University of South Australia

COMP 2019

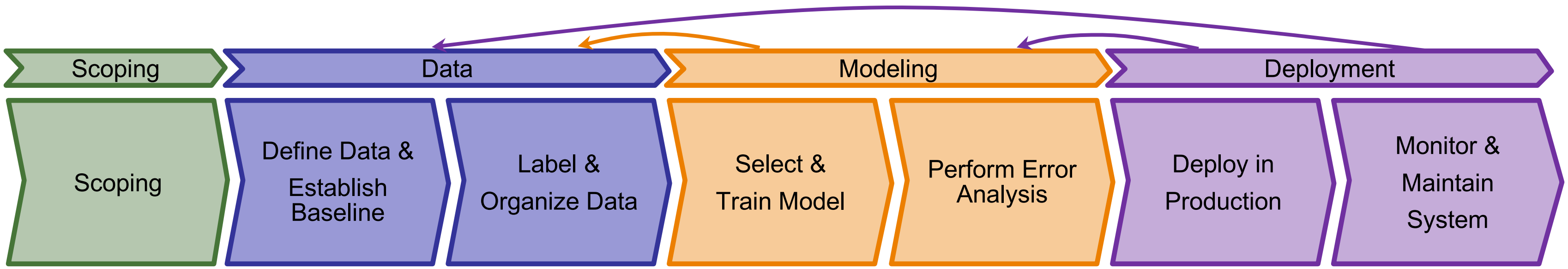
Week 11

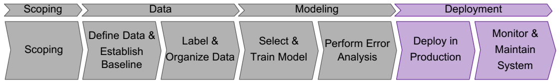
Deployment of ML Systems

Learning Objectives

- Explain the ML project lifecycle (CO3)
- Discuss key challenges and activities in the ML project lifecycle (CO3)
- Explain Deployment Options and MLOps (CO4)

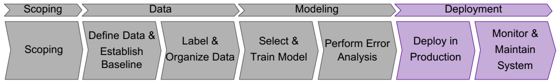
ML Project Lifecycle





Key Challenges for Deployment

- Concept Drift
  - Has the data changed?
  - What to monitor?
- Software Engineering issues
  - Realtime vs batch
  - Cloud vs Edge/Browser
  - Compute resources
  - Latency, throughput
  - Security & privacy
  - Logging

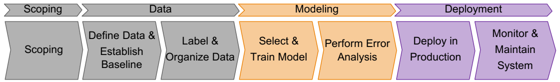


Deployment Patterns

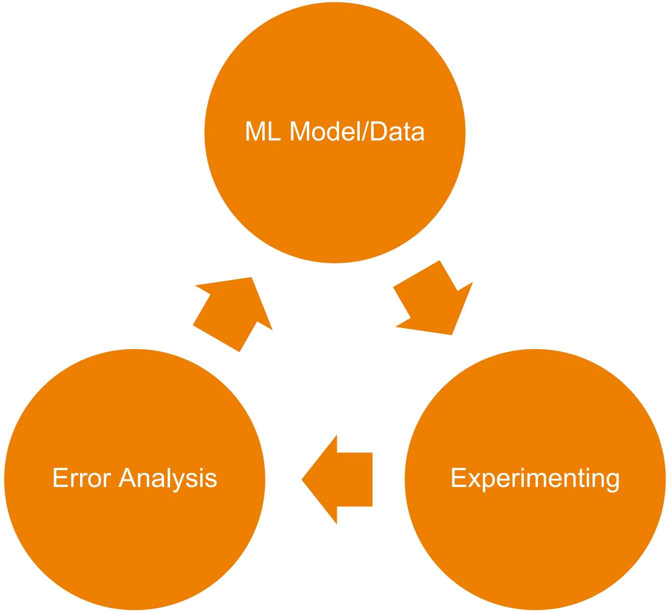
- Shadow mode
  - AI runs in parallel to the manual task
  - AI not involved in decision-making
- Canary mode
  - Deploy to a small fraction (5%) of requests
  - Monitor and ramp up gradually
- Old/New routing
  - Setup new prediction service alongside the old service
  - Router component switches to new service (gradual?)
  - Easy rollback

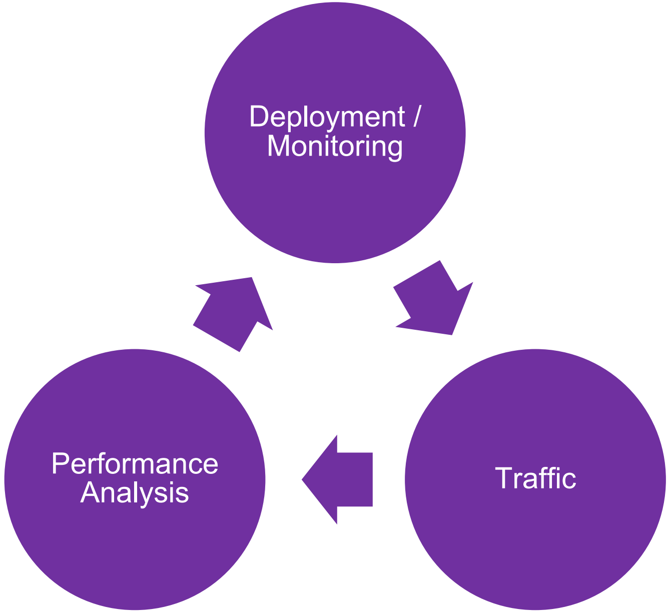
Degrees of Automation

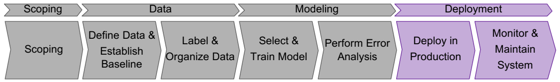
Human Only → Shadow Mode → AI Assistance → Partial Automation → Full Automation


Scoping
Data
Modeling
Deployment
Scoping
Define Data & Establish Baseline
Label & Organize Data
Select & Train Model
Perform Error Analysis
Deploy in Production
Monitor & Maintain System

Monitoring


ML Model/Data
Experimenting
Error Analysis

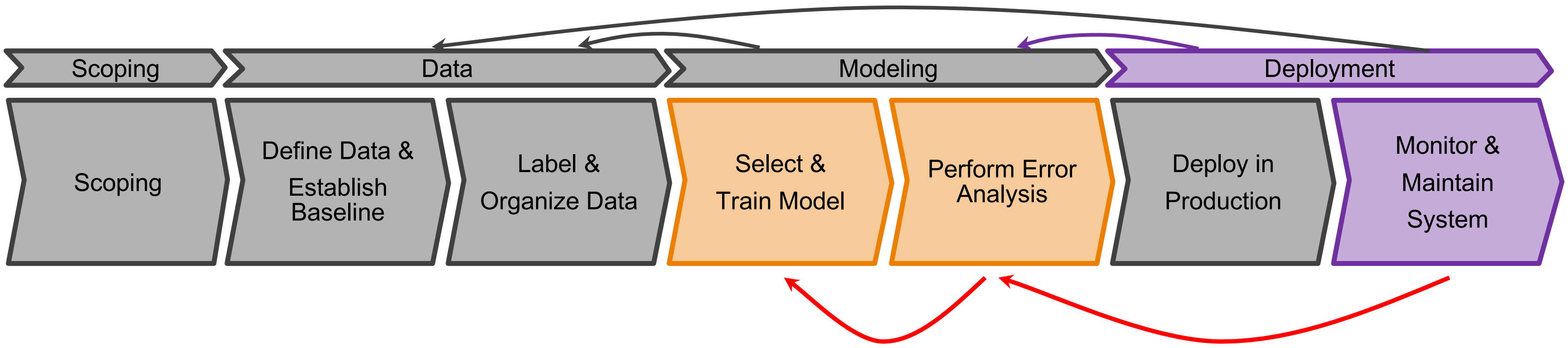

Deployment / Monitoring
Traffic
Performance Analysis

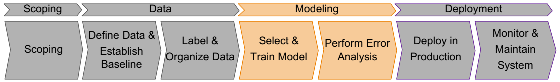


Metrics

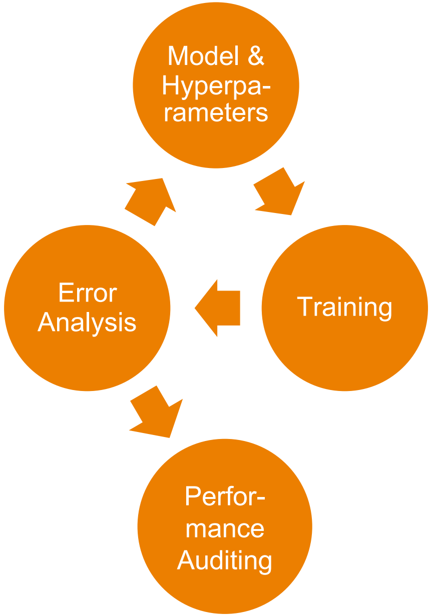
- Software metrics
  - Memory, compute, latency, throughput, server load
- Input metrics
  - Average length, average volume, missing values
- Output metrics
  - Frequency of null answers, frequency of users switching/redoing query, … (application specific)

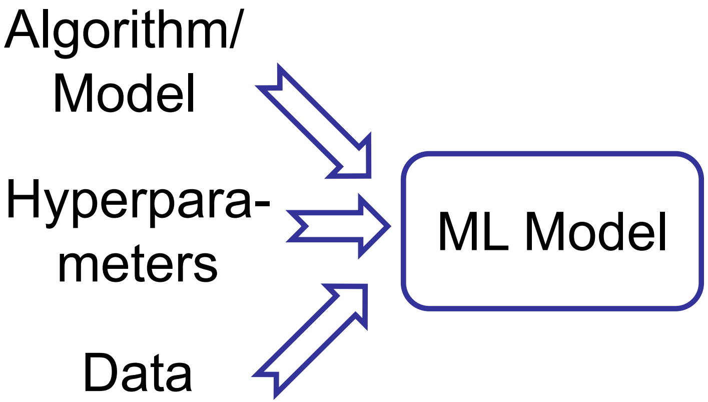
Model Maintenance

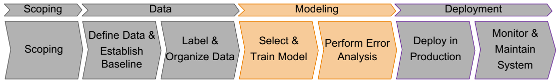




Iterative Model Development







Doing Well?

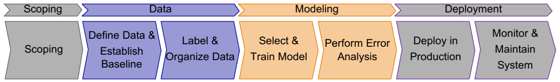
- On the training set
- On the dev/test set
- On the business metrics & goals

Word level accuracy | Query level accuracy | Search result quality | User engagement | Revenue

ML Metrics ⟷ Business Metrics

Error Analysis & Prioritization

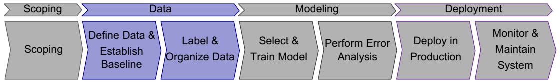
Type	Accuracy	Human Level Performance	Gap to HLP	% of data	Potential improvement
Clean speech	94%	95%	1%	60%	0.60%
Car noise	89%	93%	4%	4%	0.16%
People noise	87%	89%	2%	30%	0.60%
Low bandwidth	70%	70%	0%	6%	0.00%



Data Centric Development

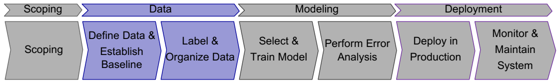
- Model Centric
  - Hold the data fixed and iteratively improve the model

- Data Centric
  - Hold the code fixed and improve the data
  - Good data will allow multiple models to do well
  - Collect more data, data augmentation, data/label quality

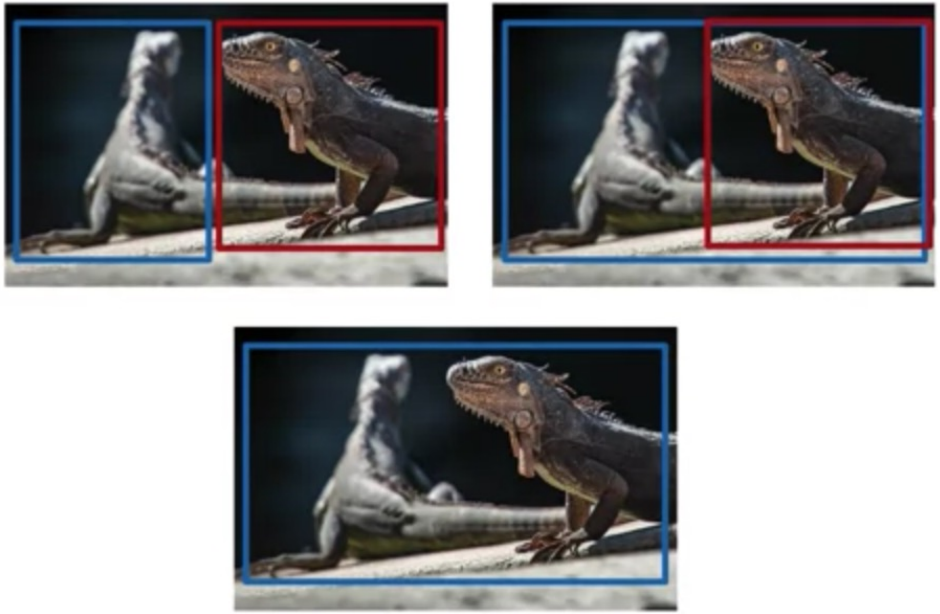


Good Data

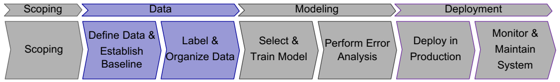
- Ensure data quality in all phases of the project lifecycle
- Good data
  - Inputs cover all important cases
  - Defined consistently and unambiguous
  - Timely feedback from production to development
  - Appropriate volume of data



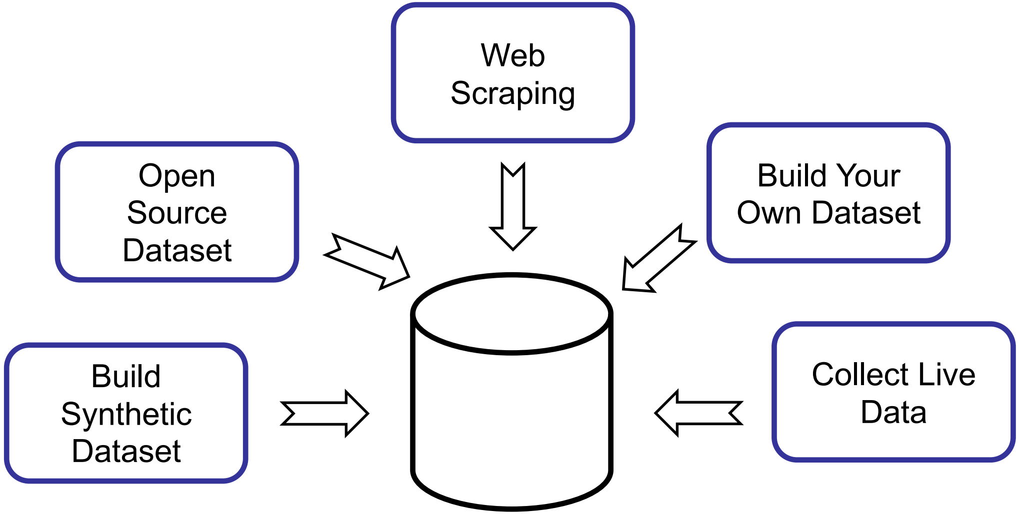
Label Quality

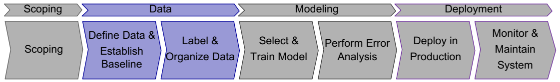


Labeling instructions: "Use bounding boxes to indicate the position of iguanas"

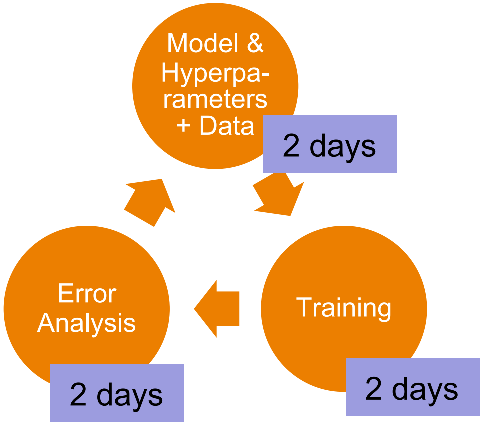


Ways to Obtain Data





Obtaining Data Quickly



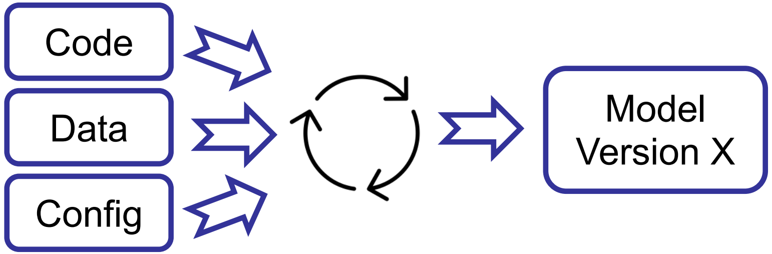
- Quick iterations
- How much data can we collection in *k* days?
  - (not: How long would it take to obtain *m* samples?)
  - Except if we know from prior experience that we need *m* samples

POC vs Production

- Proof of Concept (POC)
  - Goal is to decide if the application is feasible and worth deploying
  - Getting the prototype to work
  - Manual steps are okay, but need to be documented
- Production phase
  - Utility is established
  - Replicable, automated data pipeline
    - » Tensorflow Transform, Apache Beam, Airflow

Tracking Model Lineage

- Information needed to pre-process data and replicate model & results
- Algorithm/code/configuration versioning
- Datasets used
- Hyperparameters
- Experiment results (+summary metrics, analysis)
- Resource monitoring, error analysis, …
- Meta-data

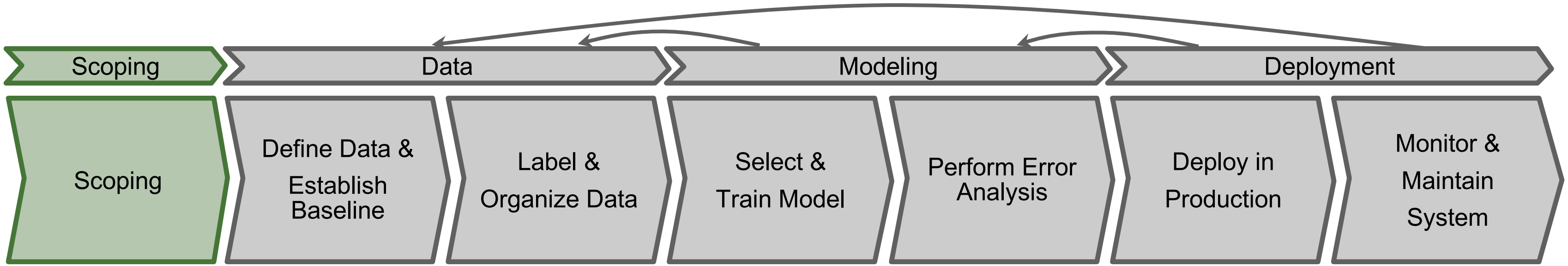


Meta Data

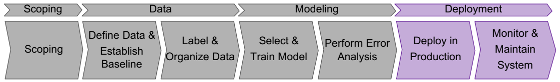
- Not directly needed for model training/prediction
- Useful for error analysis, spotting unexpected effects
- Data provenance

- Time, machines/sensors, camera settings, phone morel, inspector ID, labeller ID, …

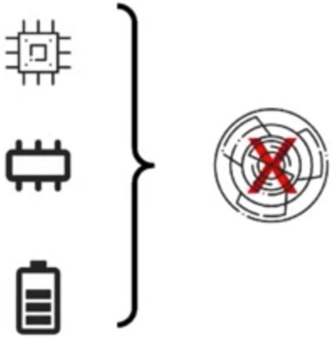
Scoping



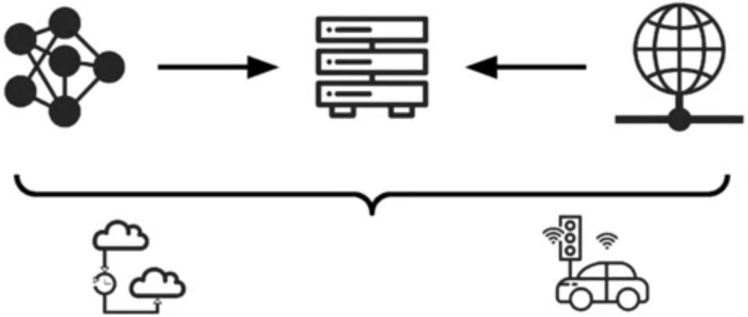
- What projects should we work on?
- What are the metrics for success?
- What resources are needed? (data, time, people)



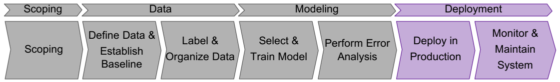
Deployment Options



Edge Devices



Servers / Datacentres

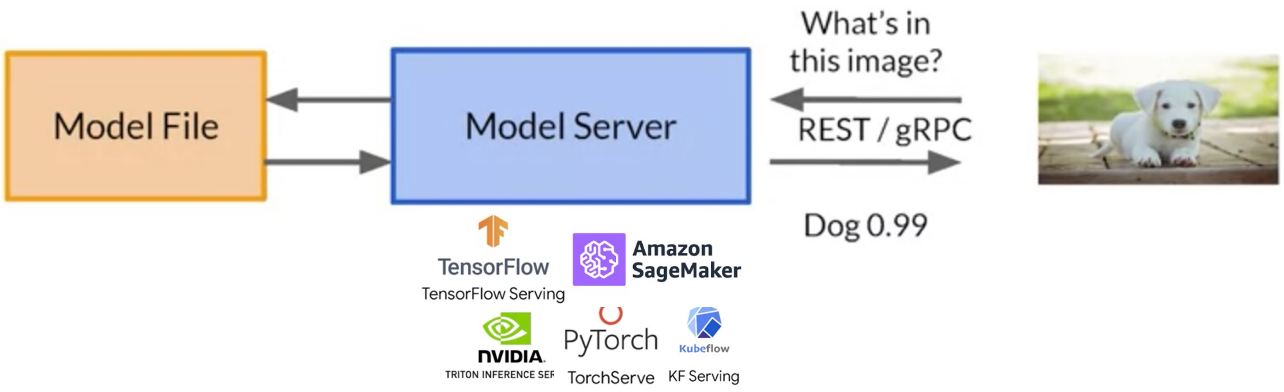


On Prem vs On Cloud

- On Prem
  - Train & deploy on own infrastructure
  - Large companies running ML projects for long time; security
- On Cloud
  - Flexible, on-demand
  - Lower cost in the short term
  - Amazon Web Services, Google Cloud, Microsoft Azure, …

Model Servers

- Simplify task of deploying models at scale
- Scaling, performance, lifecycle management, logging, …



Data Scientist vs Software Engineers

- Data Scientists
  - Often work on fixed datasets
  - Focus on models and metrics
  - Prototyping in Jupyter notebooks
  - Expert in modelling techniques and feature engineering
  - Model size, cost, latency, fairness often ignored

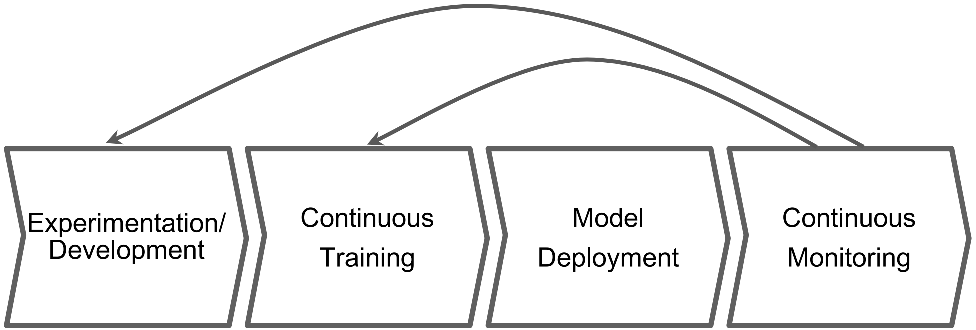
Data Scientist vs Software Engineers

- Software Engineers
  - Build a product
  - Concerned about cost, performance, stability, schedule
  - Quality = Customer satisfucation
  - Scale, large amounts of data
  - Detect and handle errors (automatically)
  - Requirements about security, safety, fairness
  - Maintain, evolve, and extend the product over long periods

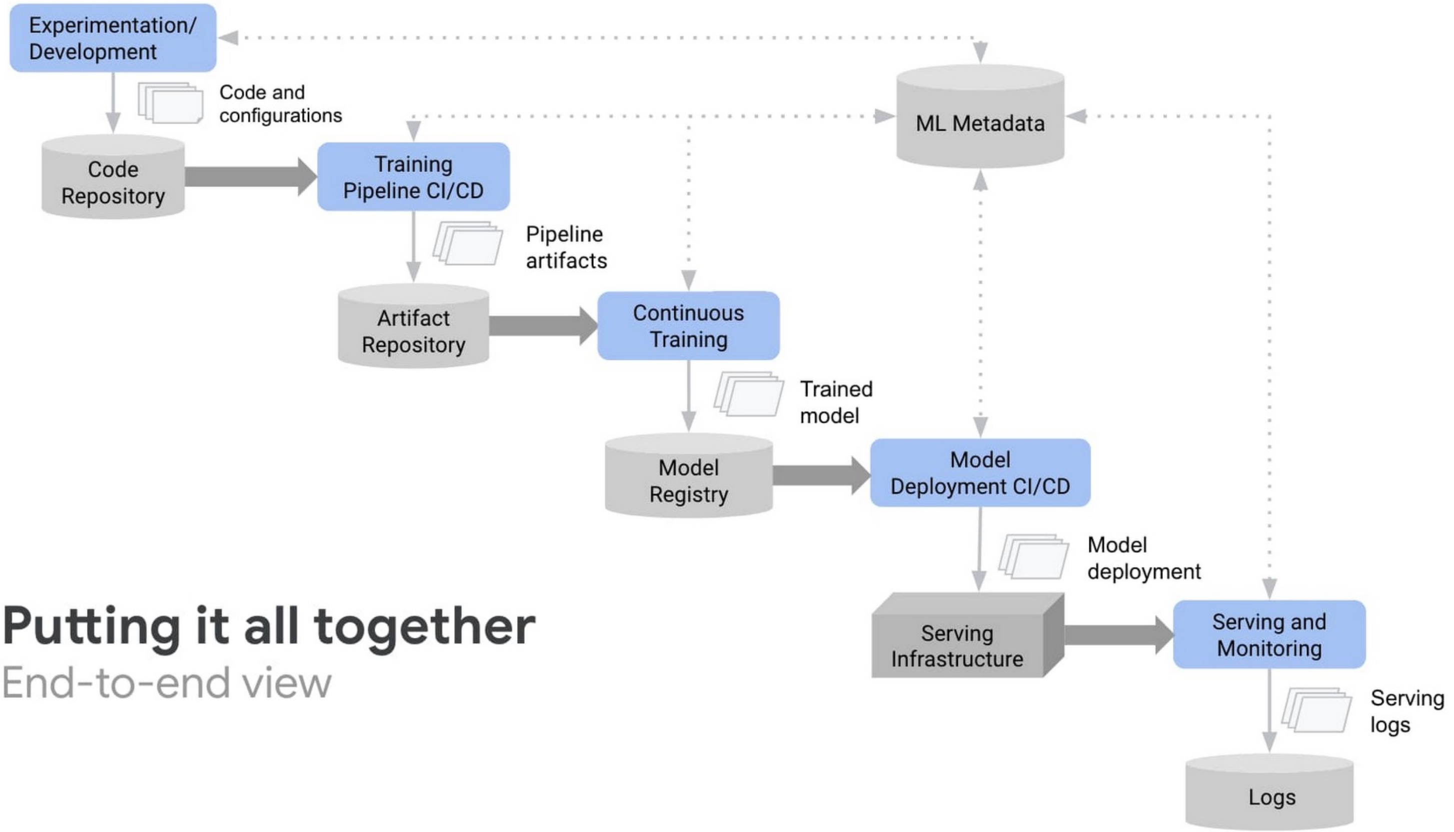
MLOps

- **Continuous Integration (CI):** Testing and validating code, components, data, data schemas, and models
- **Continuous Delivery (CD):** deploying software package/service, model servers
- **Continuous Training (CT):** automatically re-trains models for testing and serving
- **Continuous Monitoring (CM):** Catching errors in production systems, monitoring inference data and model performance metrics

ML Solution Lifecycle



ML



https://cloud.google.com/blog/products/ai-machine-learning/key-requirements-for-an-mlops-foundation

Summary

- An ML Prototype is not a production system
- The task does not end when the system is deployed
- Continuous monitoring and improvement is required throughout the lifetime of the service
- Data centric development is often advantageous
- Model servers and other infrastructure help deploy and scale production systems reliably
- MLOps apply software engineering practices to ML

University of South Australia

Questions?