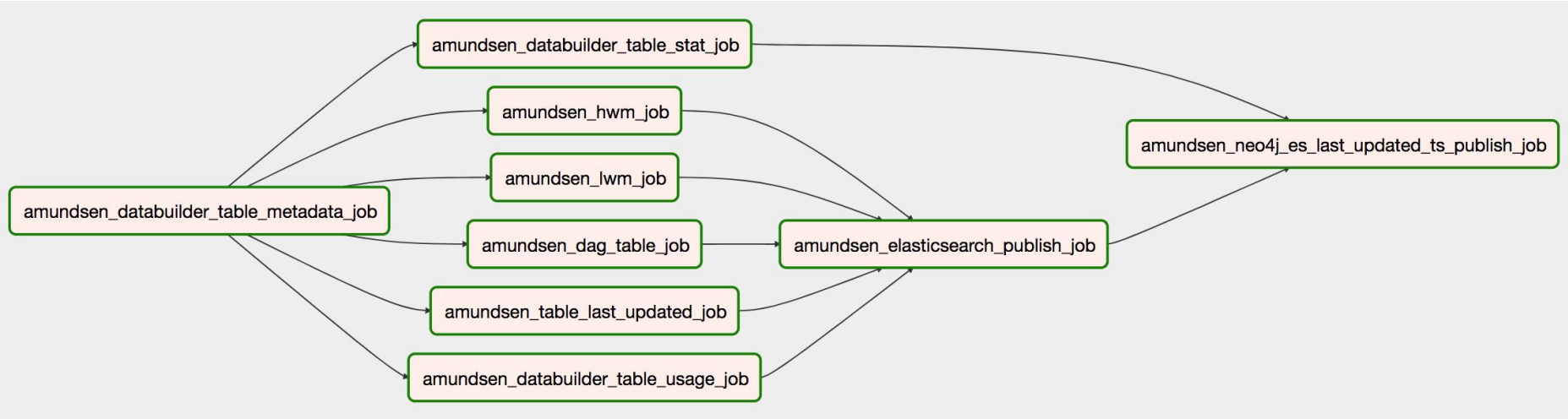

```
task = DefaultTask(extractor=SQLAlchemyExtractor(),
                   loader=FsNeo4jCSVLoader())

job = DefaultJob(conf=job_config,
                 task=task,
                 publisher=Neo4jCsvPublisher())

# run job
job.launch()
```

How is databuilder orchestrated?

Amundsen uses Apache Airflow to orchestrate Databuilder jobs

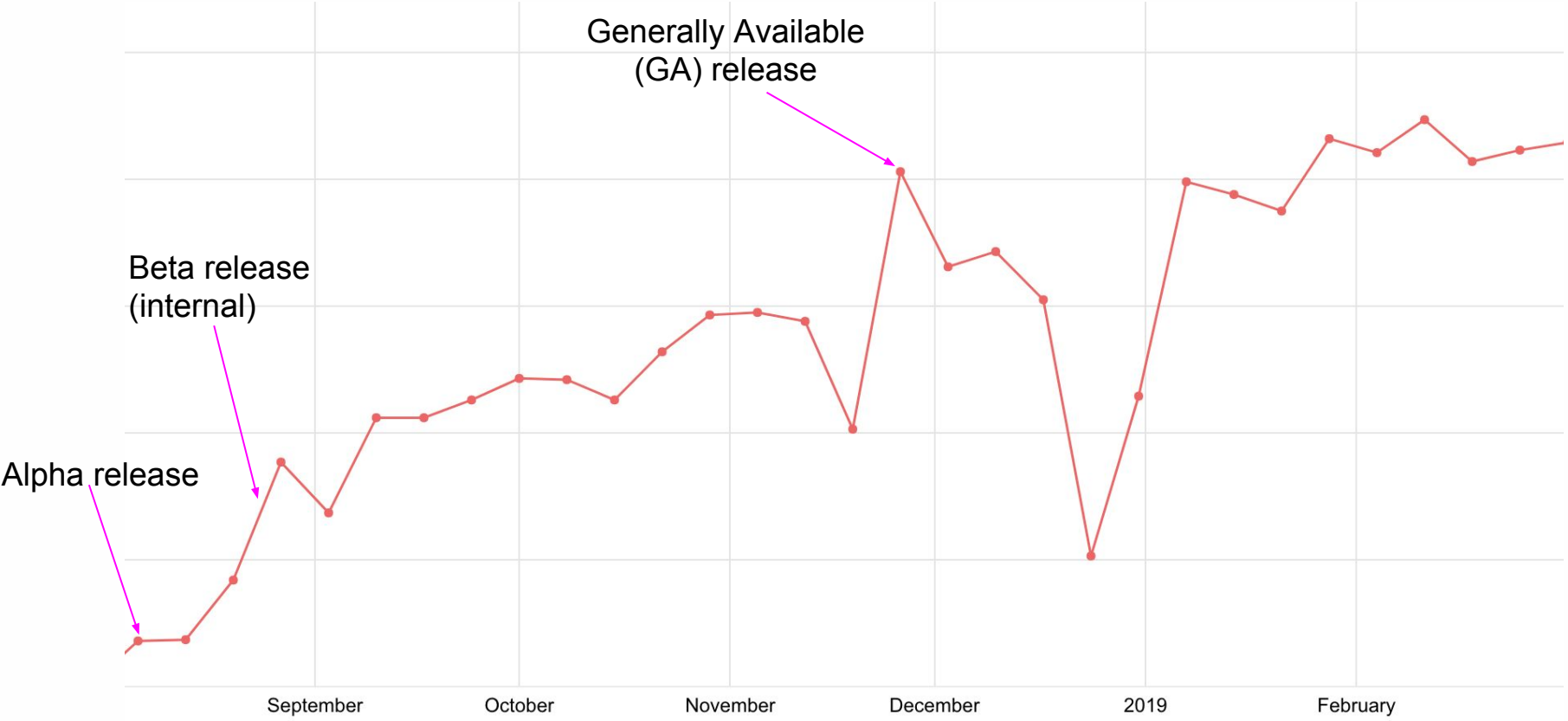


What’s next?

Amundsen seems to be more useful than what we thought

- Tremendous success at Lyft
  - Used by Data Scientists, Engineers, PMs, Ops, even Cust. Service!
- Many organizations have similar problems
  - Collaborating with ING, WeWork and more
  - We plan to announce open source soon

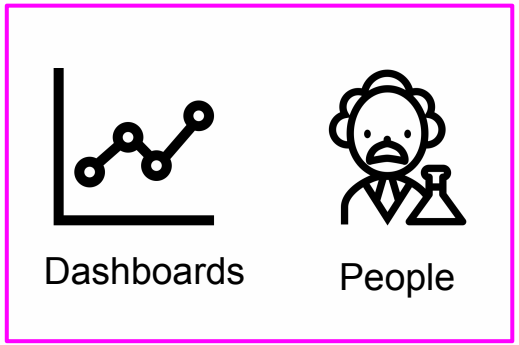
Impact - Amundsen at Lyft



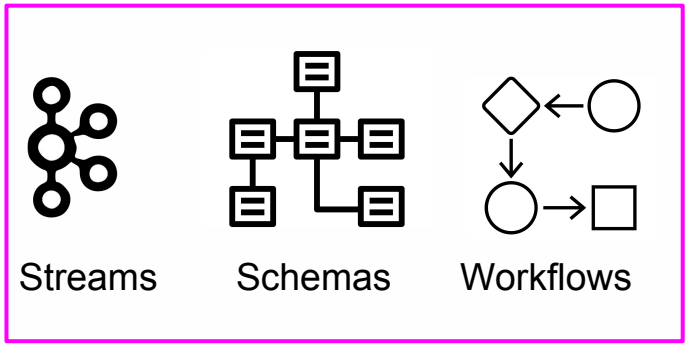
Summary

Adding more kinds of data resources

Phase 1
(Complete)



Phase 2
(In development)



Phase 3
(In Scoping)

Summary

- Data Discovery adds 30+% more productivity to Data Scientists
- Metadata is key to the next wave of big data applications
- Amundsen - Lyft's metadata and data discovery platform
- Blog post with more details: go.lyft.com/datadiscoveryblog

Jin Hyuk Chang | @jinhyukchang

Tao Feng | @feng-tao

Slides at go.lyft.com/amundsen_datacouncil_2019

Blog post at go.lyft.com/datadiscoveryblog

Backup