Self-Serve Reporting Platform on Hadoop

Shirshanka Das
Strata Singapore 2015
Reporting Pipelines

Ingest  Process  Serve  Visualize
Reporting at LinkedIn: Evolution

Sources
- Oracle
- Kafka
- Espresso
- External

Ingest
- INFA + MSTR + Scripts
- Custom
- Custom
- Custom

Process
- Jobs on Hadoop
- MySQL
- Voldemort
- Pinot

Serve
- on Teradata
- MSTR

Visualize
- Tableau
- Internal Tools
Infra Scale

Number of Hadoop clusters: 12
Total number of machines: ~7k
Largest Cluster: ~3k machines

Data volume generated per day: XX Terabytes
Total accumulated data: XX Petabytes
People Scale

Reporting Platform Team: ~10
Core Warehouse Team: 1x

Data Scientists: 10x
Business Analysts: 10x
Product Managers: 10x

Sales and Marketing: 100x
Challenges

Disjointed efforts, unreliable systems
Unpredictable SLA across all systems
Fragmented data pipelines with inconsistent data
Ingest  Process  Serve  Visualize
Houston we have a problem
Step 1
Central transport pipeline
Still have a problem
Step 2

Central Ingestion Framework
Diverse Sources

- SFTP
- REST
- JDBC
- Amazon S3
- Salesforce
- MySQL
- Google
- Oracle

Stream + Batch

- Open source @ github.com/linkedin/gobblin
- In production @ LinkedIn, Intel, Swisscom, NerdWallet

Data Quality

- @LinkedIn
- ~20 distinct source types
- Hundreds of TB per day
- Hundreds of datasets
Ingest

Process

Serve

Visualize

Unified
Metrics
Platform
Requirements

Single Source of Truth

Easy Onboarding

Operability
Workflow

1. Iterate
2. Create
3. Review
4. Check in

Central Team, Relevant Stakeholders

Metric Owner

Sandbox

Metric Definition

Code Repository

Build

Core Metrics Job

System Jobs
Metric Definition

- Name
- Description
- Owners
- Tags
- Dataset Dimensions
- Time
- Tier
- Entity Dimensions
- Entity Ids
- Metrics
- Temporality
- Input Datasets
- Script
- Formulas
An example: video play analysis

| name: "video" |
| description: "Metrics for video tracking" |
| label: "video" |
| tags: [flagship, feed] |
| owners: [jdoe, jsmith] |
| enabled: true |
| retention: 90d |
| timestamp: timestamp |
| frequency: daily |
| script: video_play.pig |
| output_window: 1d |

input_datasets:
[
  {
    name: actionsRaw
    path: Tracking.ActionEvent
    range: 1d
  }
]

dimensions:
[
  {
    name: platform
    doc: "phone, tablet or desktop"
  }
  {
    name: action_type
    doc: "click play or auto-play"
  }
]
entity_ids: [
  {
    name: member_id
    category: member
  }
  {
    name: video_id
    category: video
  }
]

metrics: [
  {
    name: unique_viewers
    doc: "Count of unique viewers"
    formula: "unique(member_id)"
    tier: 2
    good_direction: "up"
  }
  {
    name: play_actions
    doc: "Sum of play actions"
    formula: "sum(play_actions)"
    tier: 2
    good_direction: "up"
  }
]
UMP by the numbers

First version in production since early 2014
Significant redesign in 2015

Total amount of data being scanned per day: Hundreds of TBs
Total number of metrics being computed: 2k+
Total number of scripts: ~ 400
Number of authors for these metrics: ~ 200
Maximum number of dimensions per dataset: ~ 30
Number of people responsible for upkeep of pipeline: 2
Learnings so far

Ease of onboarding
Hard when you have > 1000 users with different skill sets
Need great UX to complement developer friendly alternatives

Single source of truth
Not just a technology challenge
Organization needs to rally around it

Operability
Multi-tenant Hadoop pipeline with SLA-s and QoS: hard
Cost 2 Serve: Managing metrics lifecycle is important

The Next Big Things
Bridging streaming and batch
Code-free metrics
Sessions, Funnels, Cohorts
Open source
Capabilities

- SQL-like interface (minus joins)
- Sub second query latency
- Data load from Hadoop and Kafka
Pinot Data Flow

Pinot

Samza

Kafka

Process

Hadoop

minutes

hour +
Pinot@LinkedIn
In production since 2012
Open source @ github.com/linkedin/pinot

Site-facing Apps
Reporting dashboards
Monitoring

ResourceMem (MB) by UserID
By UserID

13.3M
ResourceMem
Standardize Visualization

Leverage
- Standalone app, with support for embedding
- Can use existing analytics backend: Pinot

Strategic
- Reduces dependency on 3rd party BI tools
- Closer integration with LinkedIn’s ecosystem of experimentation, anomaly detection solutions
Requirements

Core Visualization Capabilities

Metadata Integration

Support apps ecosystem
Raptor 1.0

First version built by 3 engineers in a quarter

Features
- Integration with UMP, Pinot
- Time series, bar charts, …
- Create, Publish, Clone, Discover
  Dashboards

Numbers
- Number of dashboards: ~100
- Weekly unique users: ~400
The Future for Raptor

Social Collaboration features
Intelligence
- Anomaly detection
- Dashboards You May Like
Embedding into data products
Open Source
A Few Good Hammers

Ingest

Process

Serve

Visualize

Unified Metrics Platform

Pinot

Raptor
What we’re excited about

Ingest  Process  Serve  Visualize

Unified Metrics Platform

Pi not  Raptor

Metadata Bus
Metadata driven e2e Optimizations

Dynamic prioritization of data ingest
Surface source data quality issues in dashboard
Surface backfill status on dashboard
Cascading deprecation of dashboards, computation and data sources through lineage
Catch me offline to chat about…

What we’re doing for
- Views on Hadoop
- Data Quality
- Metadata