intuit simplify the business of life

turbotax quickbooks proconnect mint

Machine Learning
Models as a Service

May 20, 2019

Tobias Wenzel
Vigith Maurice

intuit powering prosperity

turbotax quickbooks proconnect mint

Consumers	Small Businesses	Self-Employed







Data Science Operations is not easy

"We spend more time bringing the model to production than developing and training it"

Data Science Operations is not easy

"We spend more time **bringing the model to production** than developing and training it"

Let's build a platform

Centralized or Decentralized?

Let's build a platform

Centralized!

Let's build a centralized platform

The platform controls the environment in which models are deployed

Let's build a centralized platform

The platform can implement actions of the machine learning model life cycle centrally

Let's build a centralized platform

Create self service APIs for every interaction with the platform

Security for an ML Platform



https://news.ycombinator.com/item?id=15256121

Security for an ML Platform



Package	Typo-squatting Info	Exploit Type
smplejson	Standard Library	Research / Data leakage
pkgutil	Standard Library	Research / Data leakage
timeit	Standard Library	Research / Data leakage
diango	Django package	Research / Data leakage
djago	Django package	Research / Data leakage
dajngo	Django package	Research / Data leakage
djanga	Django package	Gain persistence through modifying the .bashrc script.
easyinstall	easyinstall package	Gain persistence through modifying the .bashrc script.
libpeshka	None	Gain persistence through modifying the .bashrc script.
pyconau-funtimes	None	Reverse shell
mybiubiubiu	None	Data leakage

intuit

https://medium.com/@bertusk/detecting-cyber-attacks-in-the-python-package-index-pypi-61ab2b585c67

Let's build a centralized platform

Runtime security can be taken care of by the platform

Let's build a centralized platform

Updates can be rolled out to every running model by updating the platform

Intuit's ML Platform

Machine Learning Platform

Model **Training** → Model **Execution**

User Interfaces (Web, API, CLI)

Continuous Training (Argo)

Self Service (Model Lifecycle Management)

Monitoring (Wavefront)

Data Aggregation (Pre-fetch, Cache, Optimize)

Prediction Quality Feedback (Beacons)

Secure Data Access (Intuit Data Lake)

Access Control (Authn, Authz)

Logging Tracing (Splunk)

Auto Scaling (Cost Control)

Billing (Chargeback)

Notebooks (via Sagemaker) (Data Exploration, Visualization)

Storage (Training Data, Model Artifacts)

ML Compute (via Sagemaker) (GPUs etc.)

intuit

And Now for Something Completely Different



Let's build a centralized platform

Observability

Beaconing / Monitoring / Logging

Let's build a centralized platform

Introspection

Inspect prediction response (Monitoring Contd.)

Introspection

Meaning of life = 43

I expected 42!

Beaconing

Re-training and Monitoring

What is a beacon?

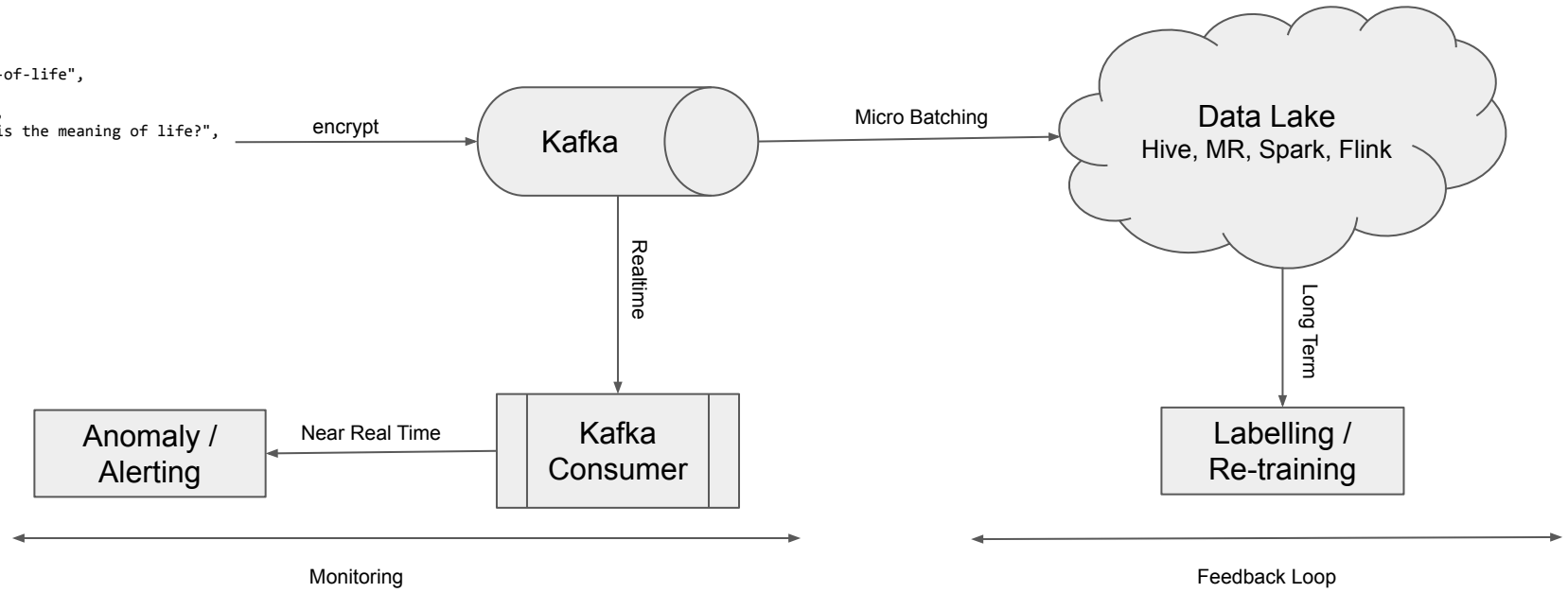
```
{
    "modelName": "meaning-of-life",
    "modelVersion": "1",
    "environment": "PROD",
    "requestBody": "What is the meaning of life?",
    "responseBody": "42"


    … Metadata …
}
```

Beaconing - BYOC

Realtime Beacons for Predict Request and Responses

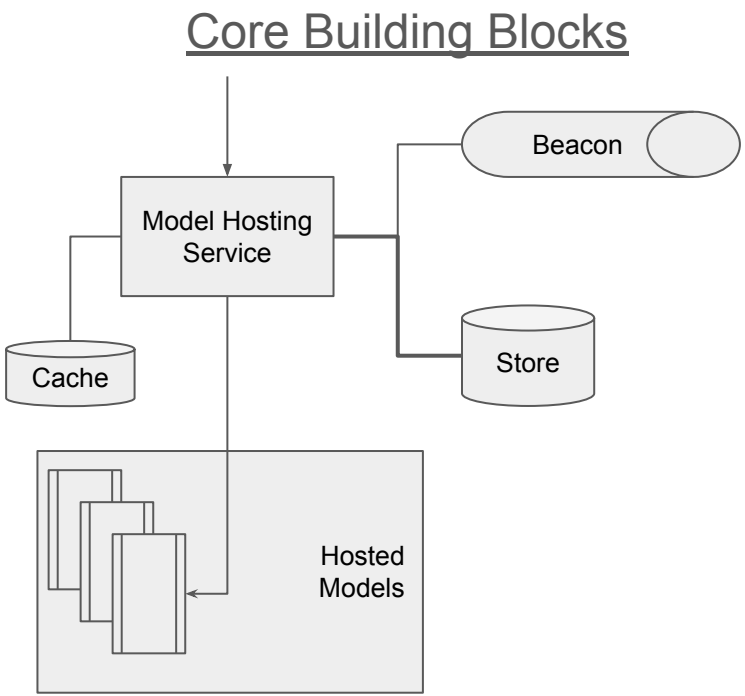
```
{
"modelName": "meaning-of-life",
"modelVersion": "1",
"environment": "PROD",
"requestBody": "What is the meaning of life?",
"responseBody": "42"
}.encode()
```



Monitoring

the bottom layer of the **Hierarchy of Production Needs**, is fundamental to running a stable service[1].

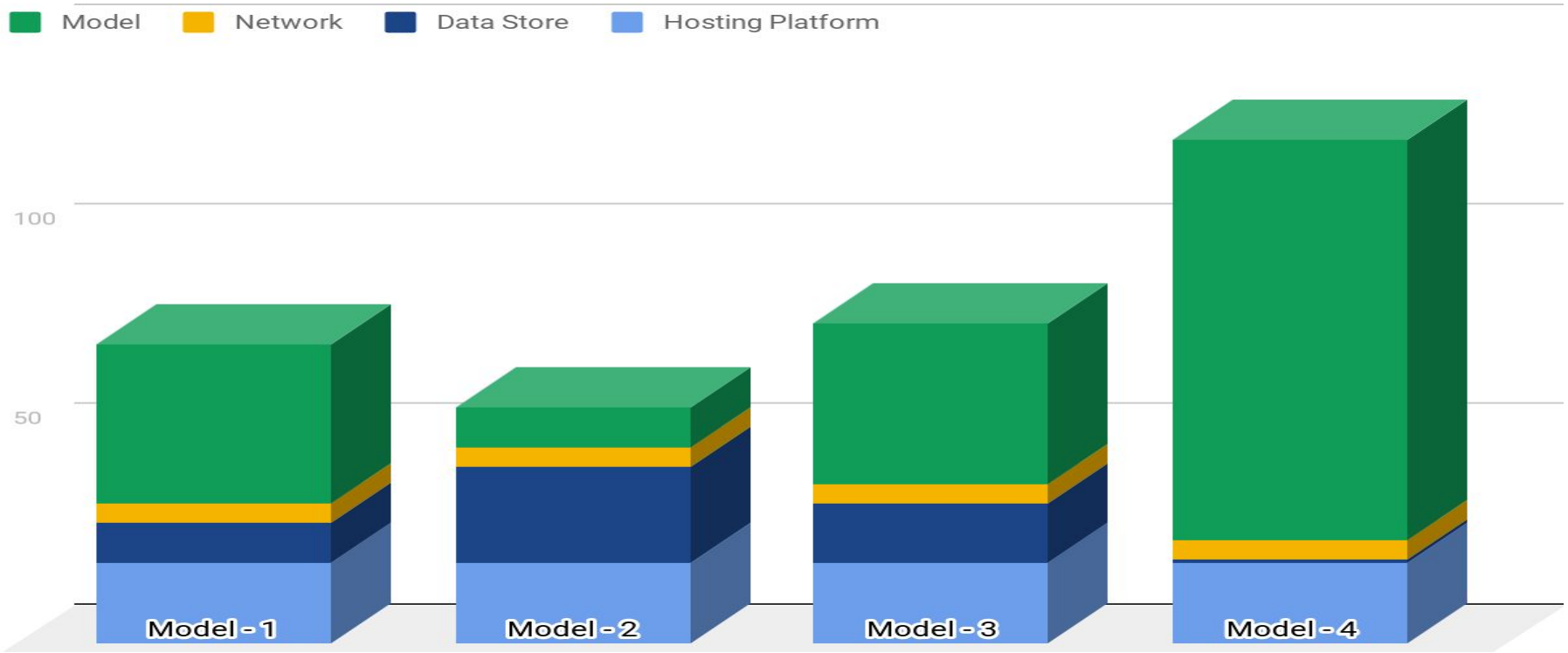
Monitoring - Platform on Platform



Counters	Timers	Distribution
Requests	Core Process Time	Request Size
HTTP Status	Total Response Time	Response Size
Store Requests	Store Process Time	Store get/put size
Cache Hit/Miss	Cache Process Time	Cache get/put size
Beacons	Beacon Emit Time	Beacon size
Authentication	LMA Process Time	

- Cardinality of each is increased by tagging
- Availability is formulated from granular metrics
- Most alerting is based on P99

Monitoring - Request Latency



intuit

Let's build a centralized platform

Logging

Centralized Logging

- Accessibility (easy to access)
- Compartmentalized (logs are grouped by Model)
- Traceability (transactional id flows from end user app all the way to Model)
- Near real time
- Log Retention

Machine Learning Models as a Service

Running ML models is just like running a Service

1. Make actions in ML lifecycle **self service**
2. Take the **operations burden off** of the data scientist
3. Make sure models are **run securely**
4. Provide **common functionalities** to all models at scale
5. Provide **logging, tracing and monitoring** out of the box