TU1BCO01

# A workflow for training and deploying Machine learning models in EPICS

Mateusz Leputa

ICALEPCS 2023 – Cape Town, South Africa

10th of October 2023

Science and Technology Facilities Council
ISIS Neutron and Muon Source

# Overview

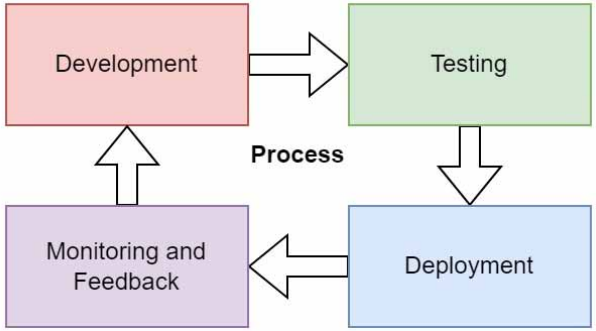
- Motivation
- Workflow development to deployment
  - Development workspaces
  - Model and data archiving
  - Deployment and serving
- Examples
- Workflow summary
- Future development

# Motivation

Developing machine learning systems for accelerator controls.

However, we have to deal with:

- Shared responsibilities.
- Frequent shelving and “reheating” of projects.
- Turnover



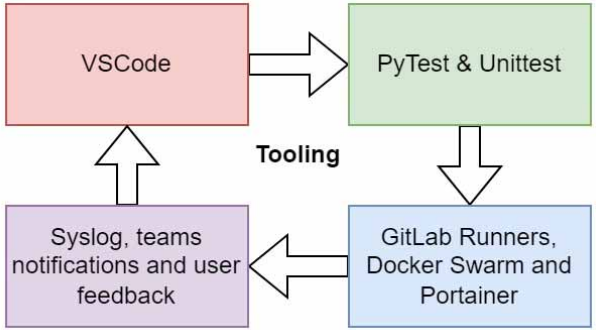

**Adapt existing DevOps to machine learning operations, MLOps.**

Thing we want to  adopt:

- CI/CD – Continuous Integration and Continuous Deployment
- Modular Architecture with majority “off-the-shelf” components.
- Version control systems for models and data

Prevent:

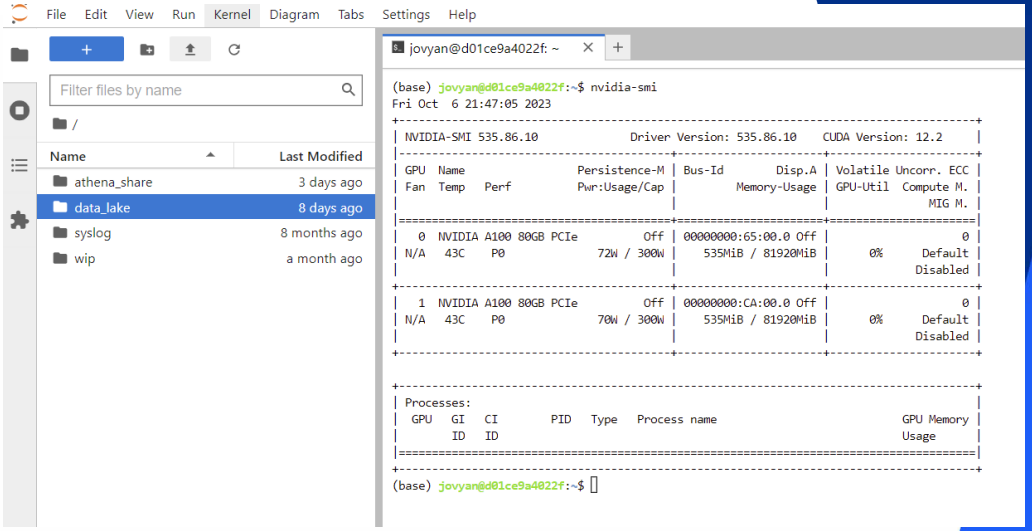
- Knowledge siloing
- Recreating the same code in every project



# Remote Workspaces/Development Environment

**Stack: JupyterLab and Hub**

- Developers are already familiar with JupyterLab
- NFS – facilitates data transfers and collaboration spaces
- High spec servers – GPUs, high spec CPUs, RAM etc.
- 24/7 uptime – no need to leave PC on or wait for jobs to finish.
- Centralised!



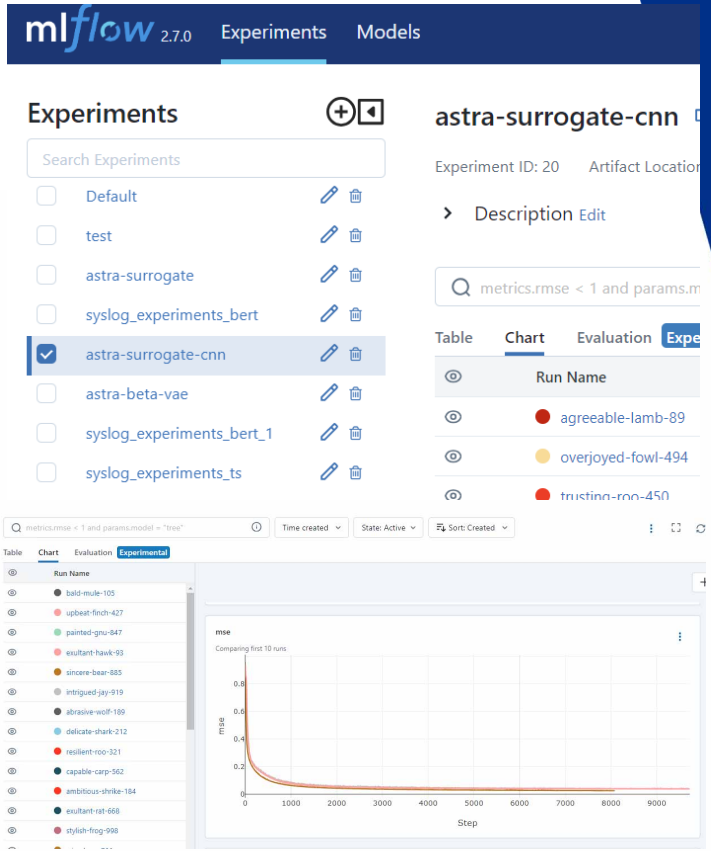


https://jupyter.org/hub

# Model and Data Archiving

**Stack: MLflow, MINIO, PosgreSQL**

- Comes with a web GUI.
- Saves experiment setup, performance metrics, datasets and model artifacts.
- Provides an API to programmatically upload and download models, query experiment results and charts etc.
- Comes with its own model serving utilities.
- Can tag models e.g. experimental or production!





Science and Technology Facilities Council
ISIS Neutron and Muon Source

www.isis.stfc.ac.uk
@isisneutronmuon
uk.linkedin.com/showcase/isis-neutron-and-muon-source

# Model Serving and Deployment

**Stack: TF-Serve, TorchServe & p4p python library**

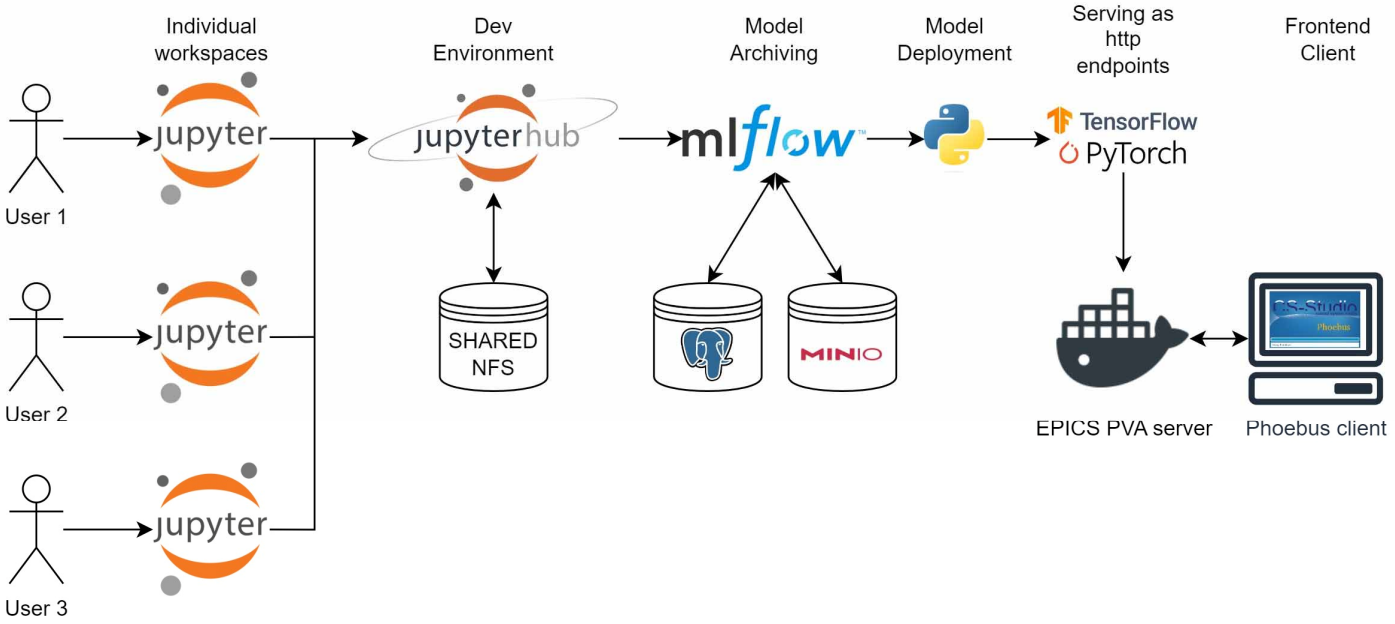
- Model deployment is facilitated though serving frameworks such as TF-Serve or TorchServe. Deployed as containers
- Both serve models as HTTPS endpoints and can run inference remotely on GPUs
- Latency of 16-40 ms for small models (mostly attributed to network latency)
- HTTPS to EPICS PVA server deployed as a service.

Frontend application can then connect to the EPICS PVs and interact with the model …

# Example 1 – ASTRA Surrogate – ISIS MEBT

# Example 2 – LEBT

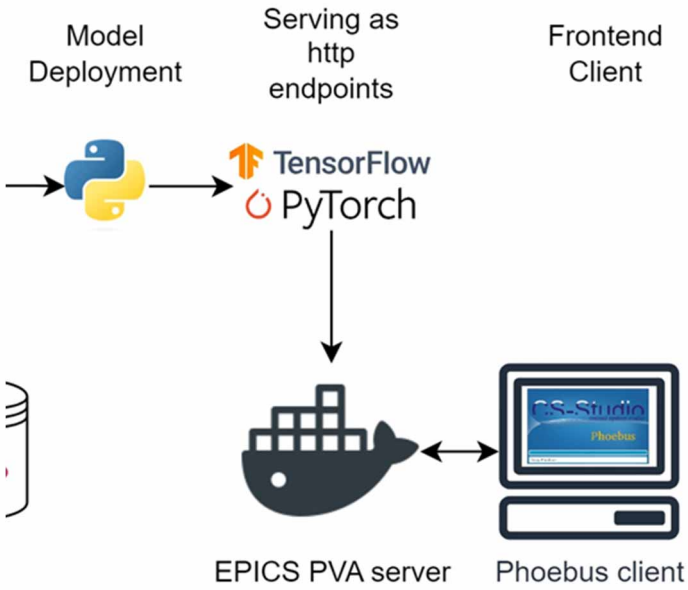
# Workflow Architecture - Summary



# Further Developments

- Standardise model HTTPS to EPICS serving.
- Model monitoring and retraining for continual learning.
- Model profiling.

Mostly concerning the later parts of the MLOps workflow



# Questions?

mateusz.leputa@stfc.ac.uk

Science and Technology Facilities Council
ISIS Neutron and Muon Source

www.isis.stfc.ac.uk
@isisneutronmuon
uk.linkedin.com/showcase/isis-neutron-and-muon-source