

# **Sunbird: Infrastructure for Knowledge Sharing**

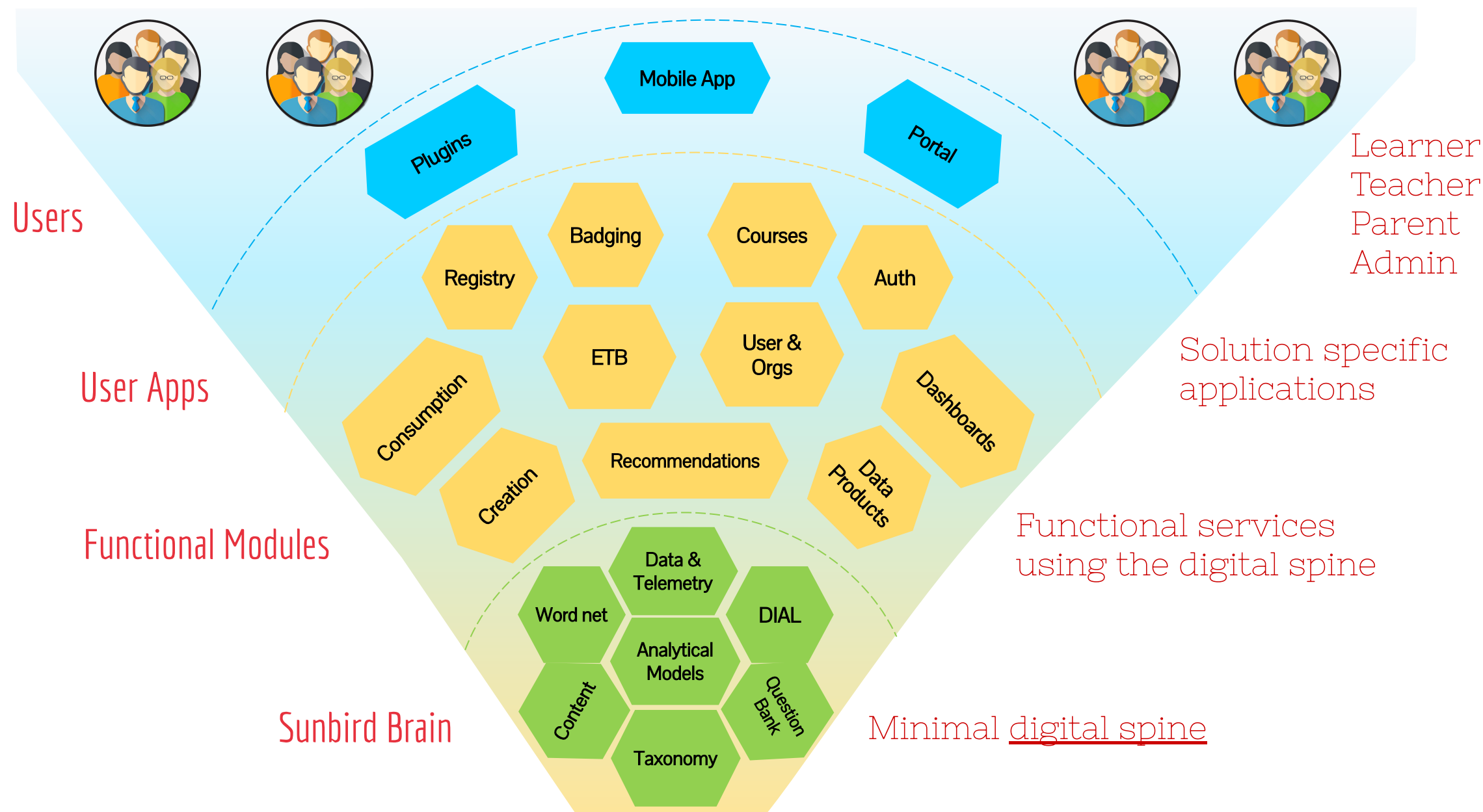
## **Technical Overview**

---

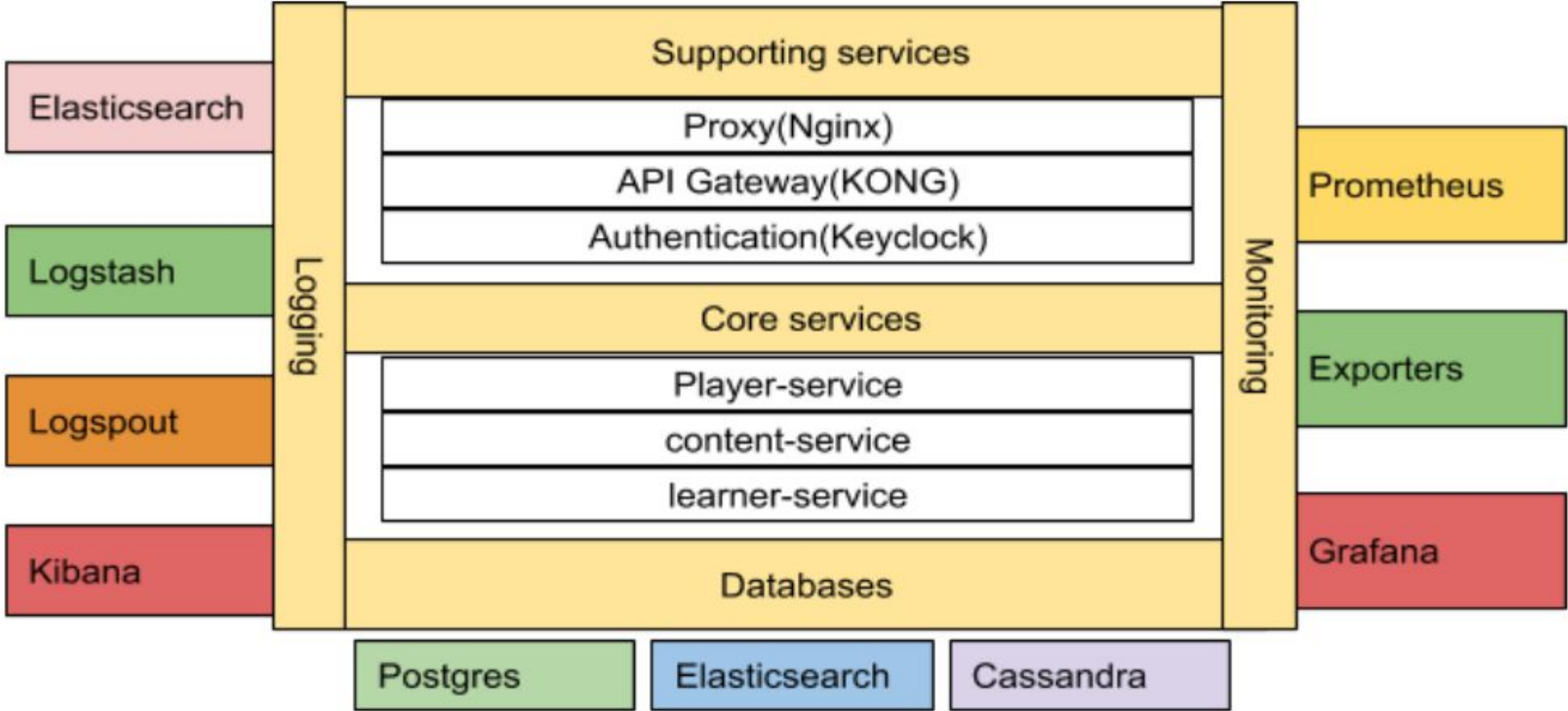
# Sunbird Subsystems & Services

- Sunbird Microservices
- Sunbird Data Pipeline
- Sunbird Reporting Architecture

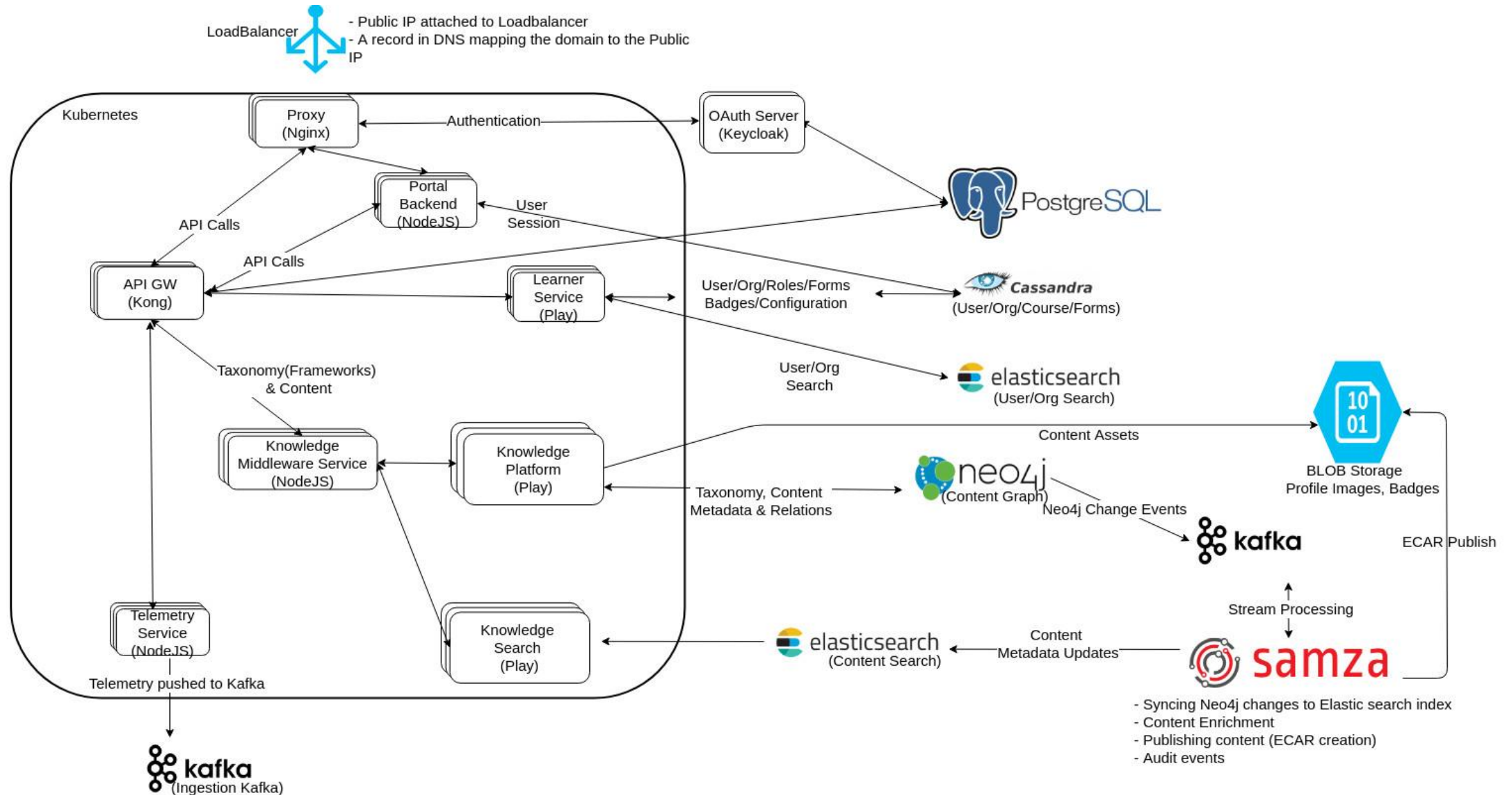
# Layered Platform



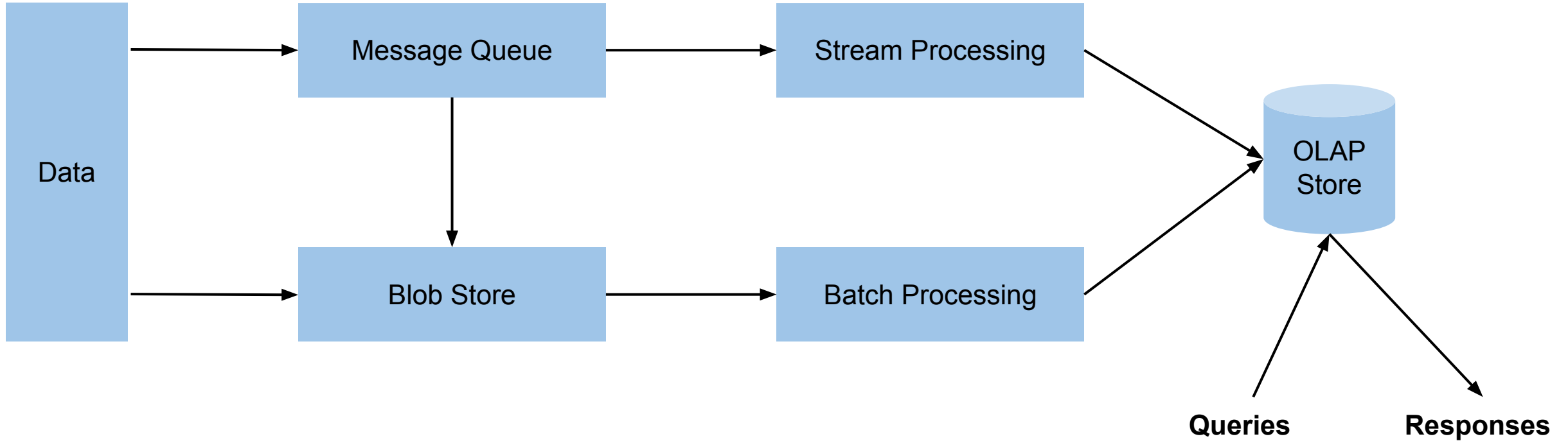
# Sunbird Microservices



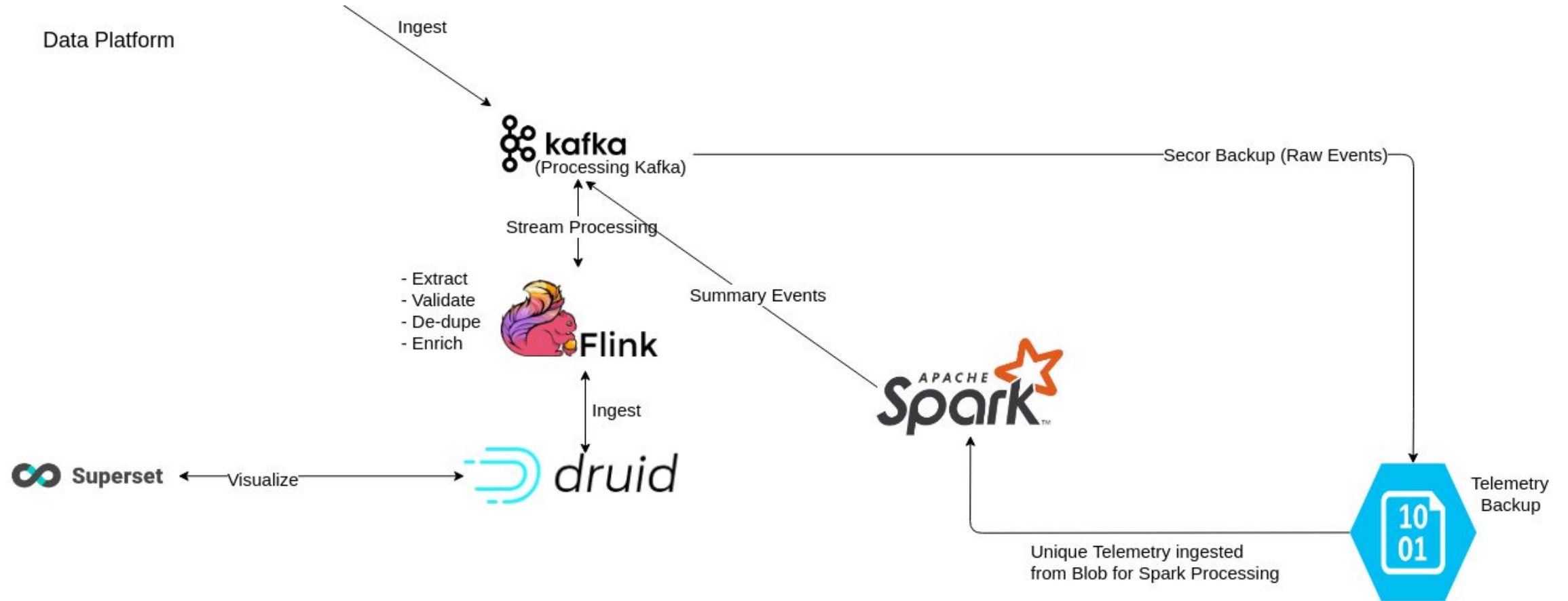
# Sunbird component view - Microservices



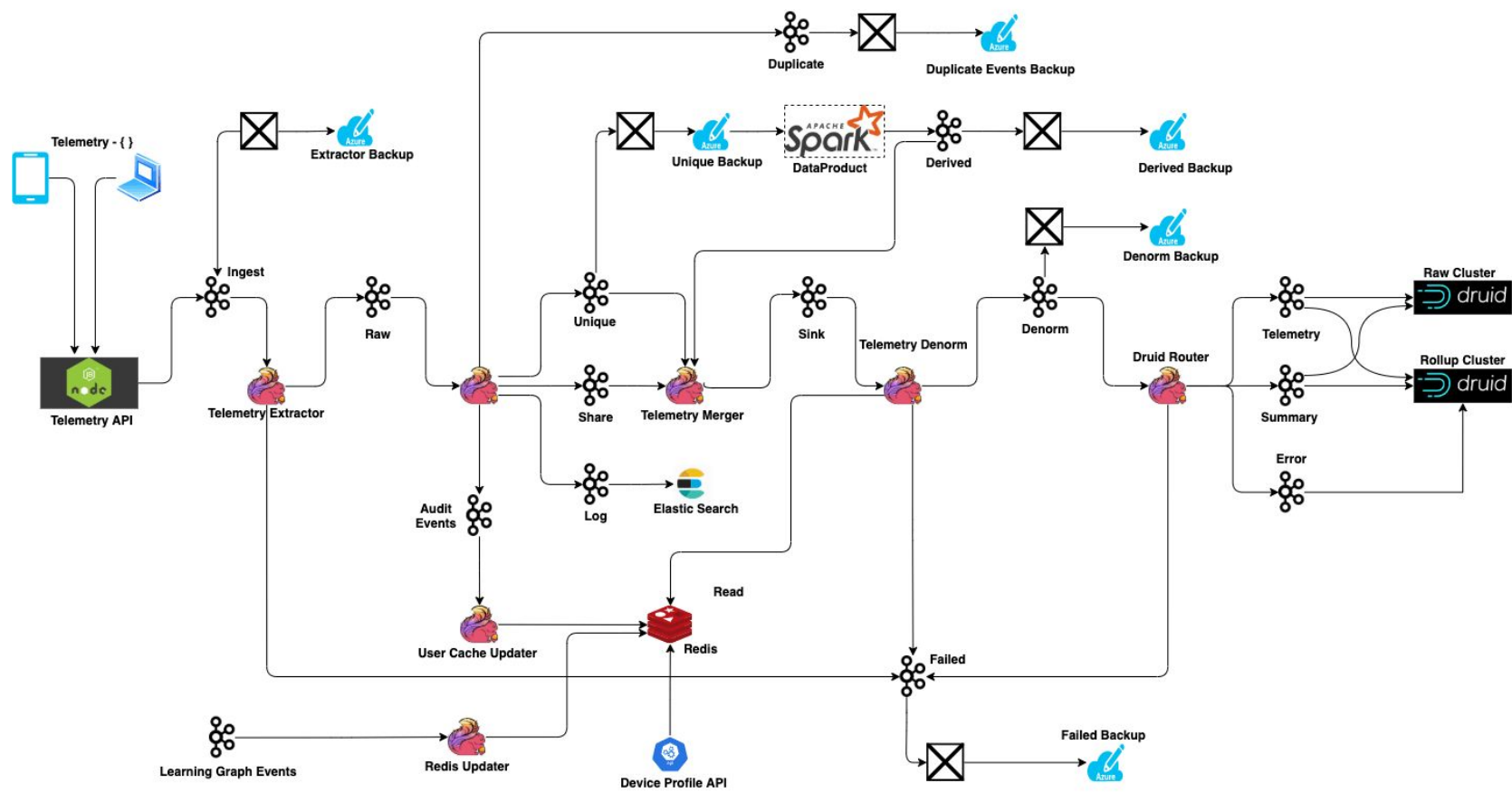
# Sunbird Data Platform - Lambda Architecture




# Sunbird component view - Data Platform



# Sunbird Data pipeline Stages



SUNBIRD ANALYTICS DATA PIPELINE

**JOBS SUMMARY**


**Extractor**  
1. Extract the Batch events  
2. Removed the duplicated batch events


**Pipeline Processor**  
1. Telemetry Validation  
2. Telemetry De-Duplication  
3. Router "SHARE", "LOG" Events to separate topic


**Telemetry Merger**  
1. Flatten the SHARE Events  
2. Merge Flattened share, unique, derived events into sink topic,


**Telemetry Denorm**  
1. Location  
2. Content & DialCode  
3. User

**Druid Router**  
1. Validate the de-normed events  
2. Route ERROR, SUMMARY, TELEMETRY Events to different topic  
3. Ingest the events Druid


**Telemetry Sync API(Node js)**


**Spark Jobs (Data products)**


**Druid Data Store**


**Flink Jobs**

**Kafka**

**Azure Storage**

**Secor process**

**Elastic Search**

**Redis**



# Sunbird Data Platform - Key Design Elements

## ➤ Generalized Telemetry Specification

- A set of 17 event types to capture data for all possible use-cases
- A common envelope for all events to capture contextual information
- All services and client apps produce telemetry as per spec
- Extendable specification for further denorm within the pipeline
- <https://github.com/sunbird-specs/Telemetry/tree/3.3.0>

## ➤ Detailed instrumentation design

- To be able to generalize the summarization
- Build sdk's to auto-generate bulk of the instrumentation

## ➤ Workflow Summarizer

- Auto summarize the input events
- Reduce the query events by a factor of 100
- <https://github.com/project-sunbird/sunbird-analytics/wiki/%5BData-Product%5D-Workflow-Summarizer>

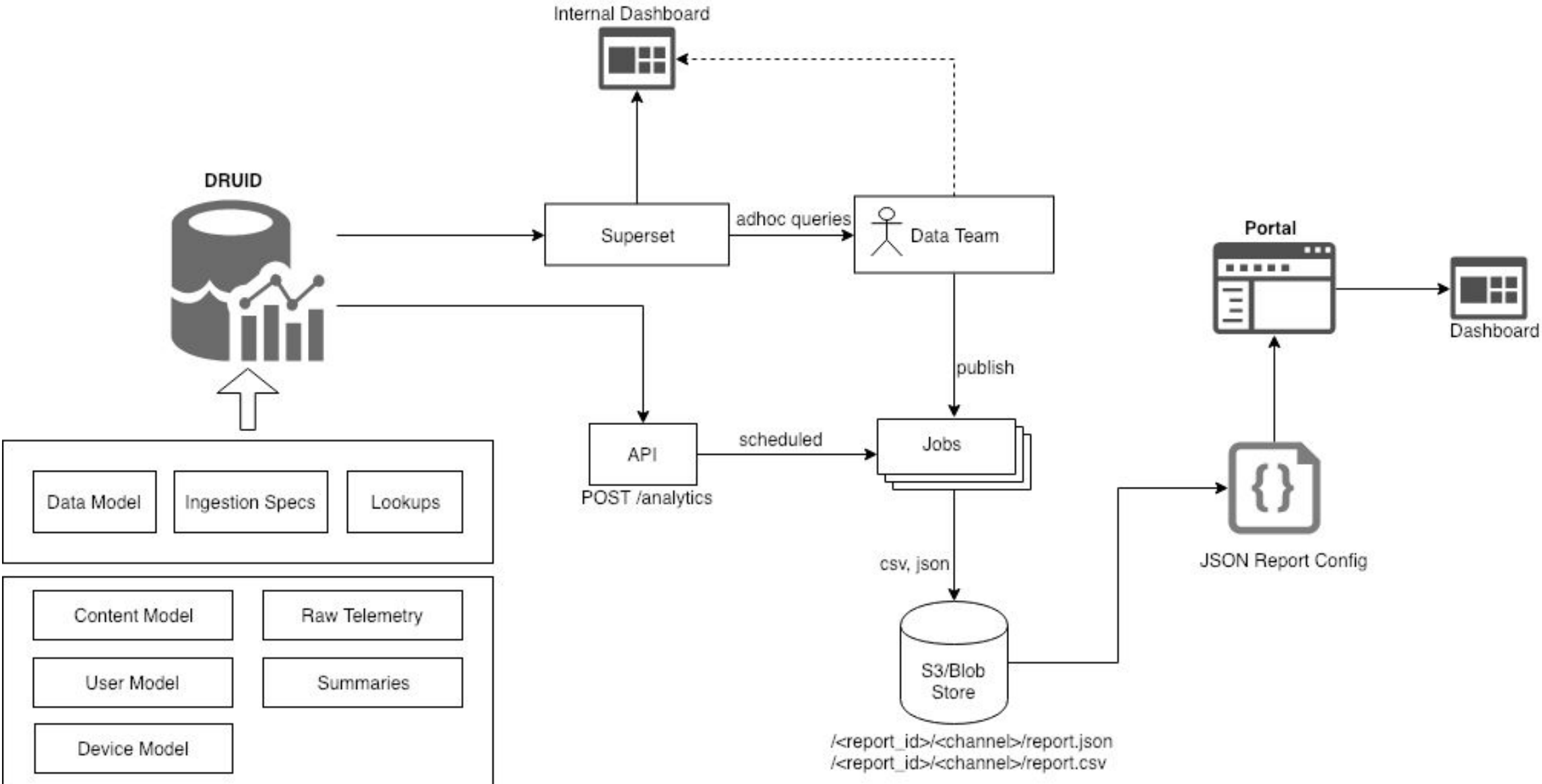
## ➤ Analytics Framework for DART

- Debuggability, Auditability, Replayability and Traceability

# Sunbird Data Platform - Diksha Instance

- Processing around 1.5B+ events (~5 TB) per day
- Server Capacity - Kafka Clusters
  - Split into two clusters for high reliability
  - Ingestion Cluster - 3 node cluster (8cpu, 16gb, 8 TB)
  - Processing Cluster - 5 node cluster (8cpu, 16gb, 16 TB)
- Server Capacity - Flink Cluster
  - 42 nodes (8cpu, 16gb)
- Server Capacity - Spark Cluster
  - 2 head nodes (4cpu, 32gb)
  - 16 worker nodes (8cpu, 64gb)
- Server Capacity - Druid Clusters
  - Split into two clusters for varying access patterns
  - Raw Cluster - 12 nodes (8cpu, 32gb)
  - Rollup Cluster - 4 nodes (8cpu, 32gb)

# Sunbird - Reporting Architecture



# Tech Stack

NGINX

ionic

node  
JS

ANGULAR

Java

KEYCLOAK

Jenkins

HTML  
JS  
CSS

play

CODACY

Kong

docker

Grafana

Semantic UI

akka

Scala

neo4j

ANSIBLE

kibana

APACHE CORDOVA

druid

spring

Superset

samza

kubernetes

Azure

badgr

python

Apache Airflow

APACHE Spark

APACHE kafka

PostgreSQL

elasticsearch