FEATURE STORE SUMMIT 2025

On-Demand Feature Lifecycle Management

Organized by HOPSWORKS

About Me

- *Live in SLC, UT*
- *Data Science / Engineering Background*
- *Nine years in ecommerce*
- *Many titles, mostly the same fu*
- *About five years in specifically in online inference*
- *Passionate about ML/AI design patterns*

Clicklease

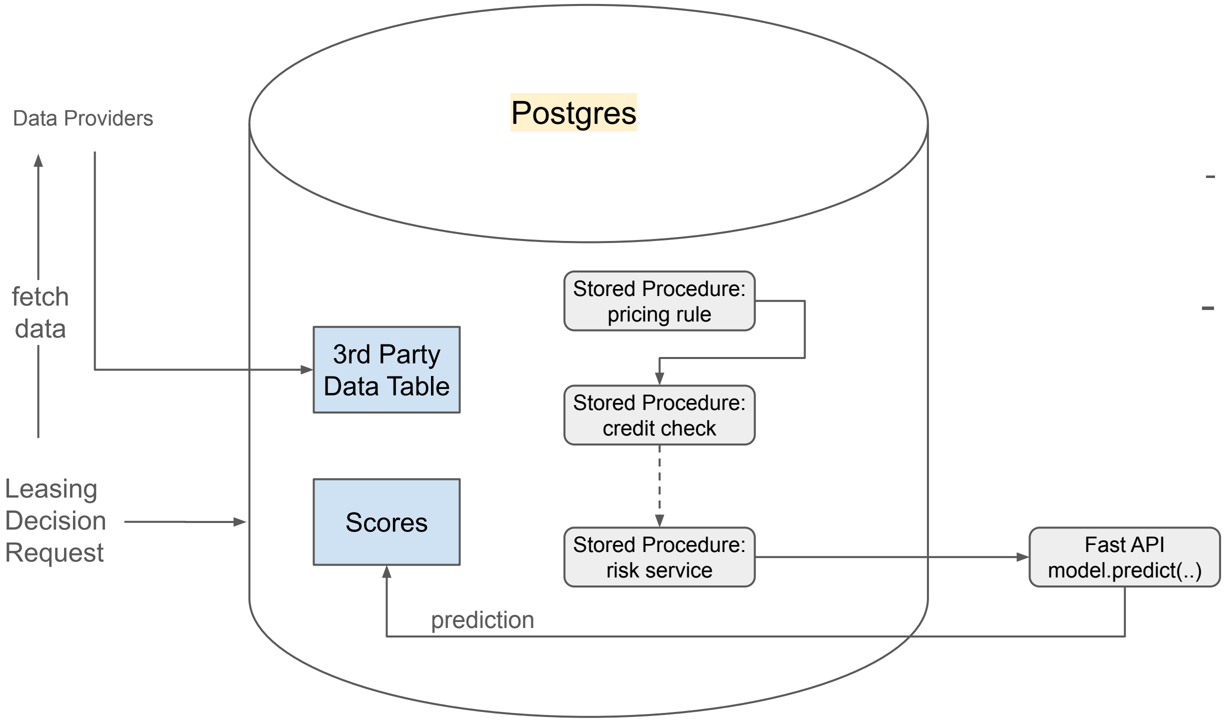
Mission:

To fulfill the capital needs of the underserved entrepreneurs and their main street businesses with simple, fast, and innovative equipment leasing solutions.

Online Decisioning

- Predict the probability of default
- Fraud
- Pricing
- Regulations and Rules
- High Cost for Mistakes

First Version of ML Decisioning Architecture


Postgres
Data Providers
fetch data
Leasing Decision Request
3rd Party Data Table
Scores
Stored Procedure: pricing rule
Stored Procedure: credit check
Stored Procedure: risk service
Fast API model.predict(..)
prediction

- Stored Procedures for everything
- No modularity

Online Decisioning Challenges

Performance

- High number of On-Demand Features (only available at request time)
- Latency

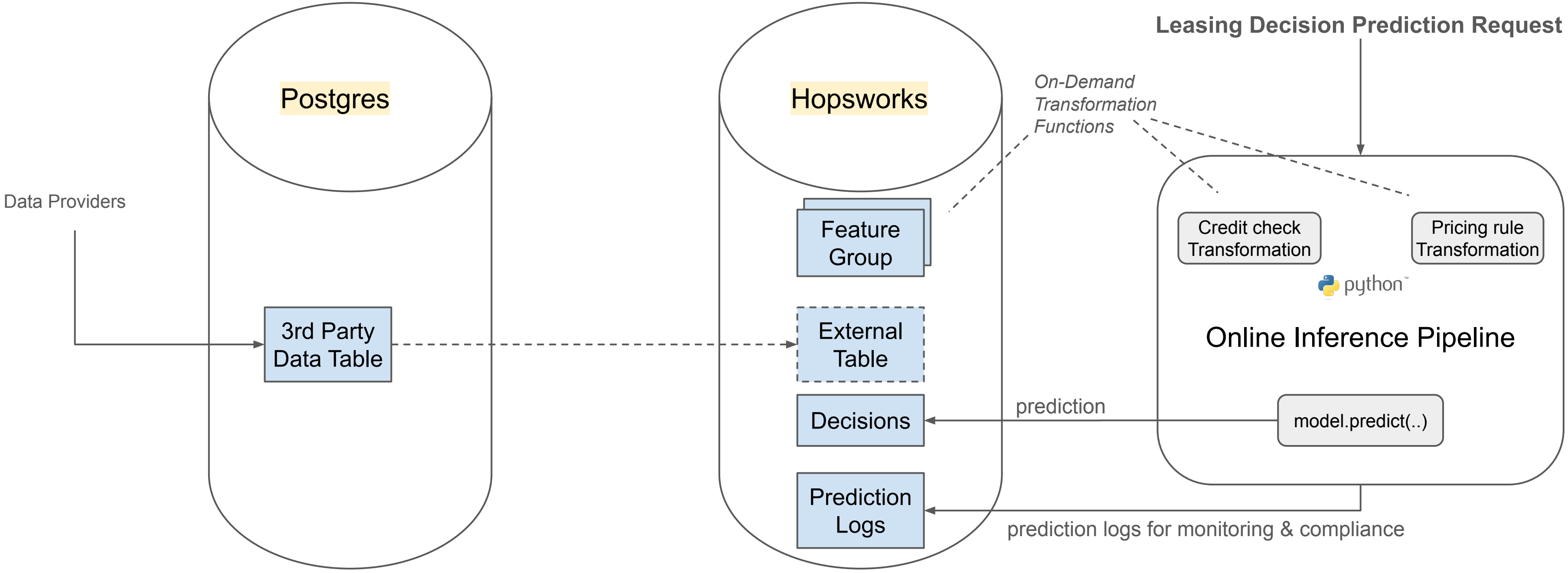
Development

- Models coupled to software deployments
- Updating models was not possible

Management

- Poor validation experience
- Schema issues

Architecture with Hopsworks Feature Store


Leasing Decision Prediction Request
Postgres
Hopsworks
On-Demand Transformation Functions
Data Providers
3rd Party Data Table
Feature Group
External Table
Decisions
Prediction Logs
Credit check Transformation
Pricing rule Transformation
python
Online Inference Pipeline
prediction
model.predict(..)
prediction logs for monitoring & compliance

Results of Using Hopsworks

- Python centric feature management (Minimal runtime Errors)
- Unified schema for online and offline
- Low latency server calls
- Built-in Prediction Logs for Monitoring and compliance
- Reusability of code and features
- Testable feature transformations

Additional Benefits

Little things that make my life easier

First Online Serving Pattern

- Tightly coupled to microservice
- Dependencies are not isolated
- Feature logic is not reusable

One Model One Endpoint

- Tightly coupled to microservice
- Feature logic is not reusable
- No. way isolate model updates

Online Inference Management

- Control over deployments, changes, and model versions
- Decoupling of the model and artifacts
- Fast updates in production

Thank You

FEATURE STORE
SUMMIT
2025

Organized by HOPSWORKS