Feathr: The Enterprise-Grade, High Performance Feature Store

David Stein, Senior Staff Engineer, LinkedIn
Xiaoyong Zhu, Principal Data Scientist, Microsoft

FEATURE STORE SUMMIT 2022

Organized by HOPSWORKS

feathr

an open source,
enterprise-grade,
high-performance feature store

- built at LinkedIn
- in collaboration with Microsoft Azure
- a Linux Foundation AI&Data Sandbox project

Agenda

1. Why we built Feathr
2. What a feature store should be
3. Feathr at LinkedIn
4. Feathr on Azure – Demo

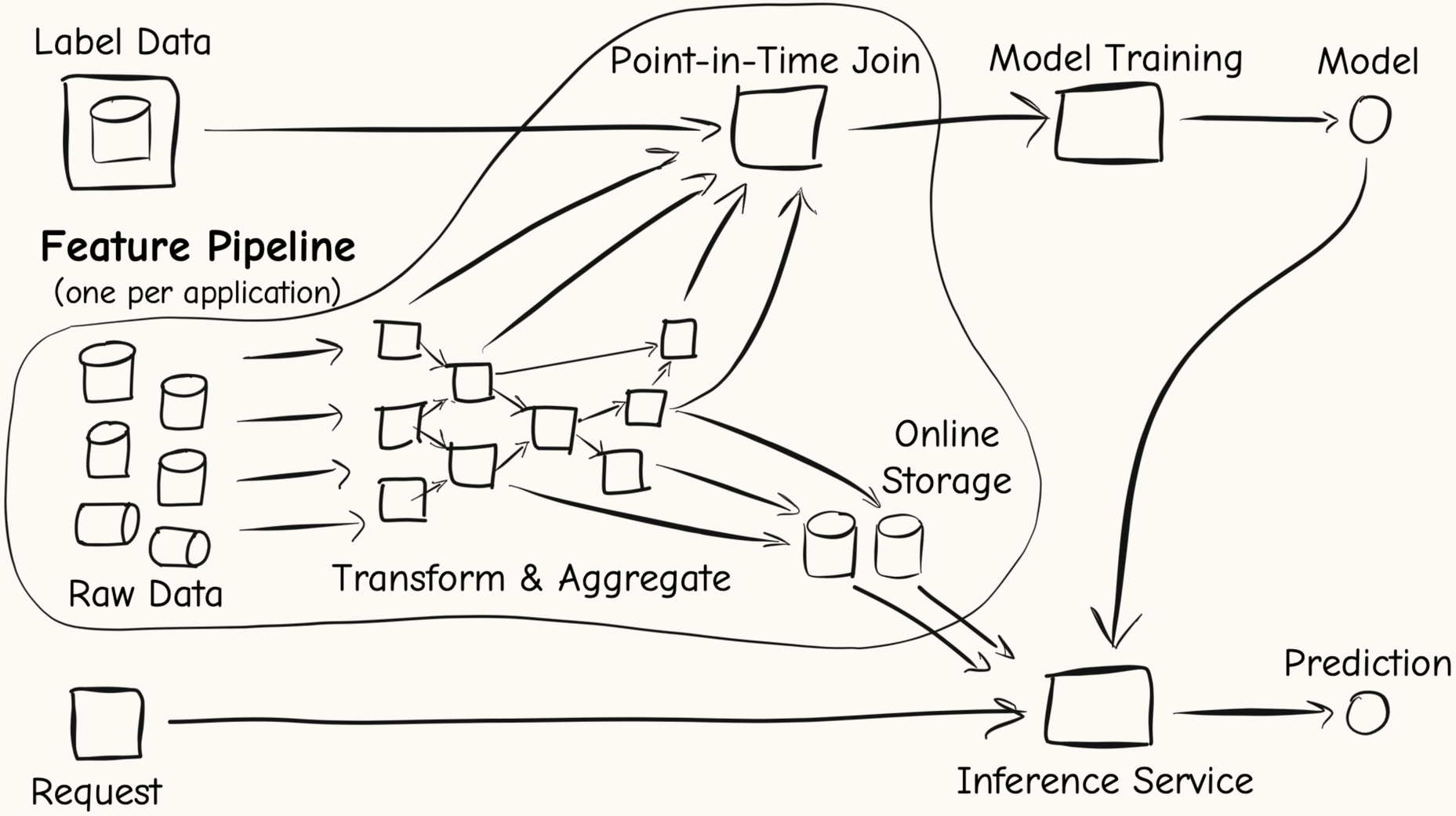
Why we built Feathr

Getting features into ML models should be easy.

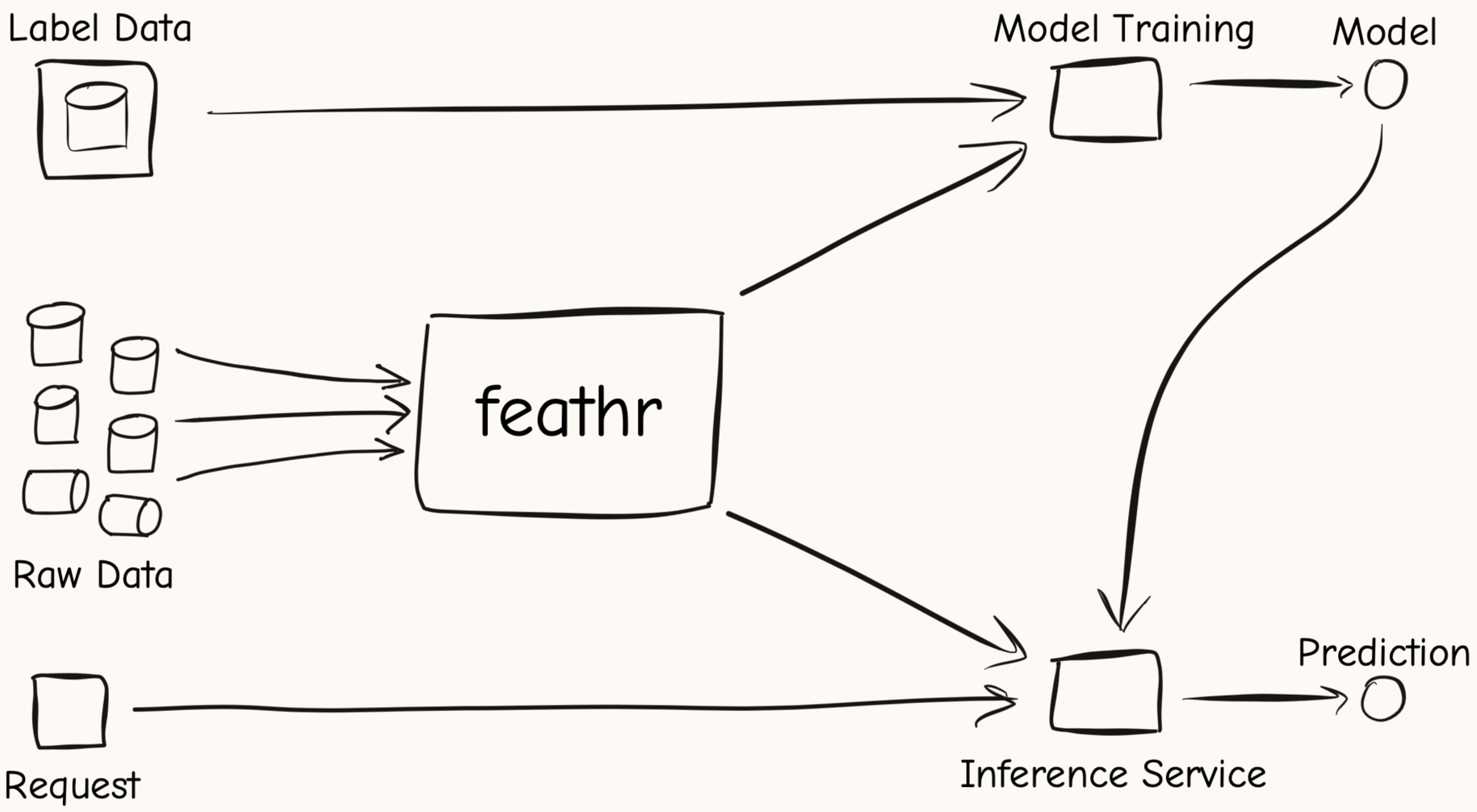
Like a music streaming app … for feature engineering

	Music "Workflow"	ML Feature Workflow
Old Way	- Manually get music files from various sources - Convert them to a format my device can play - Load onto my device (different for home/car) - Worry about storage, bitrate, compatibility	- Write jobs to get entity data from various sources - Extract, aggregate, join, convert into proper format - Load into model framework (different for train/serving) - Worry about scale, perf, leakage, train/serve skew
New Way	- **Just ask virtual assistant to play the song.**	- **Just import the feature by name into model code.** - If feature doesn't exist, define and register it via simple APIs.

Problem: Feature preparation is complicated



Solution: Feathr feature store



Like a package manager for feature engineering

Code

```
import module1
import module2
import module3
import module4
```

Features

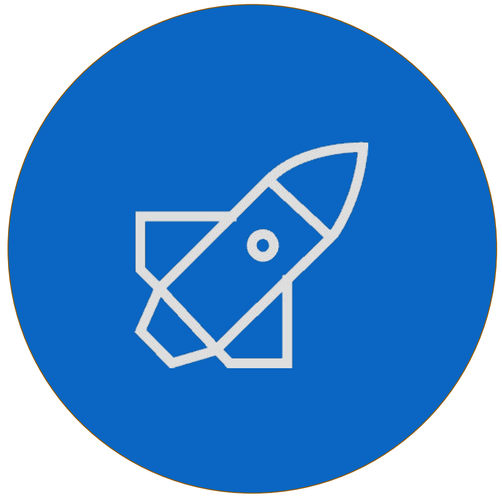
```
query = FeatureQuery(
  feature_list=[
    "feature_1",
    "feature_2",
    "feature_3",
    "feature_4"
  ],
  key=item_id)
```

What a feature store should be

Feature store principal use cases

Develop Features

Based on raw data, using simple APIs



Deploy Features

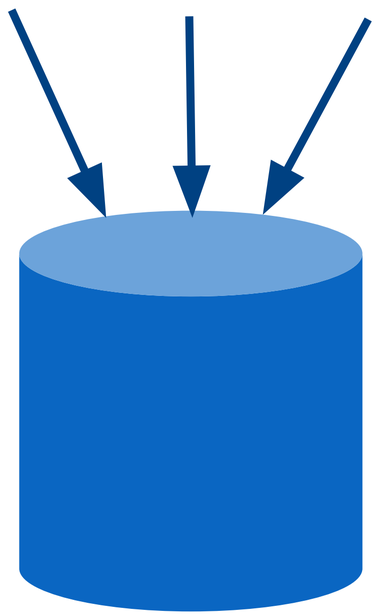
For training and online model inferencing

Manage Features

Monitor feature health and share across teams

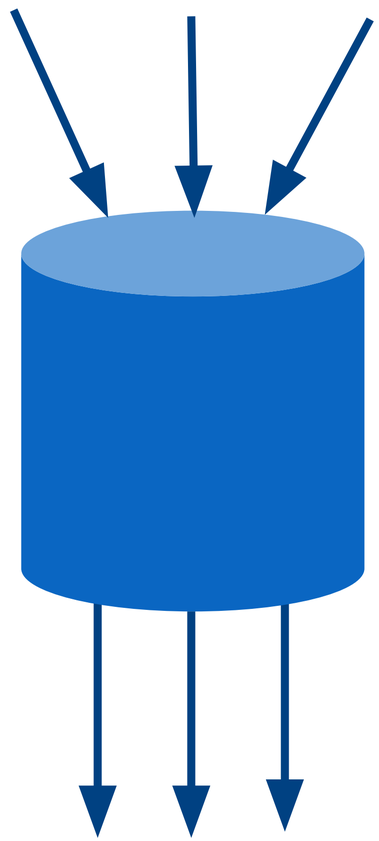
The “feature store” abstraction

- “Put a feature in” (Producer)
  - Develop a feature based on **raw data sets**
    - Sliding time windows
    - Aggregations
    - Transformations
    - Lookups/joins
  - Develop a feature based on other feature(s)

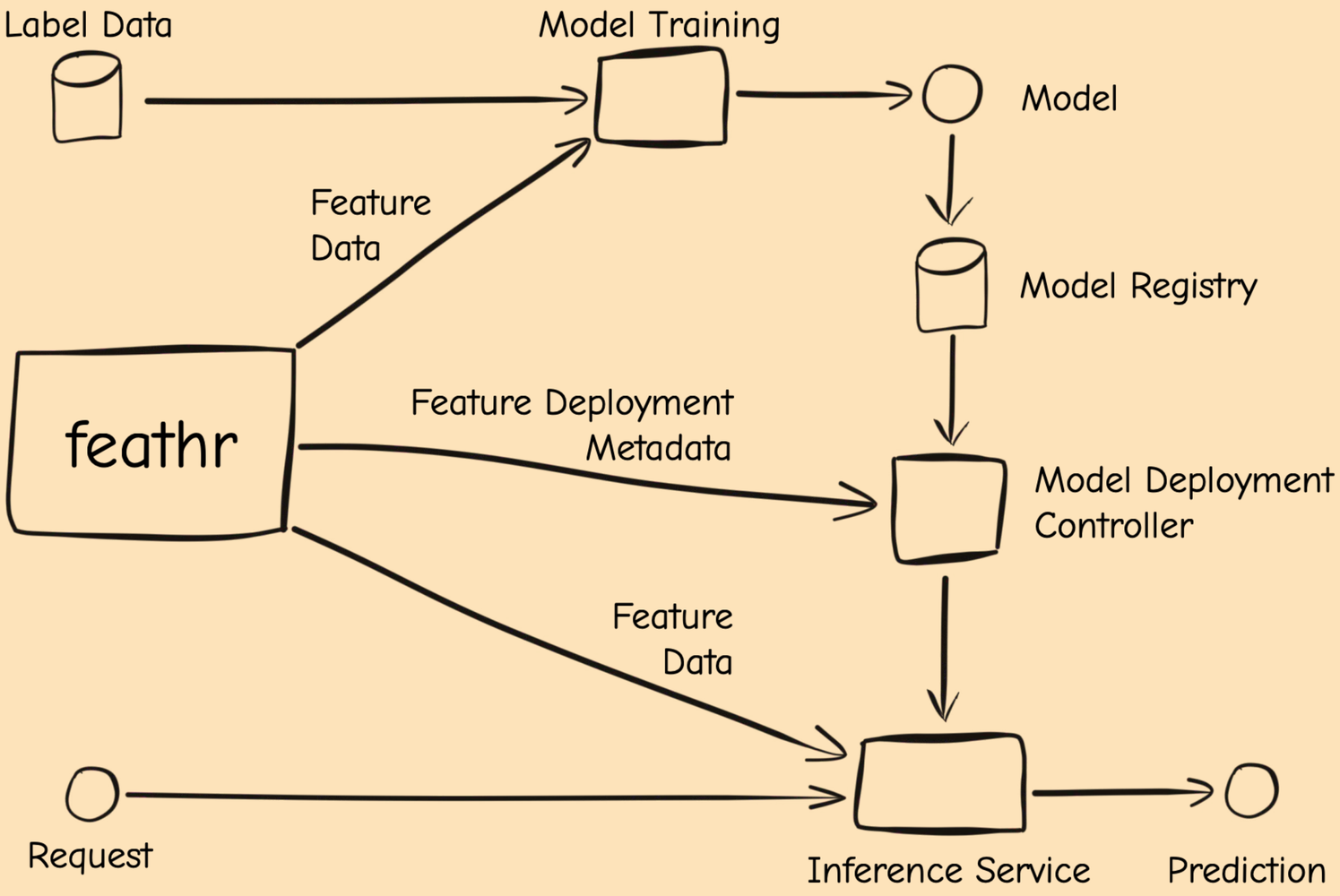


The “feature store” abstraction

- “Put a feature in” (Producer)
  - Develop a feature based on **raw data sets**
    - Sliding time windows
    - Aggregations
    - Transformations
    - Lookups/joins
  - Develop a feature based on other feature(s)
- “Get some features out” (Consumer)
  - Join features to training labels
  - Backfill historical values of features (**point-in-time correctness**)
  - Efficiently compute, store, and serve features for **online inference**



Feathr at LinkedIn



Feathr is a pillar of LinkedIn's ML platform

Model deployment service uses Feathr to ensure a model's feature dependencies are deployed, before deploying the model.

Feathr at LinkedIn

- hundreds of models
- thousands of features
- many kinds of entities (economic graph)
- petabyte scale

Timeline

2017 Initial development and launch

2018 Broad adoption within LinkedIn

2020 Majority of LinkedIn ML applications onboarded

2022 Open source, Azure collaboration, joined Linux Foundation AI & Data

Impact at LinkedIn

Majority of ML applications at LinkedIn have adopted Feathr

Improved Productivity

Faster experimentation with new features, from weeks to days

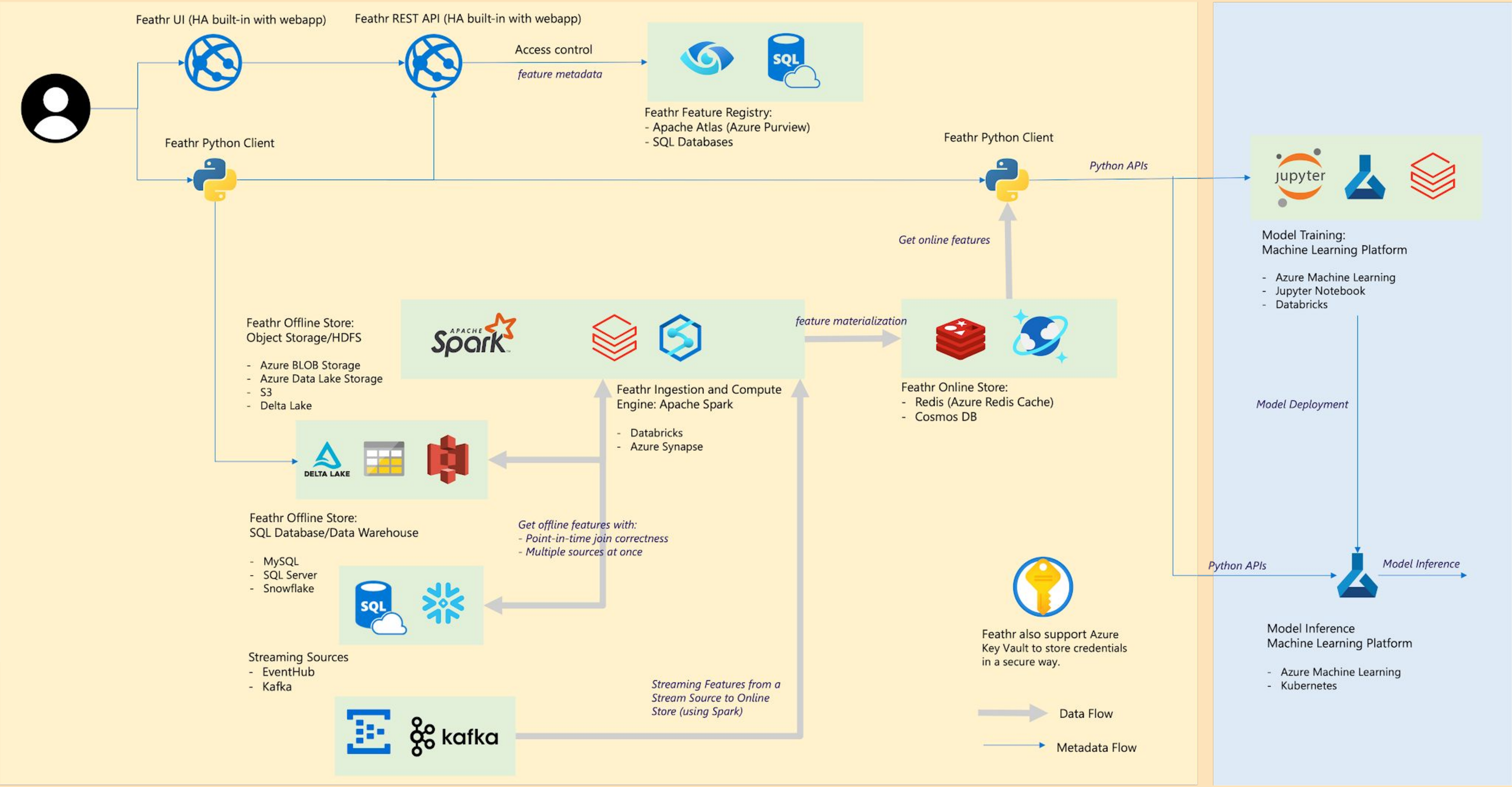
Improved Performance

Running time improved over custom pipelines, as much as 50%

Improved Collaboration

Applications can share features, which was difficult previously

Feathr on Azure Demo


Feathr UI (HA built-in with webapp)
Feathr REST API (HA built-in with webapp)
Access control
feature metadata
Feathr Feature Registry:
- Apache Atlas (Azure Purview)
- SQL Databases
Feathr Python Client
Feathr Python Client
Python APIs
Get online features
Model Training:
Machine Learning Platform
- Azure Machine Learning
- Jupyter Notebook
- Databricks
Feathr Offline Store:
Object Storage/HDFS
- Azure BLOB Storage
- Azure Data Lake Storage
- S3
- Delta Lake
feature materialization
Feathr Ingestion and Compute Engine: Apache Spark
- Databricks
- Azure Synapse
Feathr Online Store:
- Redis (Azure Redis Cache)
- Cosmos DB
Model Deployment
Feathr Offline Store:
SQL Database/Data Warehouse
- MySQL
- SQL Server
- Snowflake
Get offline features with:
- Point-in-time join correctness
- Multiple sources at once
Python APIs
Model Inference
Feathr also support Azure Key Vault to store credentials in a secure way.
Model Inference
Machine Learning Platform
- Azure Machine Learning
- Kubernetes
Streaming Sources
- EventHub
- Kafka
Streaming Features from a Stream Source to Online Store (using Spark)
Data Flow
Metadata Flow

Thank you

LinkedIn