Analyzing Operational Data at Scale using ML at Intuit

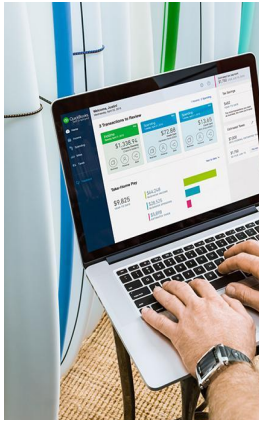
Amit Kalamkar
Vigith Maurice

Overview

- Intuit background
- Objective
- Problem
- Operational Data Lake
- Applications on Operational Data Lake
- Fuzzy Demo!
- What’s next

Who we are

1983	1993	5,000	21	$7.7B FY19	50M
Founded	IPO	Developers	Locations	Revenue	Customers

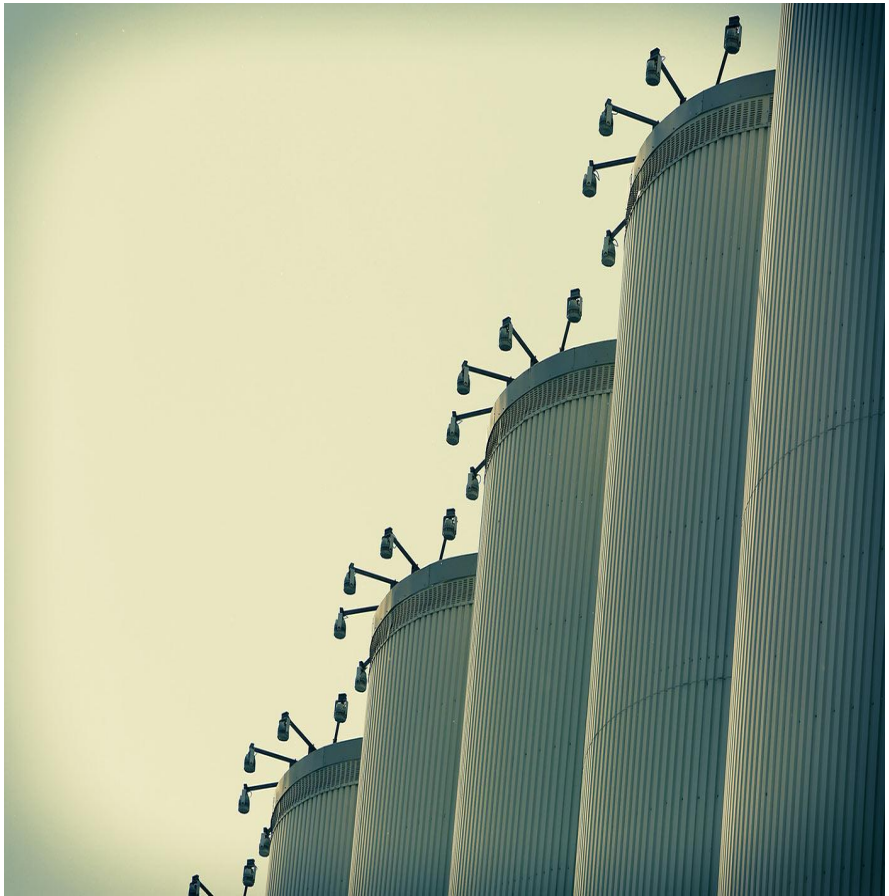


Kubernetes at Intuit

- 1200+ Services in Prod
- 200+ clusters (Intuit managed)
- 10k+ nodes
- 85k+ cpu cores
- 120k+ pods
- Open source Projects- **Argo** (CNCF incubation), **Keiko**, **Admiral**

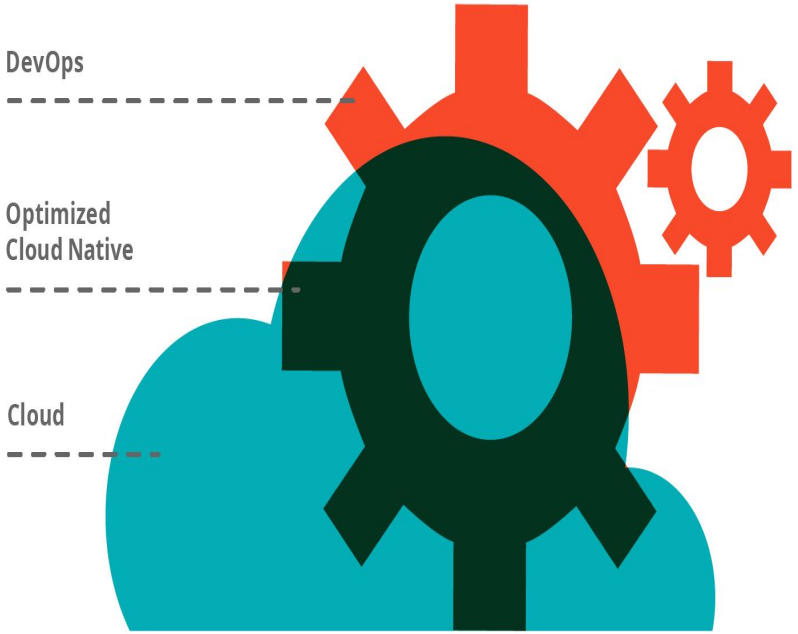
Objective- Data driven insights

Problem



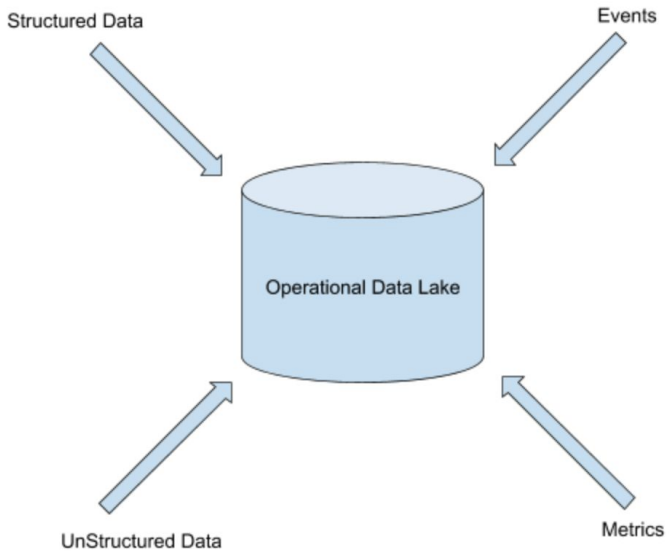
- **Silos of disjoint operational data** generated by different tools for providing **meaningful insights**
- **No standard way to correlate** and process huge volumes of operational data
- Too many dashboards with **no standardization across dashboards**
- No easy way to apply **standardized**, **scalable ML** on high data volume

Now is the time



- Increased complexity due to **dynamic cloud native platform** with many inter-dependencies
- Potentially increased risks with **greater deployment frequency** on dynamic, distributed infrastructure
- **DevOps paradigm** requires specialized, data driven tooling for operability and observability

Operational Data Lake (ODL)



- Build a **warehouse for clean, documented,** and **schematized operational data**
- Collects and publishes all aspects of an **application's operational lifecycle data** including development, build & test, production, security, monitoring, usage, to enable **operational analytics** and **AIOps**
- **Democratized Operational data** by any consumer at Intuit

Drives automation and faster, better operational decisions (e.g. incident response)

Self Service Analytics Platform

Collection (Streaming)

Processing (with curation)

Realtime ML and Analytics

Storage and Retrieval

Self Service Platform

Collection

- Real time Ingestion through **Kafka** stream
- Batch loading of **structured Logs** using Cribl
- **Data Governance** in built and moderated by data stewards

ML and Analytics

- **Standardized ML platform** trained to detect anomalous measurements
- **Highlight anomalous behaviour** in metrics and data
- **Guided debugging** using anomaly score

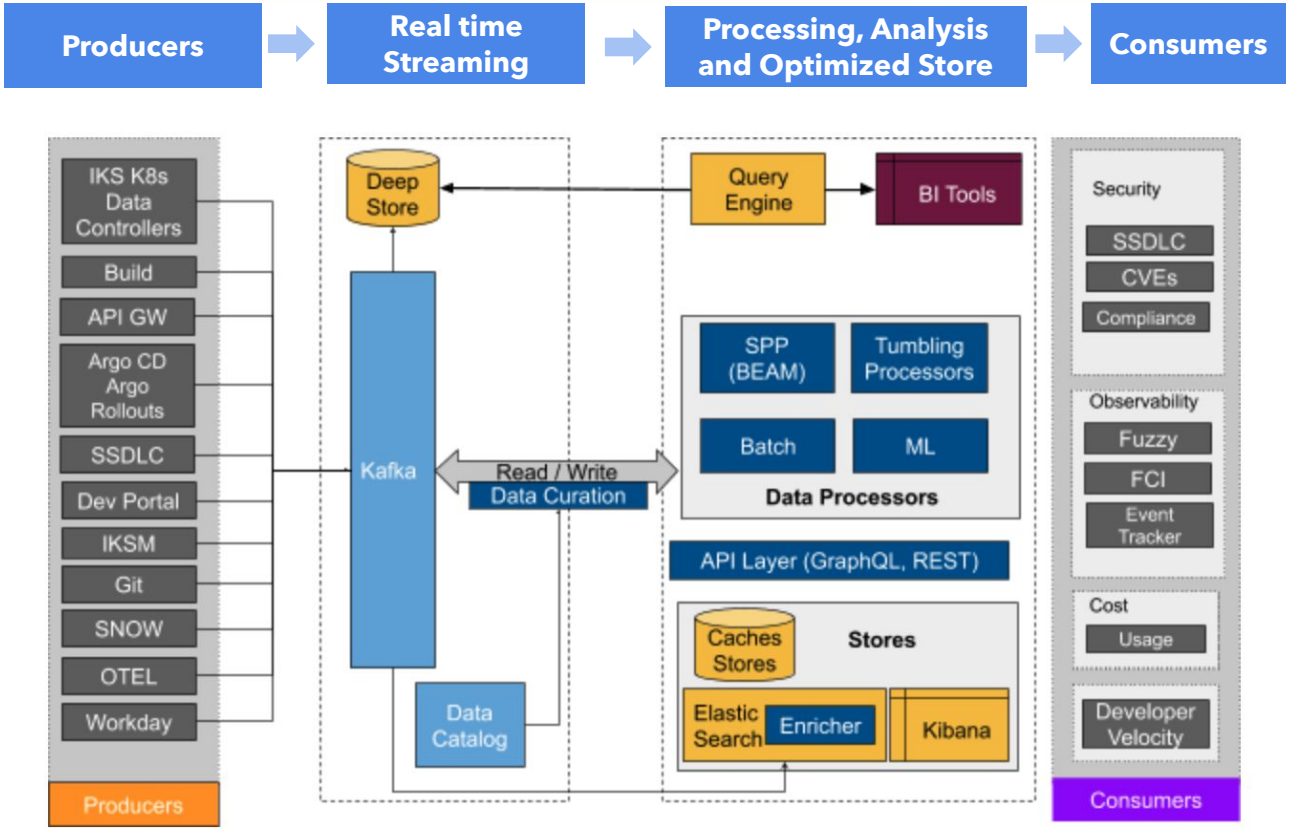
Processing

- **Cataloging (Apache Atlas)** and **scrubbing** to remove and standardize fields
- **enrichment** using golden entity attribute (AssetID)
- **stream processing (Apache Beam)** for source JOINS and Summarization

Storage and Retrieval

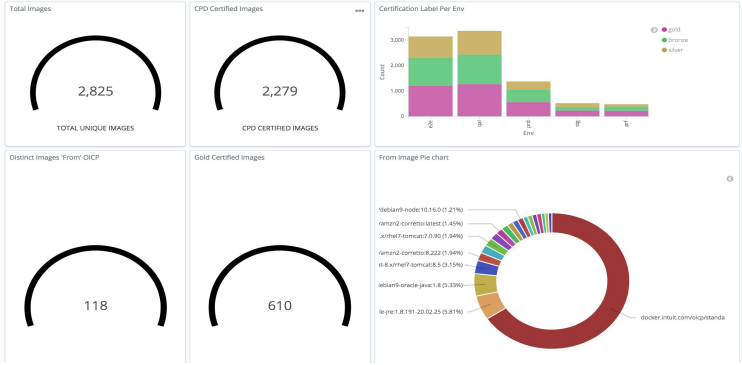
- Standardization for **visualizing** and **Triaging** capabilities over generic UX
- **Interactive exploration** using OLAP (Druid) querying on real time data
- **Long term stores** (S3/ELK) for batch training

ODL Architecture

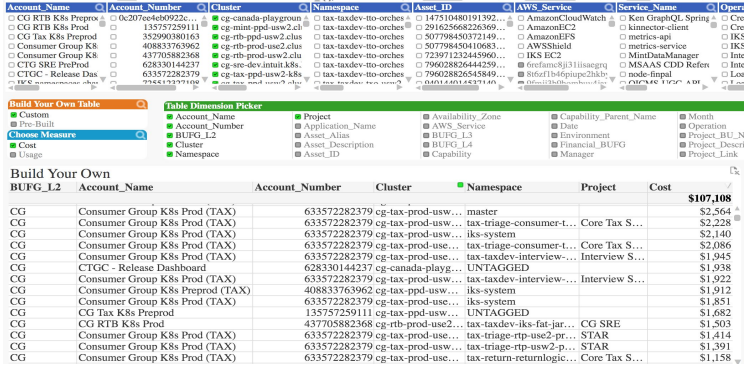


Applications on top of Operational data

KubeCon CloudNativeCon North America 2020 Virtual



SSDLC REPORTING AND ENFORCEMENT



COST REPORTING AND ANALYSIS

DEVELOPMENT VELOCITY

FUZZY (Observability)

Fuzzy - Observability Application Deep Dive

Fuzzy- Speed up incident response

- **Mean time to detect** is too long
- Not able to quickly **ascertain customer impact**
- Not able to quickly **isolate the source and root cause** of an incident
- Large **latency due to exponential growth** in operational data.

Fuzzy- Charter

- Make it obvious **what services are affected**
- Make it obvious **what is the causal service**
- Make it obvious **what is the causal event**

- Enable **non-experts** to triage problems
- **Reduce MTTI** by order of Magnitude
- **Reduce MTTR**

How?

Collection

- Collect **AWS CloudWatch** and **CloudTrail** using Kafka
- K8s **Audit logs, Objects** and **Events using Data Controllers**
- Application **Timeseries Metrics** (Kafka)
- Sampled Open Telemetry **Traces** (Jaeger)

ML

- **Unsupervised ML models** trained to detect anomalous measurements per service
- **Highlight anomalous behaviour** in per service and dependent services

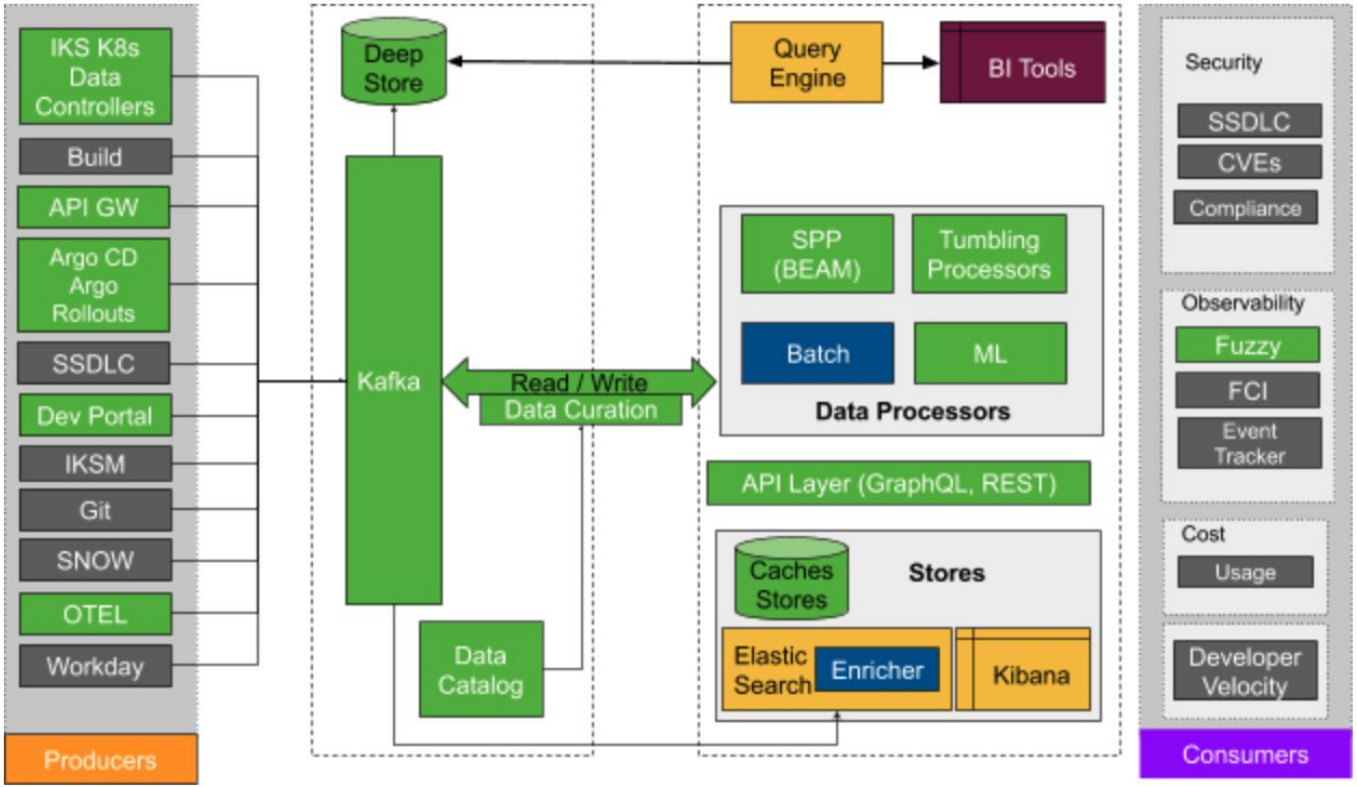
Processing

- **Windowing** (Fixed and Sessions) for summarizations
- **Correlate** metrics using AssetID
- Write to **multiple stores** for quick retrieval (Redis/ES/Druid)

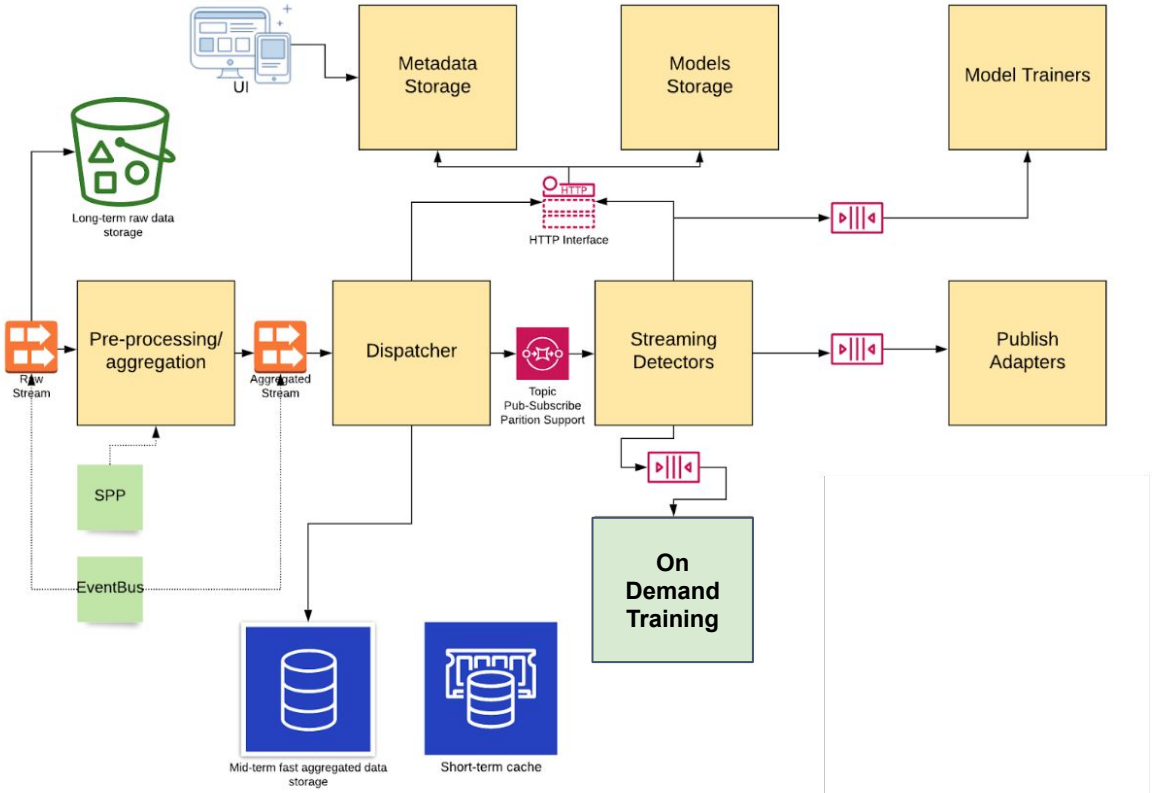
Analytics

- Personalized view for the services with **Guided debugging** using anomaly score
- **Hierarchical view** (Intuit, BU, Scrum) to group the impact at services level.
- Expose data over **GraphQL** for everyone

Fuzzy Architecture



ML Architecture



Demo

What's Next

- Support for onboarding Custom Metrics (BYOM)
- Self Serve Capability for open ended debugging (Unknown - Unknown)
- Expand on-demand Anomaly score using ML
- Open Source

Thank You