Tracking Data Lineage at Stitch Fix
Neelesh Srinivas Salian

Strata Data Conference - New York
September 12, 2018
Stitch Fix

Personalized styling service serving Men, Women, and Kids

Founded in 2011, Led by CEO & Founder, Katrina Lake

Employ more than 5,800 nationwide (USA)

Algorithms + Humans
About Me

STITCH FIX

Cloudera

Apache Software Foundation

Spark

Apache Flink
This talk

- Data Ecosystem
- Data Lineage
- The Need
- Challenges
- Approach
- Architecture
- Questions
Data Ecosystem
Data Lineage
The Need and Challenges
Key Terminology

Resource

- Structured Data - Hive Table
- Postgres Database

Job

- Service defined batch jobs
- Performs read/write on resources

ID - Unique identifier

- Service generated
- Synthesised

Event

- Read Resource
- Write Resource
Managing a Resource

- **Visibility** - Data Scientists need to know what could break.
  - Upstream and Downstream to a Resource
- **Effects of Change** - If a resource is modified what does it affect?
  - Schema change
  - Data type modification
- **Tracing** - How did we get to this resource - source to destination?
  - Journey of a resource
- **Debugging** - How can you reliably debug a large pipeline?
- **History** - What has been writing to this resource?
  - Historical information
Upstream and Downstream

Upstream to C
Resource A

Downstream to C
Resource D

Resource C

Resource B

Resource E
Traceability

1. Initiates
2. Reads
3. Reads
4. Writes
5. Reads
6. Writes
7. Writes
Challenges - Consistency

- Multiple services
- Different Job Representations
- Different points of concern
- Extractable information needs to be identified
Approach
# Simplifying the Data Model

<table>
<thead>
<tr>
<th>Owner (User/ Team)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job</td>
</tr>
<tr>
<td>Parent Job</td>
</tr>
<tr>
<td>Read Resource / Write Resource</td>
</tr>
</tbody>
</table>
Augmenting Code

- Avoid breaking API Changes
  - If any, there needs to be better communication
- Augment with necessary information to pass to Data Ingestion pipeline
- Most of the changes are backend libraries
- Idempotency in workflows
  - Behavior
  - Function
Architecture
Data Acquisition

Event Driven
- Using the Data Ingestion pipeline
- A Custom S3 Sink to write to Hive table
- Clients can send lineage information

Scheduled
- Ad-hoc usage
- Use only if additional information is needed
- Harder to maintain
Event Driven

Service

Read/ Write Event

Data Ingestion Pipeline

Kafka Topic

Consumed

Sink

Hive Table
Intermediate Data Collection

Resource Attributes
- database
- table
- batchId

Service Data Attributes
- owner
- jobId
- serviceName
- parentId

Hive Tables
Presto Data Lineage

- Extract information from Queries
- Currently implemented
- Missing pieces
  - Parent-Child relationship
  - Augmenting various clients
Spark Data Lineage

- Adding ability to log reads and writes as the happen
- Move over to Parquet as the default FileFormat
- Augmenting library + clients to pass parentage information
Data Refinement

- Regular cadence of ETLs extracting Lineage information
- Output into clean Postgres Tables
- ETLs for
  - Aggregated Metric Extraction
  - Resource Relationships
User Interaction

- Dashboards for Resource Views
  - Showing Upstream and Downstream dependencies
- Static Views
  - Metrics from the Warehouse
- Dynamic Views
  - In-flux changes to Resources
- Custom dashboards can be built
Reach Out

neeleshssalian@gmail.com
Thank you!

https://multithreaded.stitchfix.com/