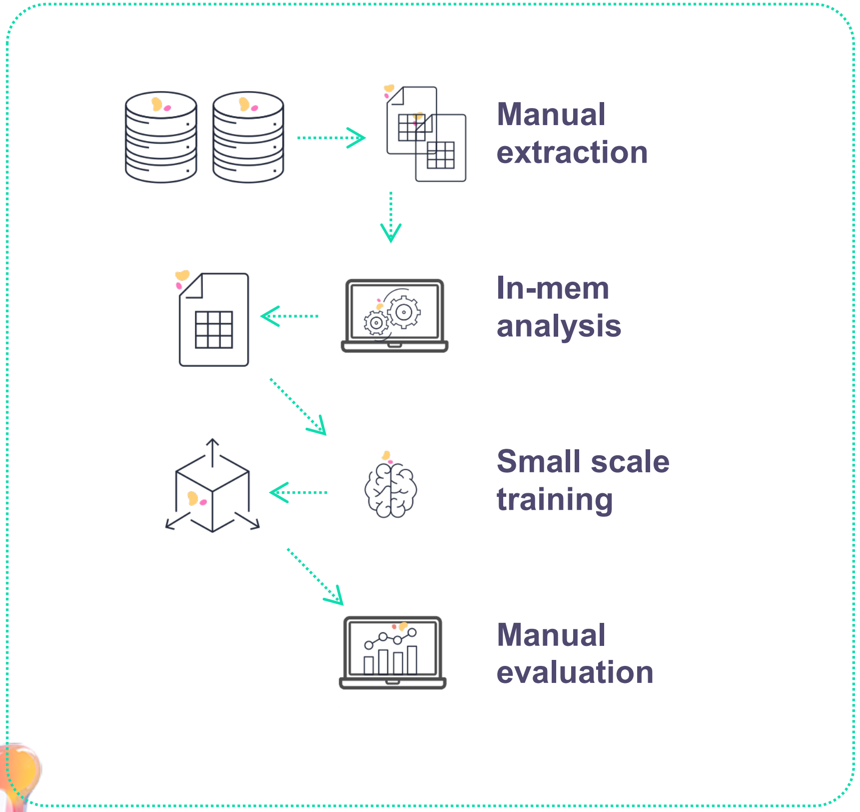
iguazio

MLOps Automation with Git Based CI/CD for ML

Yaron Haviv
CTO, Iguazio
@yaronhaviv

80% of AI Projects Never Make it to Production

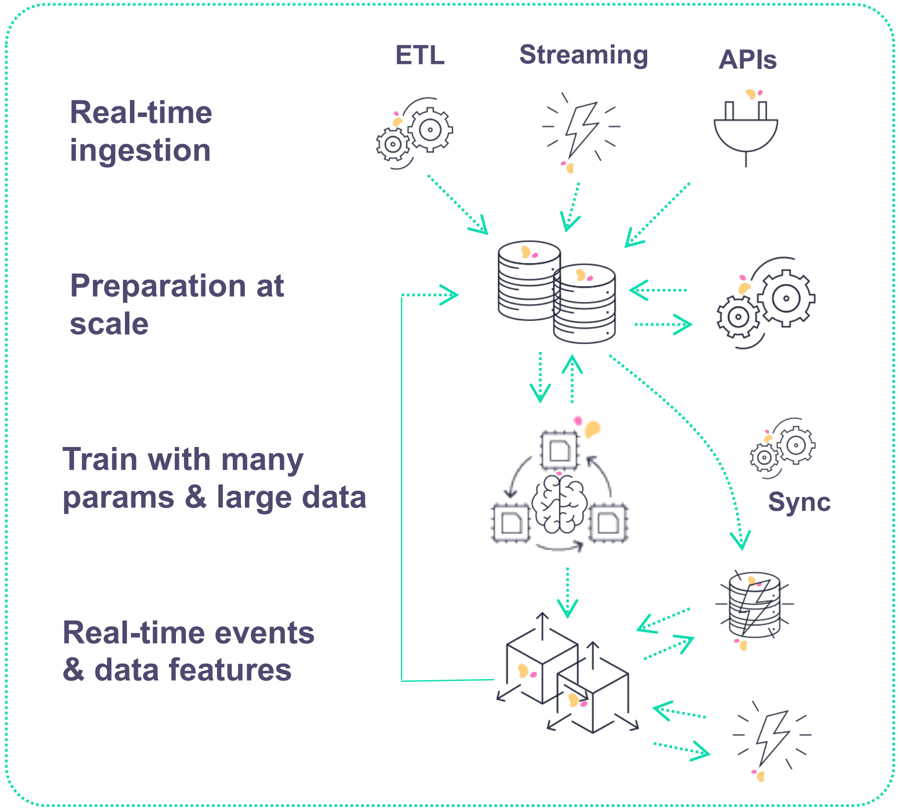
Research Environment



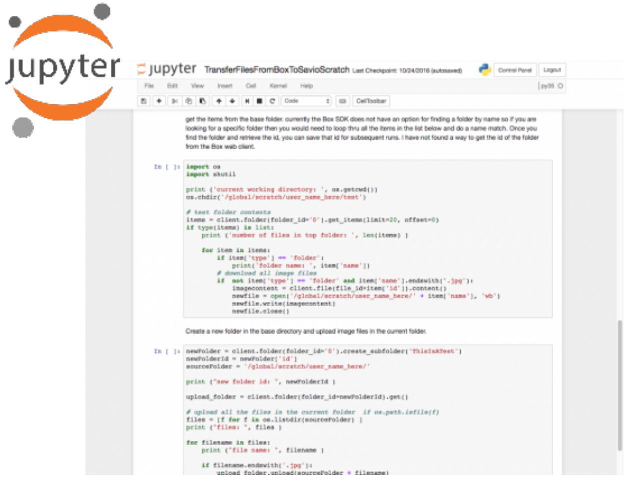
Build from Scratch

with a Large Team

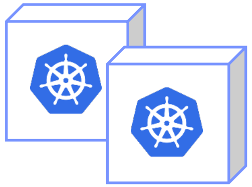
Production Pipeline



Did you Try Running Notebooks in Production?



Refactor and operationalize



Micro-services

Model and Code Development are Just the First Step

Production

Develop and Test Locally

Package
—
- Dependencies
- Parameters
- Run scripts
- Build

Scale-out
—
- Load-balance
- Data partitions
- Model distribution
- AutoML

Tune
—
- Parallelism
- GPU support
- Query tuning
- Caching

Instrument
—
- Monitoring
- Logging
- Versioning
- Security

Automate
—
- CI/CD
- Workflows
- Rolling upgrades
- A/B testing

Weeks

with **one** data scientist or developer

Months

with a **large team** of developers, scientists, data engineers and DevOps

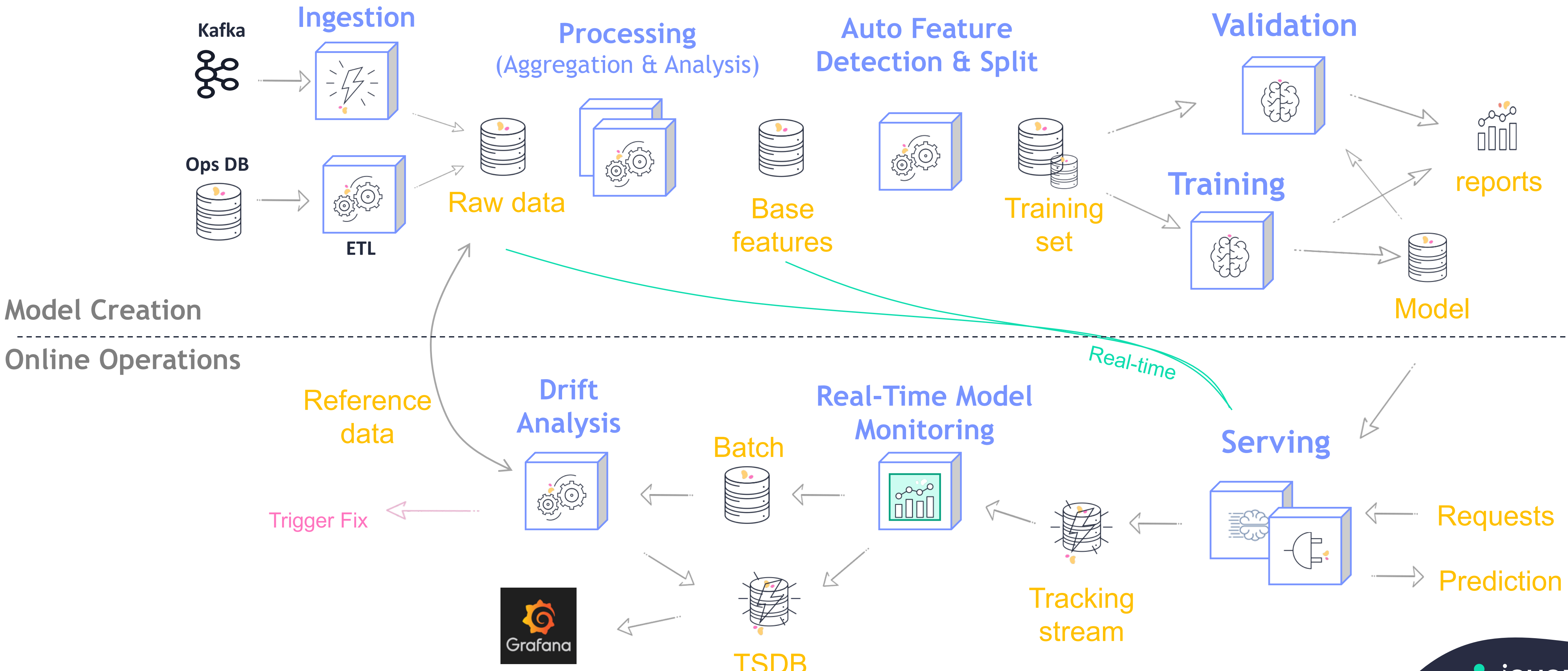
~~DevOps~~ MLOps

And data-science

"Combine Dev and Ops to shorten the systems-development life cycle while delivering features, fixes, and updates frequently in close **alignment with business objectives.**"

Example: Predictive Maintenance Pipeline

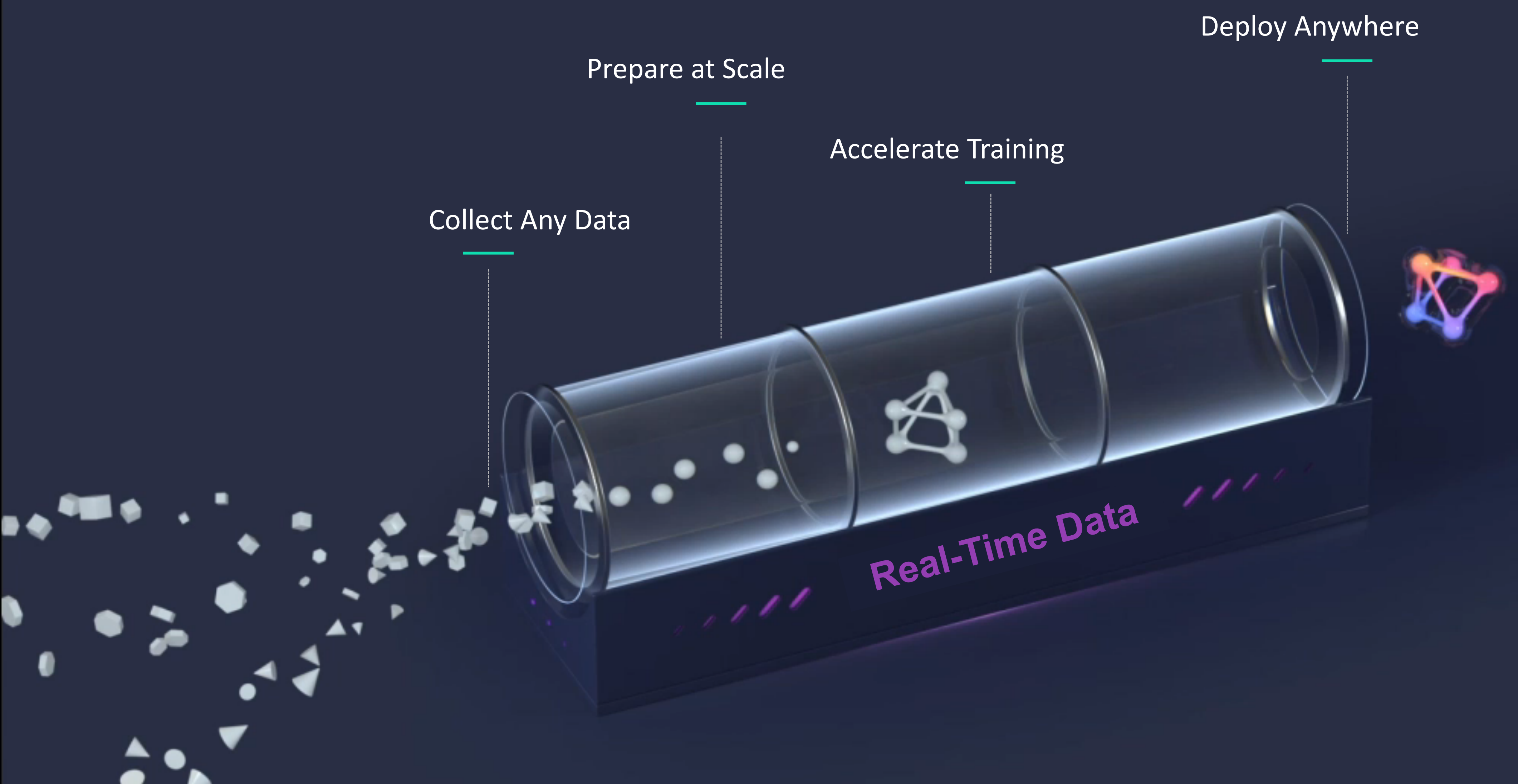
https://github.com/mlrun/demo-network-operations



iguazio

You can use Separate Tools & Services, Or you can Use Kubernetes as the Baseline

Bring your AI Applications to Life with Data Science Automation and MLOps

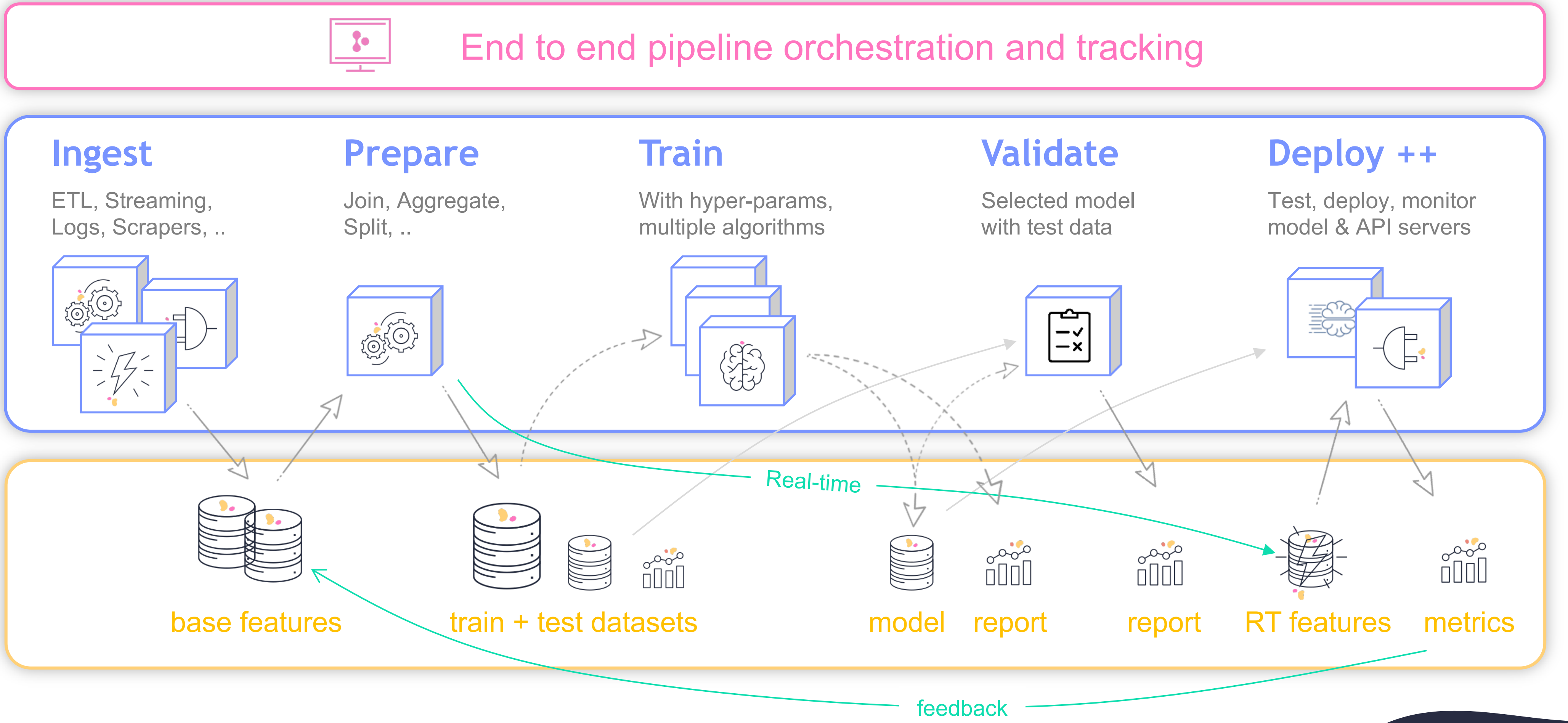


iguazio

What is an Automated ML Pipeline ?

Serverless:
ML & Analytics
Functions

Features/Data:
Fast, Secure,
Versioned



Under The Hood:
Open, Scalable, Production Ready

MLRun

Pipeline Orchestration

Auto ML | Experiment Tracking | Feature Store | Workflows (Kubeflow)

Kubeflow

Serverless Automation

Managed Functions and Services

DASK · jupyter · HOROVOD · scikit learn · PyTorch · presto · TensorFlow · nuclio · Apache Spark

kubernetes | Shared GPU/CPU Resources

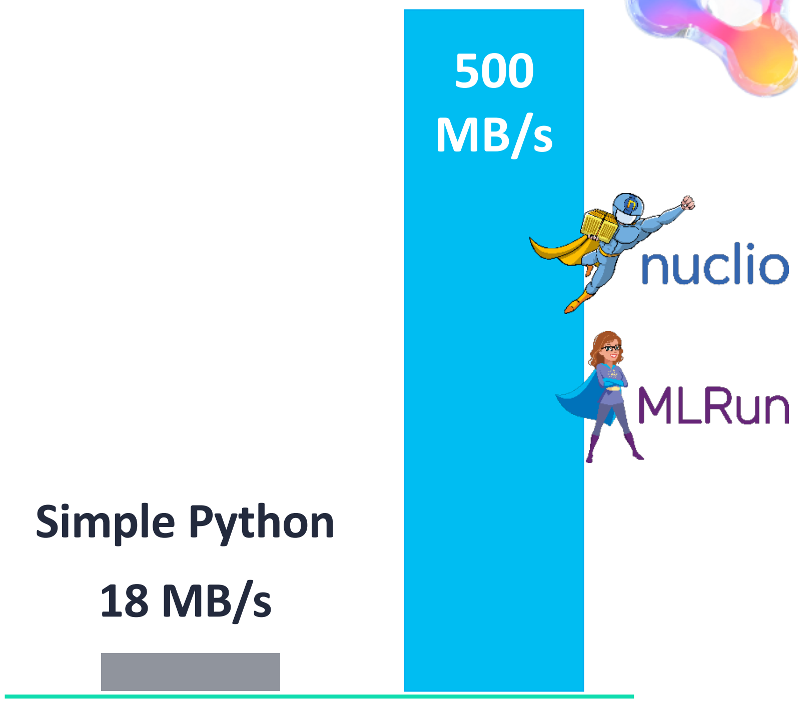
Data layer

Real-Time Data Layer

External Data Sources (file, object, DBs)

Serverless Simplicity, Maximum Performance

- **Automated** code to production
- **Elastic** resource scaling (zero to N)
- **Effortless** logging, monitoring, and versioning
- **High-performance** runtimes + fast data access
- **Glue-less** pipeline and tracking integration
- **Reusable** internal/public function marketplace



”Moving from Hadoop/Java to Serverless reduce 90% of our code footprint and got us much better performance”

Serverless: Resource Elasticity, Automated Deployment and Operations

So why not use Serverless for training and data prep?

	Serverless Today	Data Prep and Training
Task lifespan	Millisecs to mins	Secs to hours
Scaling	Load-balancer	Partition, shuffle, reduce, Hyper-params
State	Stateless	Stateful
Input	Event	Params, Datasets

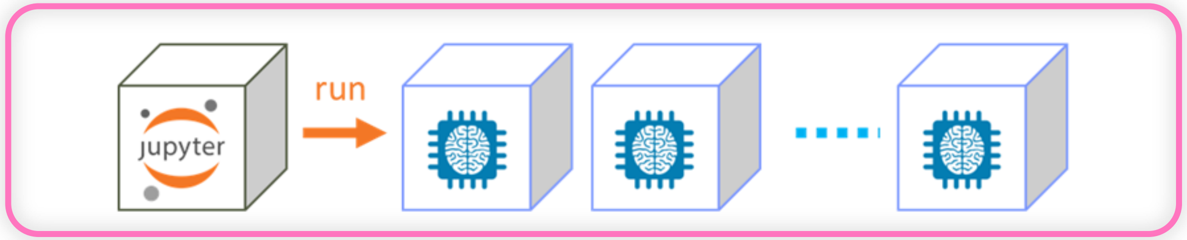
It's time we extend Serverless to data-science!

Dynamic Scaling for Intensive Workloads

- Scale + Performance for intensive ML & Data processing tasks
- Seamless transition from user code to elastic, auto tracked jobs + data
- AutoML & Hyper-params are built-in
- Make frameworks “Serverless”
  - Spark, Dask
  - MPI/Horovod
  - SQL (via Presto)
  - Nuclio

MLRun

Dynamically Scaled Containers + Distributed Tracking



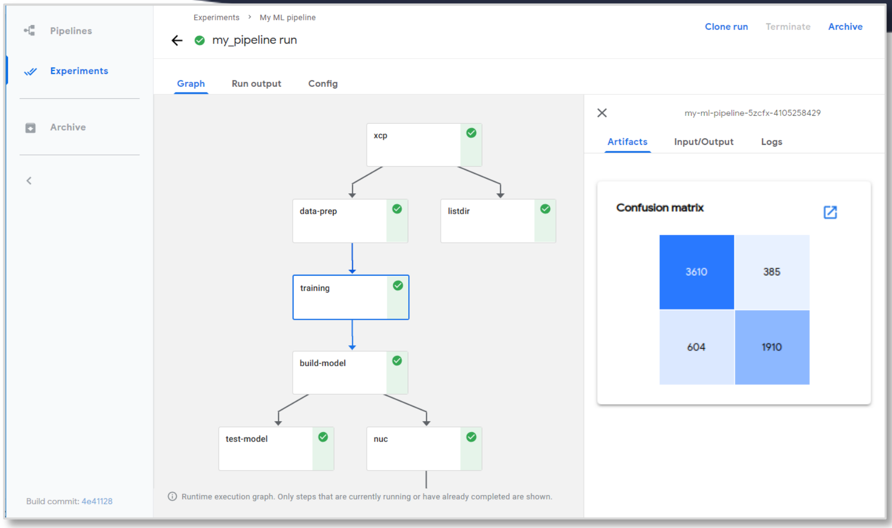
Fast inter cluster messaging (MPI, Dask, Spark, ..)

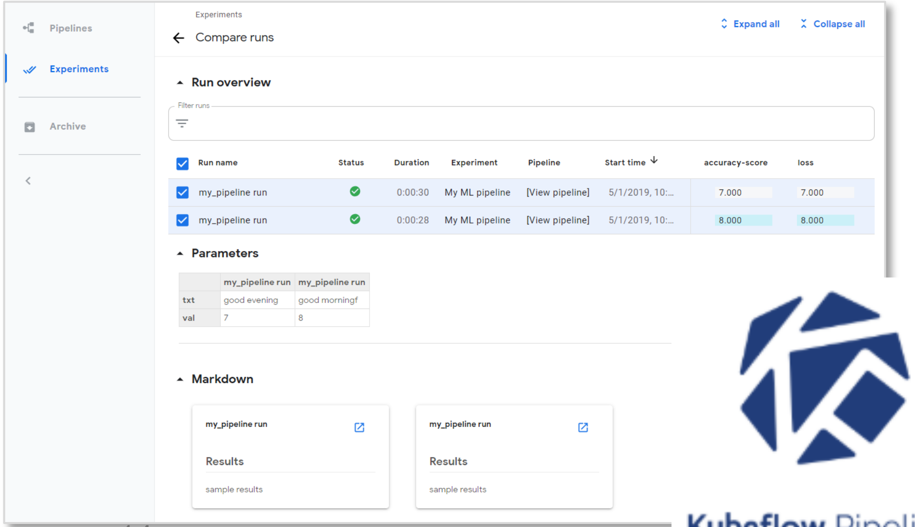
Low-latency data layer (shared code, files, dataframes)

https://github.com/mlrun/mlrun

iguazio

KubeFlow: Automated ML Pipelines & Tracking





Kubeflow Pipelines

Integrating and Extending KubeFlow Pipelines

Manage experiments, runs, and artifacts

Build workflows using code or reusable components (across many cloud/3rd party ML and data framework)

With MLRun & Nuclio:

Automated code to serverless function

Glueless data access and parallelism

Distributed training and GPU Acceleration

Code + execution + data tracking and versioning

Simple, Production-Ready Development Process

Data-Scientist / Developer

1. Write and test functions locally

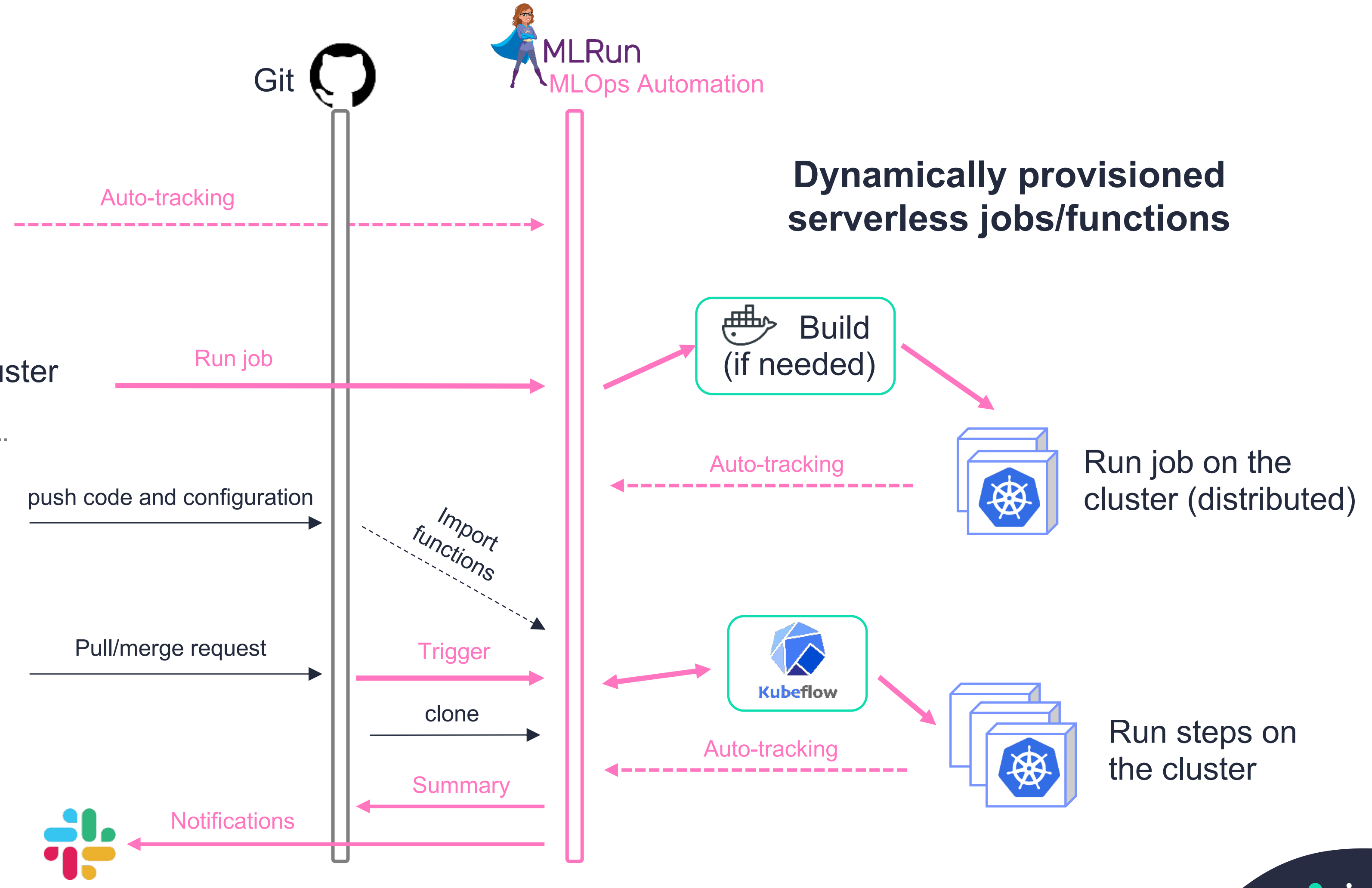
2. Add requirements, run on the cluster

Specify image, packages, cpu/gpu/mem, data, ..
Requirements via annotation or function spec

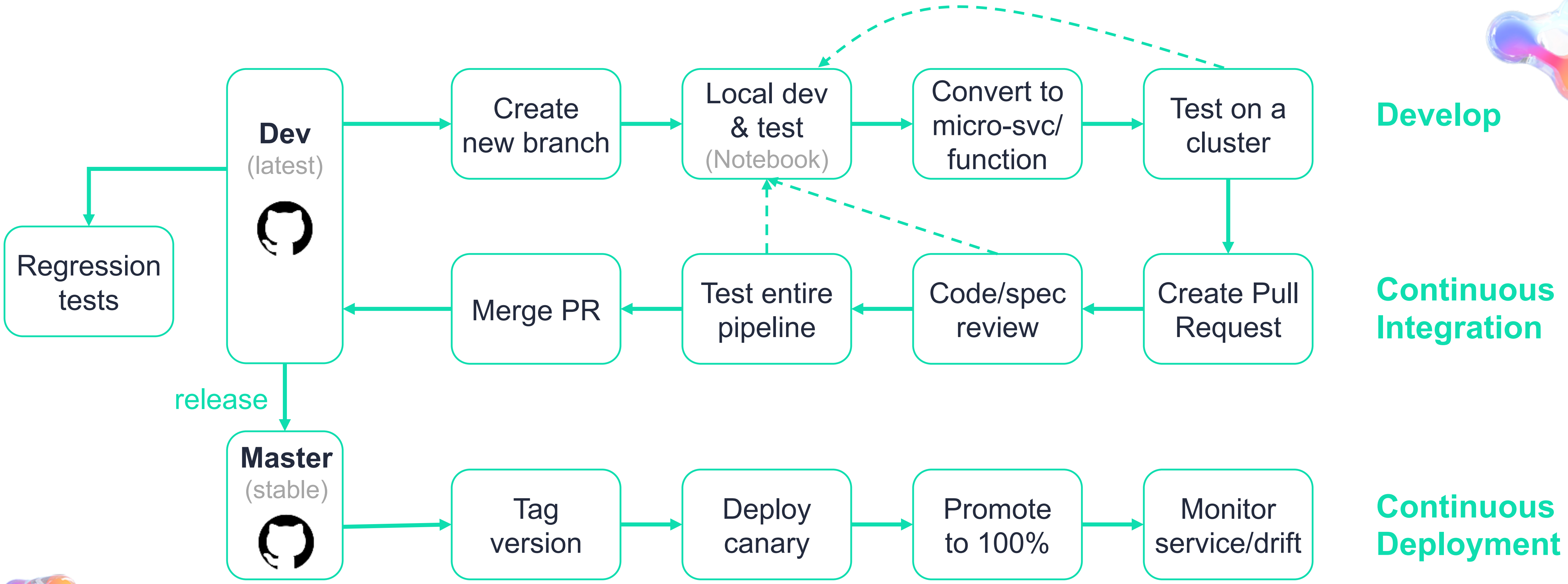
```
%nuclio cmd -c pip install pandas
%nuclio config spec.build.baseImage = "mlrun/mlrun"
```

ML Engineer

3. Build/run ML pipeline
   (interactive or via triggers)

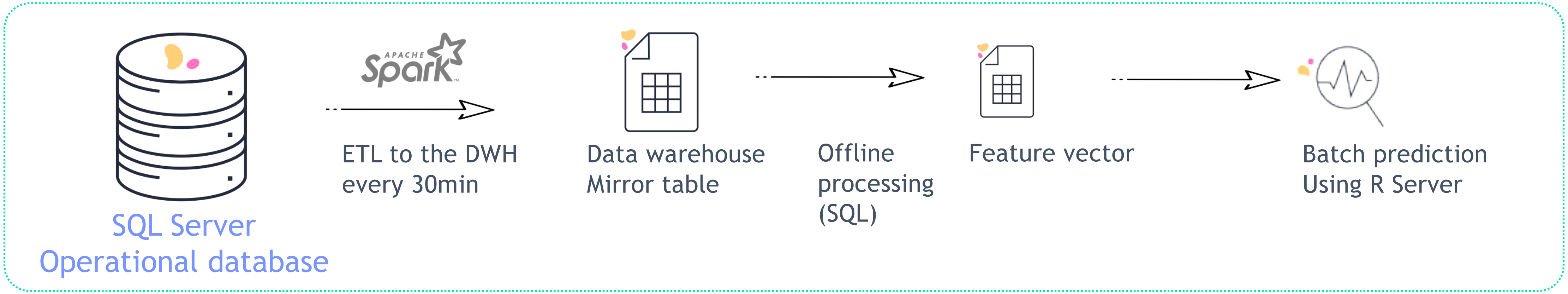


Building CI/CD Process for ML(Ops)



Traditional Fraud-Detection Architecture (Hadoop)

40 **Precious** Minutes (detect fraud after the fact)

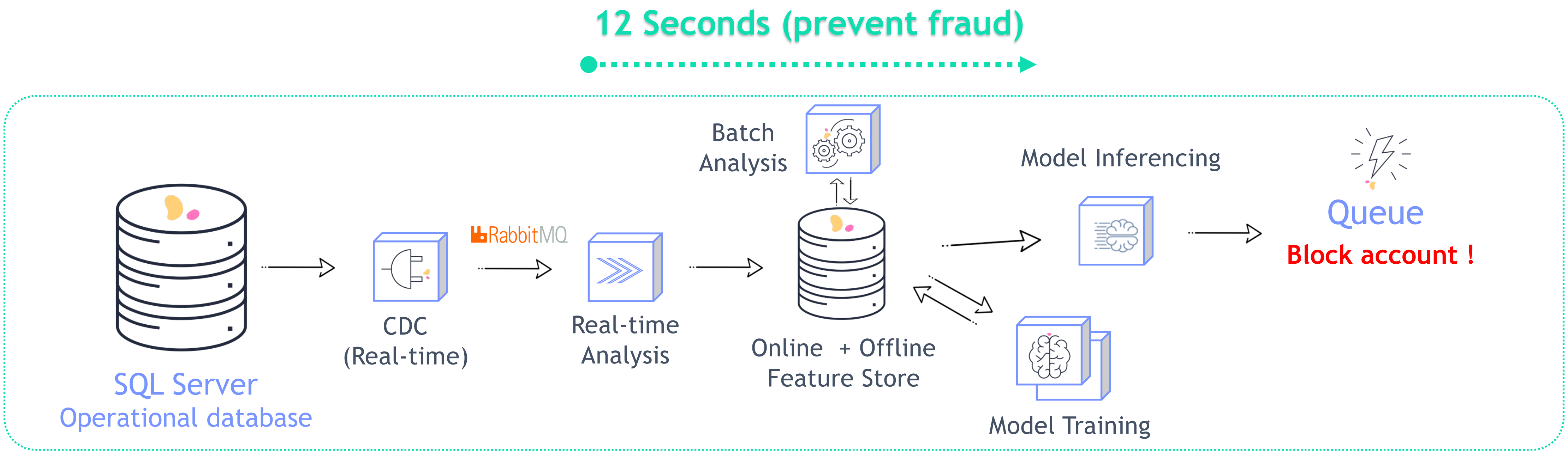


40 minutes to identify suspicious money laundering account

Long and complex process to production

iguazio

Real-Time Fraud Prediction & Prevention



12 Seconds to detect and prevent fraud !

Automated dev to production using a serverless approach

DEMO

Start creating real-world business impact with AI, today

www.iguazio.com