ML Fridays

Demystifying MLOps

Automating ML Workflows with Amazon SageMaker

Sumir Kumar,
Solutions Architect
AISPL

Tridib Mukherjee
VP Data Science
Games24x7

Agenda

- The current stage of AI/ML practice
- From DevOps to MLOps
- MLOps from 3 different points of view
- Automating ML workflows with Amazon SageMaker
- Demo – Amazon SageMaker Projects and Pipelines
- How Games24x7 is working on MLOps challenges.
- Going forward

Current state of AI/ML

- A decade of ML practices
- Main learnings
- Barriers to AI implementation

State of machine learning

Today

- 53% of POCs make it into production
- Average 9 months

  \- Gartner

Last decade

- Focusing mostly on building ML models
- Operationalization was an afterthought

By end of 2024

- 75% of organizations will shift from piloting to operationalizing AI

  \- Gartner

https://www.idgconnect.com/article/3583467/gartner-accelerating-ai-deployments-paths-of-least-resistance.html

Main learnings

- Publishing a ML model is not enough.
- Managing the published ML models is as important as developing them.

*"IT leaders responsible for AI are discovering **'AI pilot paradox'**, where launching pilots is deceptively easy but deploying them into production is notoriously challenging."*

Chirag Dekate, Senior Director Analyst, Gartner

https://www.gartner.com/smarterwithgartner/gartner-predicts-the-future-of-ai-technologies/

From DevOps to MLOps

- The ML process
- Challenges with productionizing ML
- What is DevOps
- From DevOps to MLOps
- Why MLOps

Release process stages

Source

- Check-in source code
- Peer review new code

Build

- Compile code
- Unit tests
- Style checkers
- Create container images and function deployment p ackages

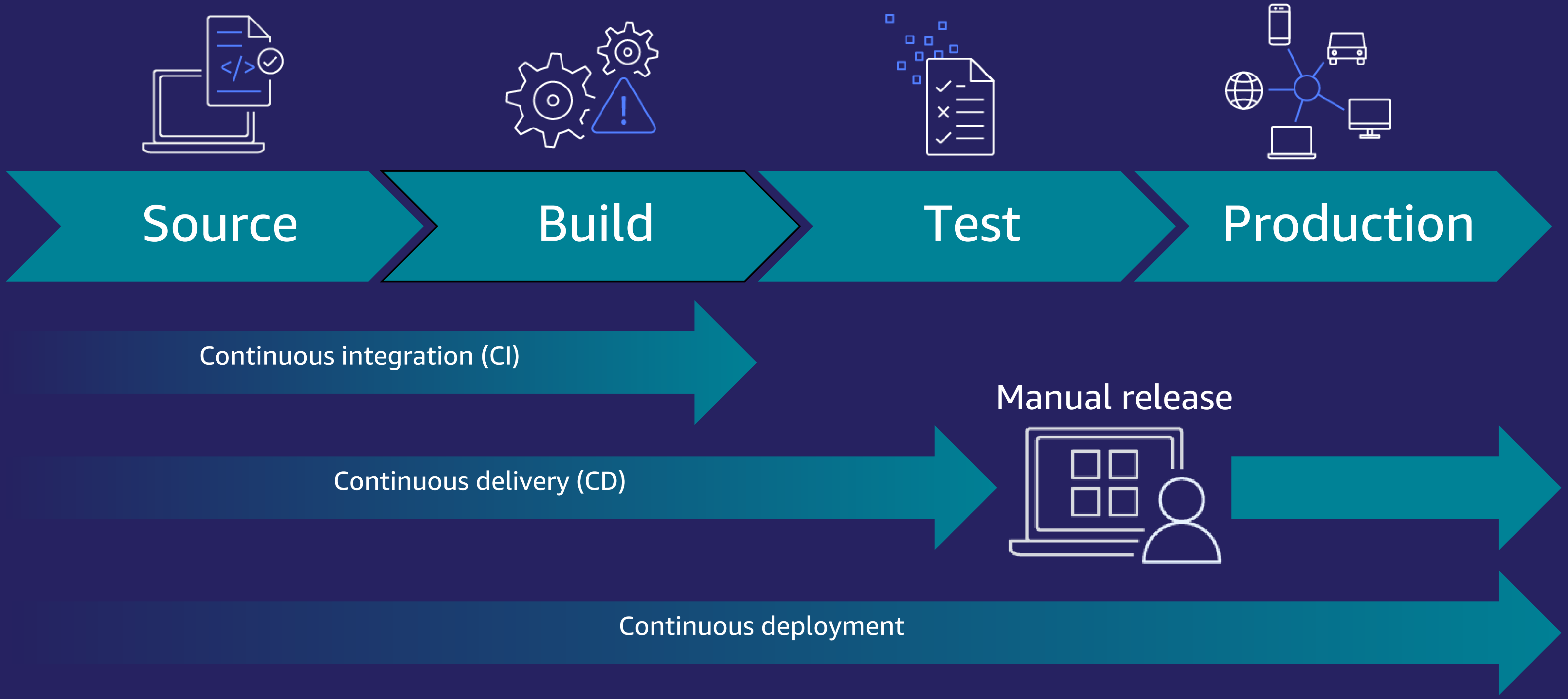
Test

- Integration testing with other systems
- Load testing
- UI testing
- Security testing
- Functional testing
- API testing

Production

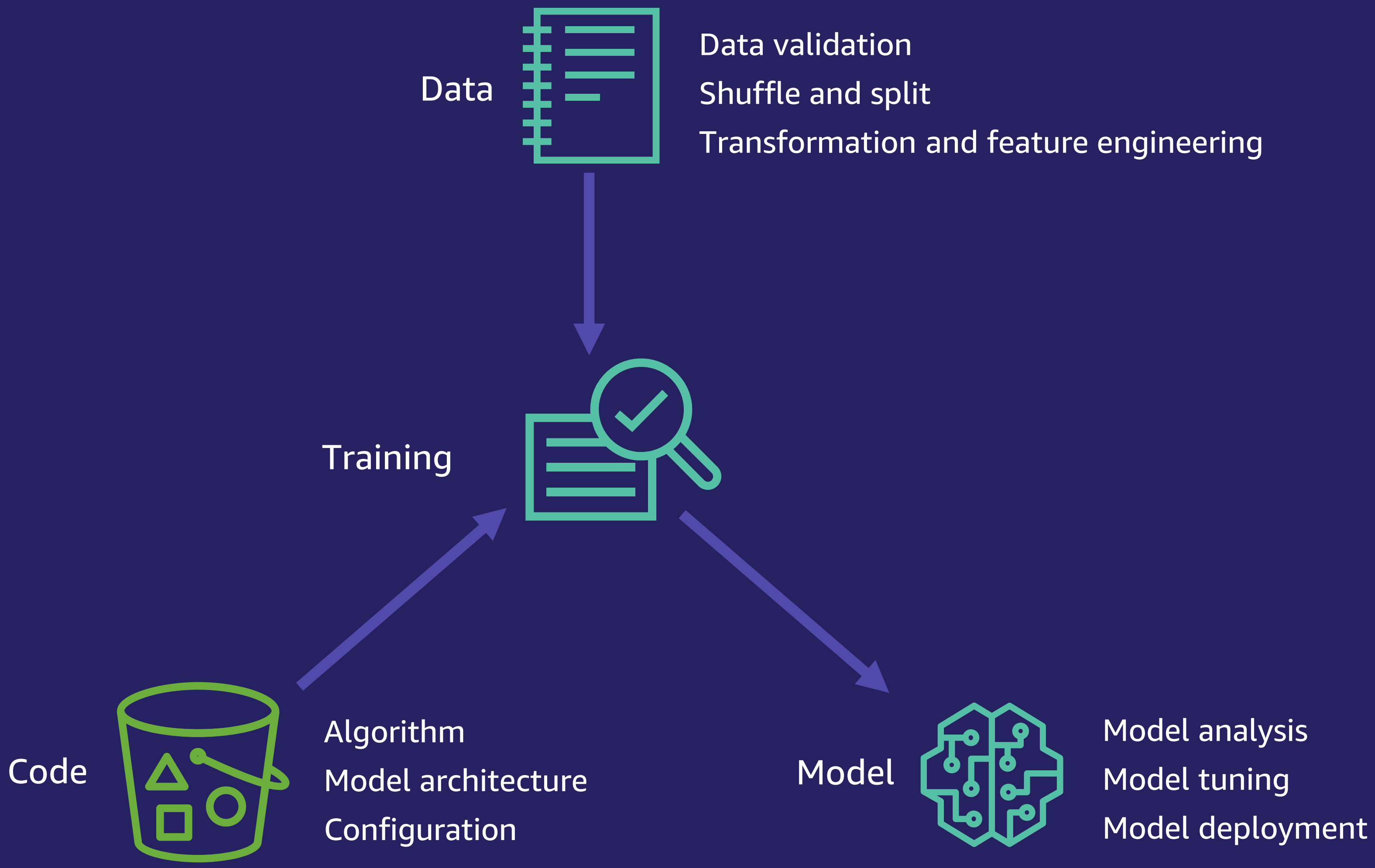
- Deployment to production environments
- Monitor in production to quickly detect any issues errors

Release process automation



CI/CD: Continuous Integration & Continuous Delivery

ML code and data are independent



ML has additional requirements

Consistency	• Minimal variance between environments (i.e. using containers)
Flexibility	• Can accommodate most frameworks
Reproducibility	• Can recreate past experiments/training
Reusability	• Components are reusable across projects
Scalability	• Able to scale resources to efficiently meet demand
Auditability	• Logs, versions and dependencies of artifacts are available
Explainability	• Decision transparency

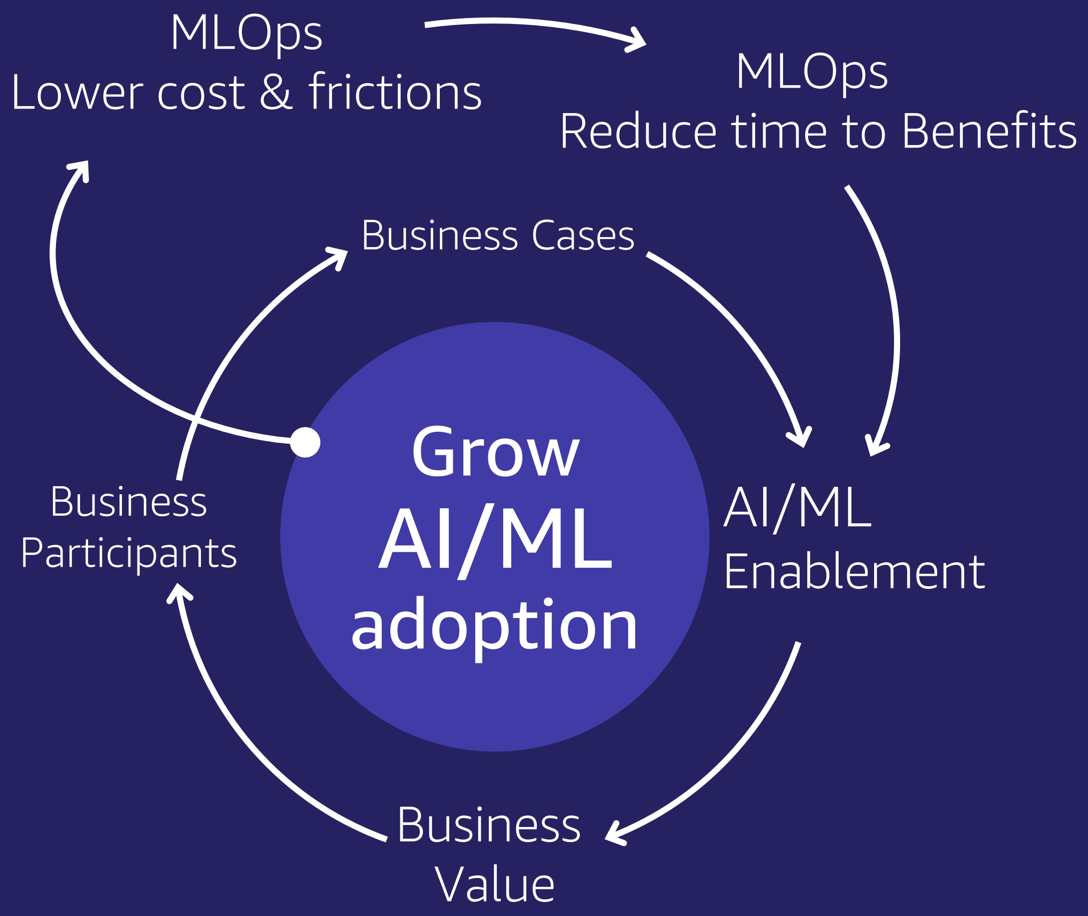
MLOps = DevOps for ML

	DevOps	MLOPS
Code versioning	√	√
Compute environment	√	√
Continuous integration/delivery (CI/CD)	√	√
Monitoring in production	√	√
Data provenance		√
Datasets		√
Models		√
Hyperparameters		√
Metrics		√
Workflows		√

MLOPS
End-to-end ML lifecycle management

https://medium.com/analytics-vidhya/mlops-the-epoch-of-productionizing-ml-models-4eec06d93623

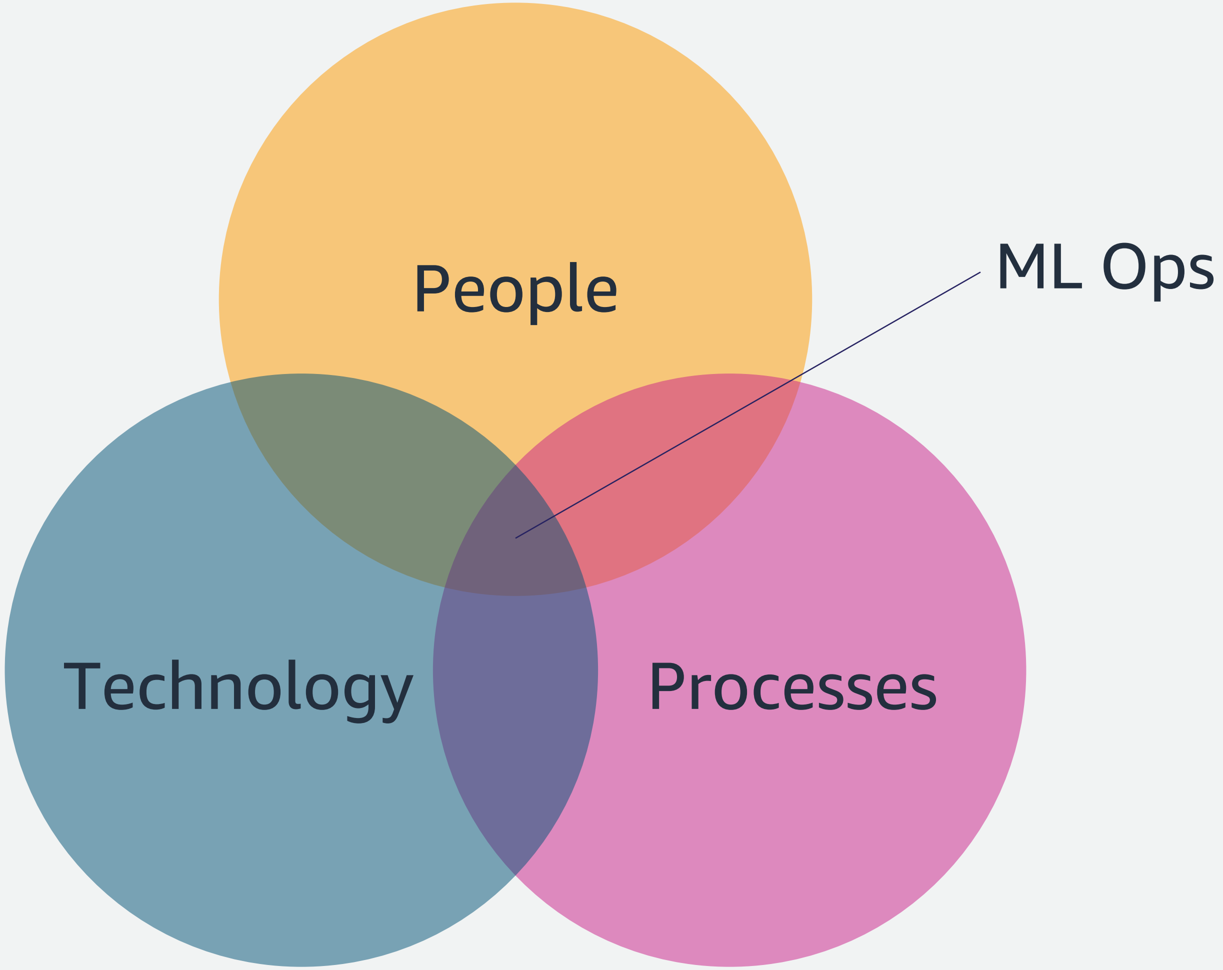
Why MLOPS?


MLOps
Lower cost & frictions
MLOps
Reduce time to Benefits
Business Cases
Grow
AI/ML
adoption
Business
Participants
AI/ML
Enablement
Business
Value

The 3 dimensions of MLOps

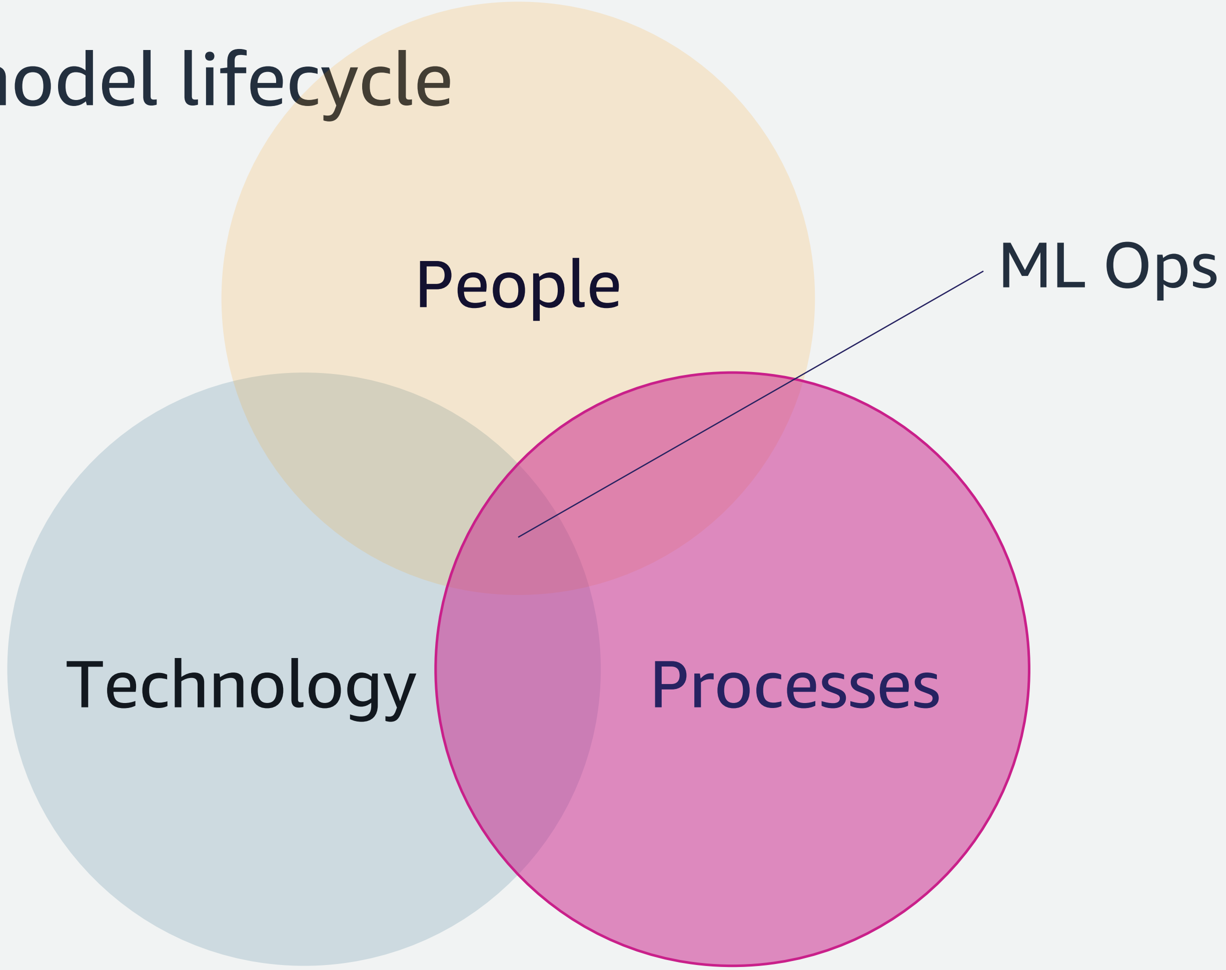
- Processes
- Personas
- Technology

MLOps: Intersection between People, Technology, Process

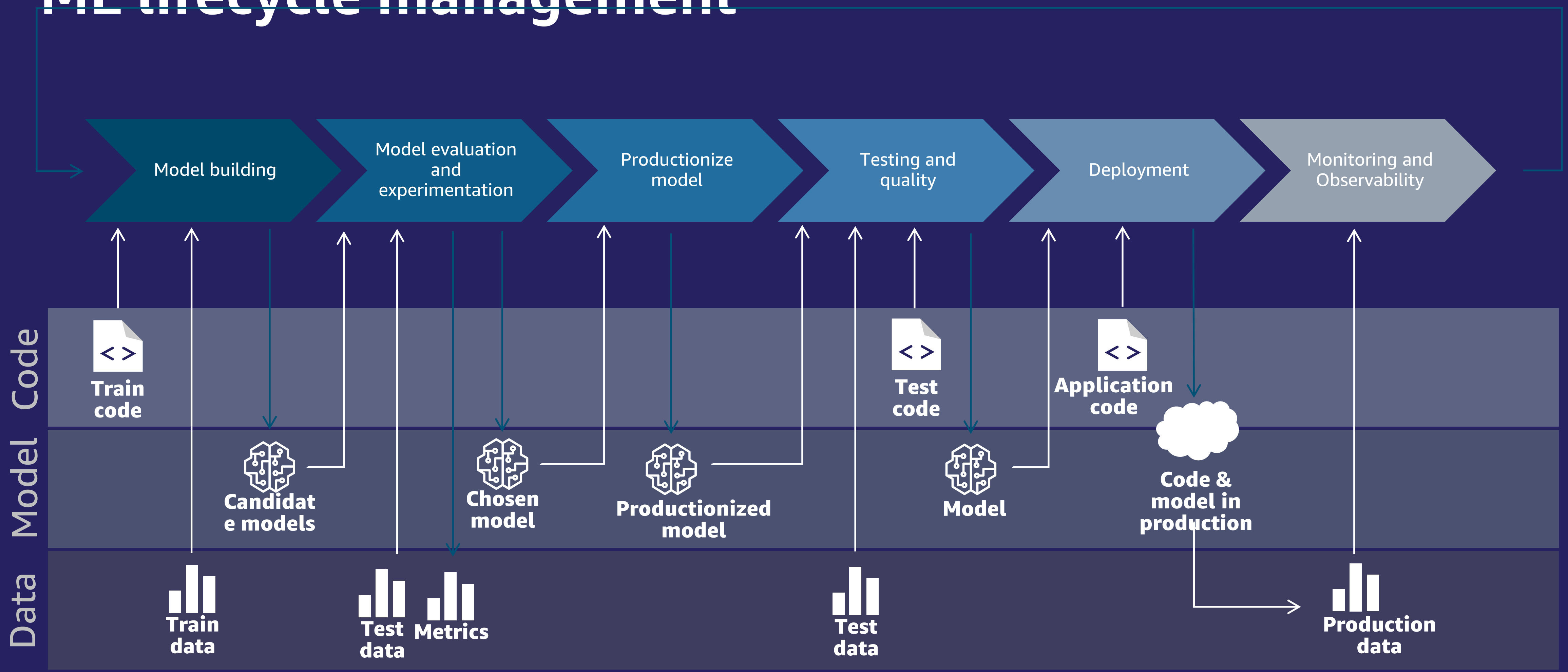


Processes

- End-to-end ML model lifecycle



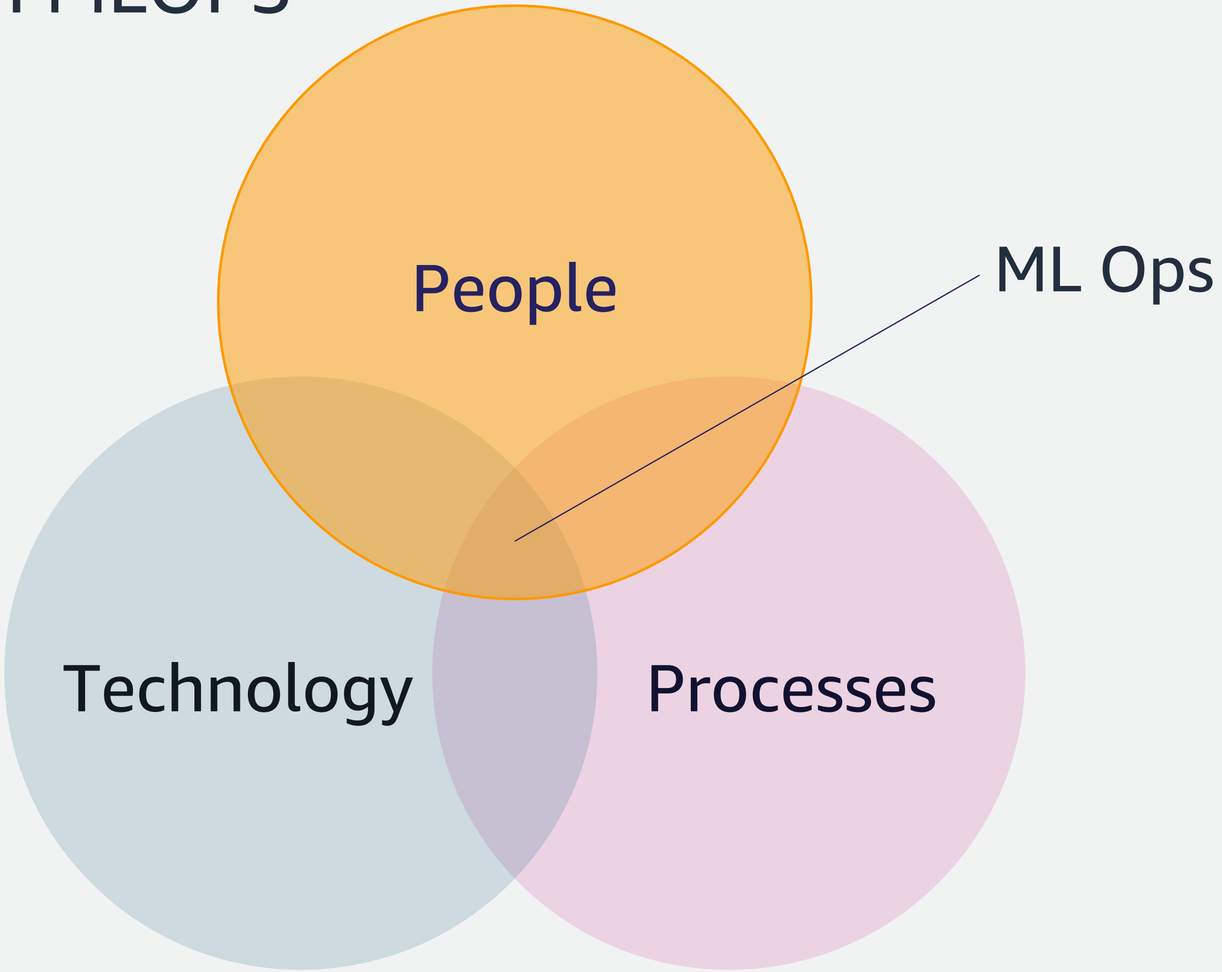
ML lifecycle management



AWS MLOPS whitepaper: https://d1.awsstatic.com/whitepapers/mlops-continuous-delivery-machine-learning-on-aws.pdf

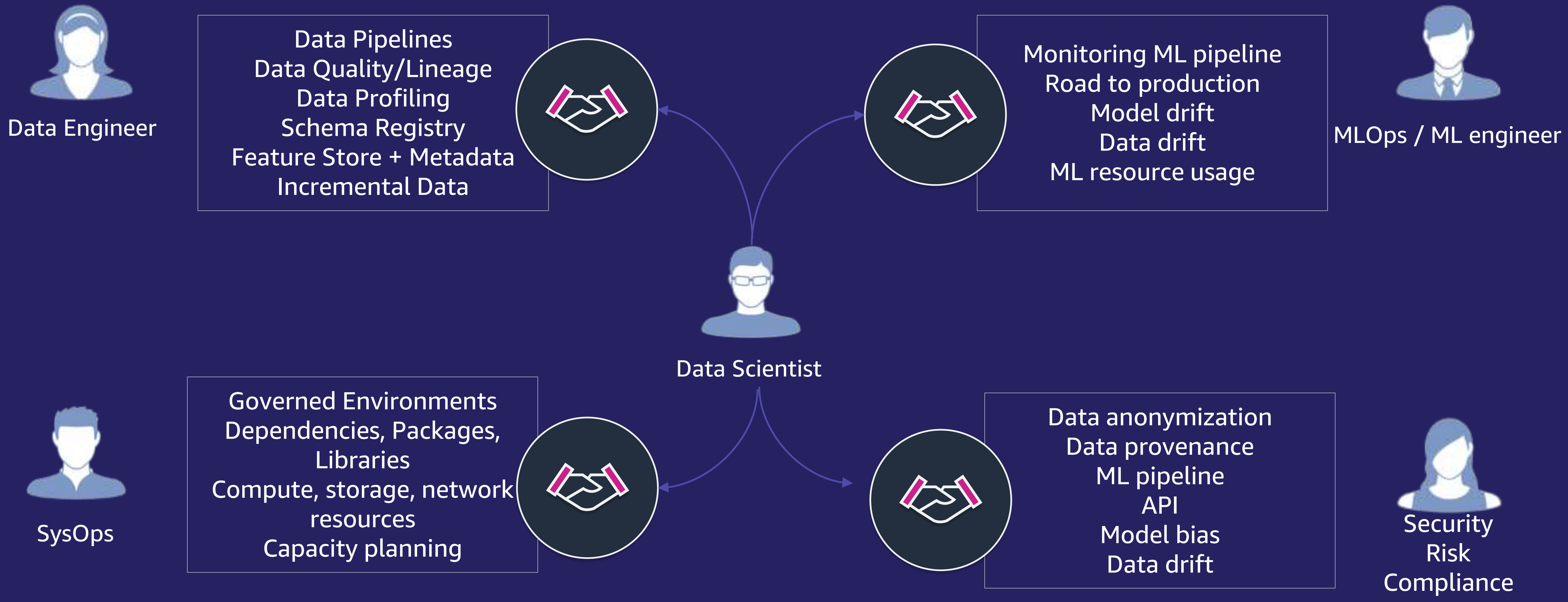
People

Personas involved in MLOPS



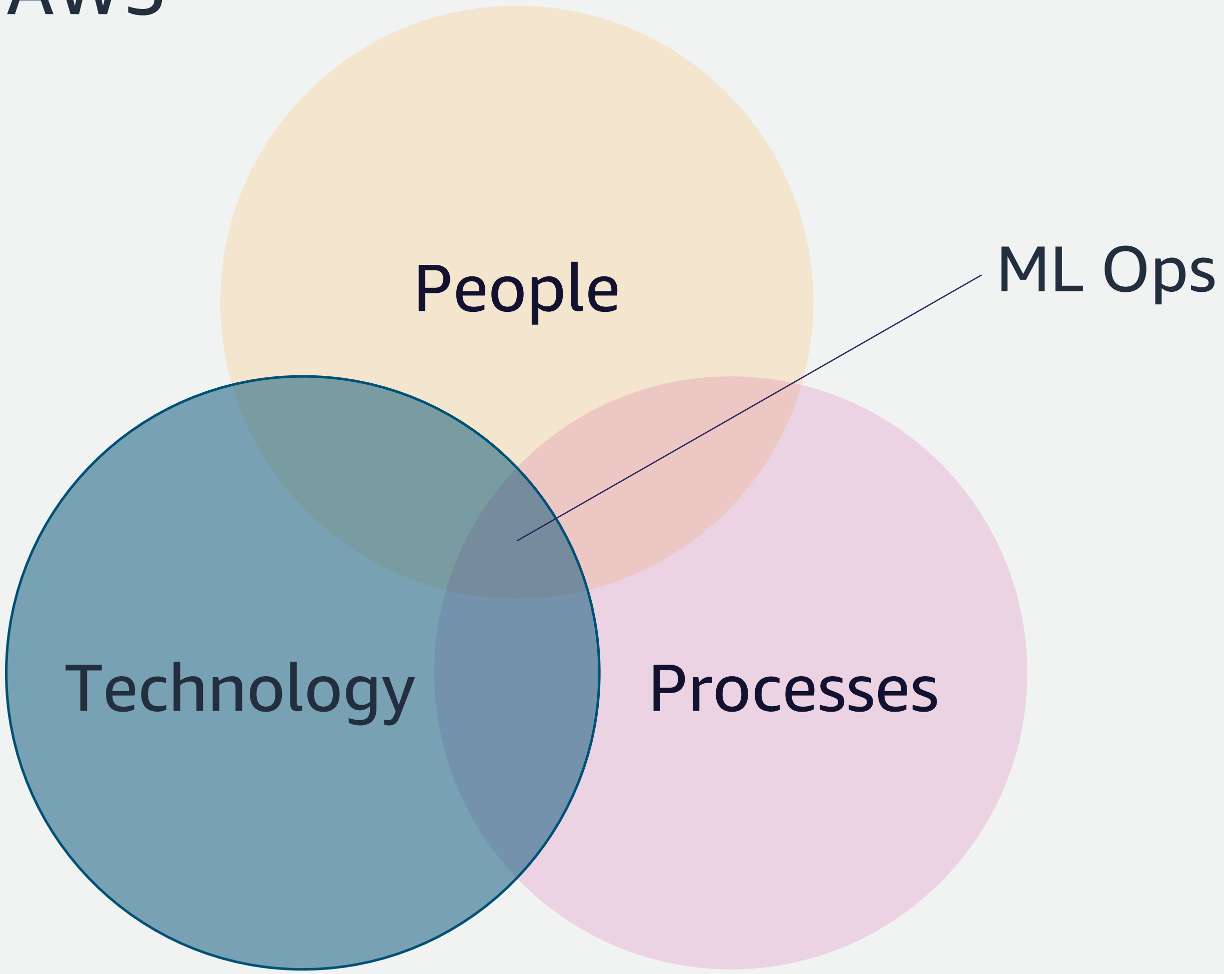
ROLE	PRIORITIES	NEEDS
Data Scientist	Makes sense of data, generates and communicates insights to improve or create business processes, creates predictive ML models to support them	• Data access • ML compute environments • Robust ML tools
Data Engineer	Builds scalable pipelines, transforms and loads data into structures complete with metadata that can be readily consumed by the Data Scientist	• Ad hoc querying • Quick visualization
Security	Risk and Compliance across the enterprise. Prevent data leakages. Audit user activity. Detect model bias.	• Alerts – data leakages, breaches • Reports – data, user activities
MLOps / ML Engineer	Monitoring for reliability, quickly diagnose deployment or availability issues	• Data drift, Model drift • Dashboards
SysOps Engineer	Provision the right infrastructure for the right team without incurring idle resources expenses	• Capacity planning • Resource usage • Governance at scale
Business Sponsor	Vetting the priortization and ROI, funding projects, providing ongoing feedback	• Reporting • Results • Dashboards

ML Road to Production – Collaboration



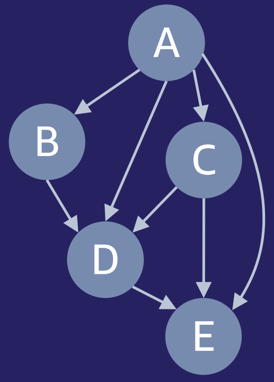
Technology

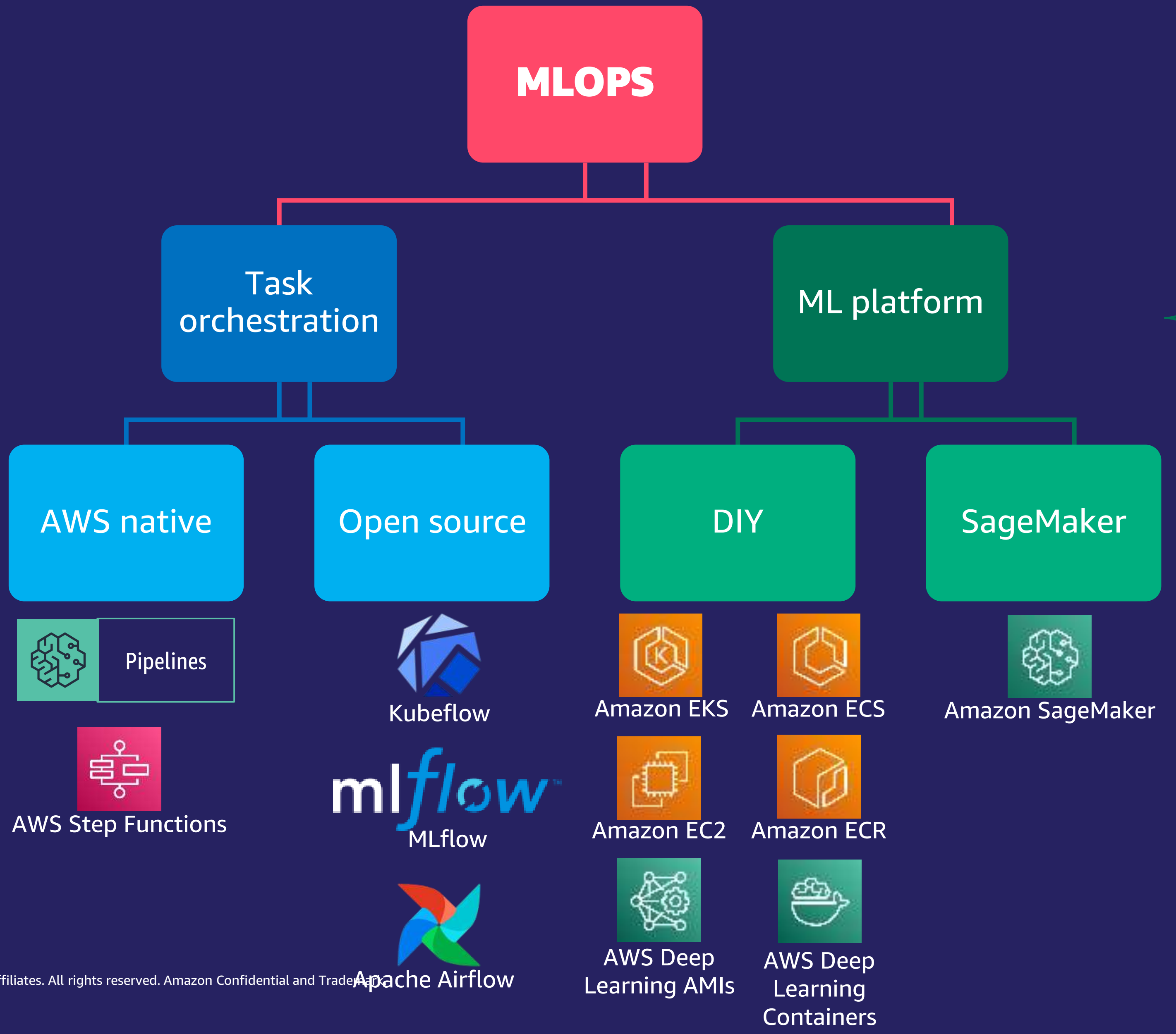
Tools for MLOPS in AWS



Technology components in MLOps

- Create and manage workflows
- Automate ML steps & pipelines
- Implement CI/CD
- Form a Directed Acyclic Graph (DAG)





- ML development, experimentation, collaboration
- Compute/training environment
- Model registry
- Feature store
- Model deployment
- Monitoring in production
- Hyperparameter optimization
- Dataset management

Task orchestration options

Open source 3rd party options

MLflow
Open source platform for the ML lifecycle

Apache Airflow
Platform to author, schedule and monitor workflows

Kubeflow
ML toolkit for Kubernetes

Native AWS options

AWS Step Functions
Serverless pipeline orchestration

Amazon SageMaker Pipelines
Managed ML pipelines in SageMaker Studio

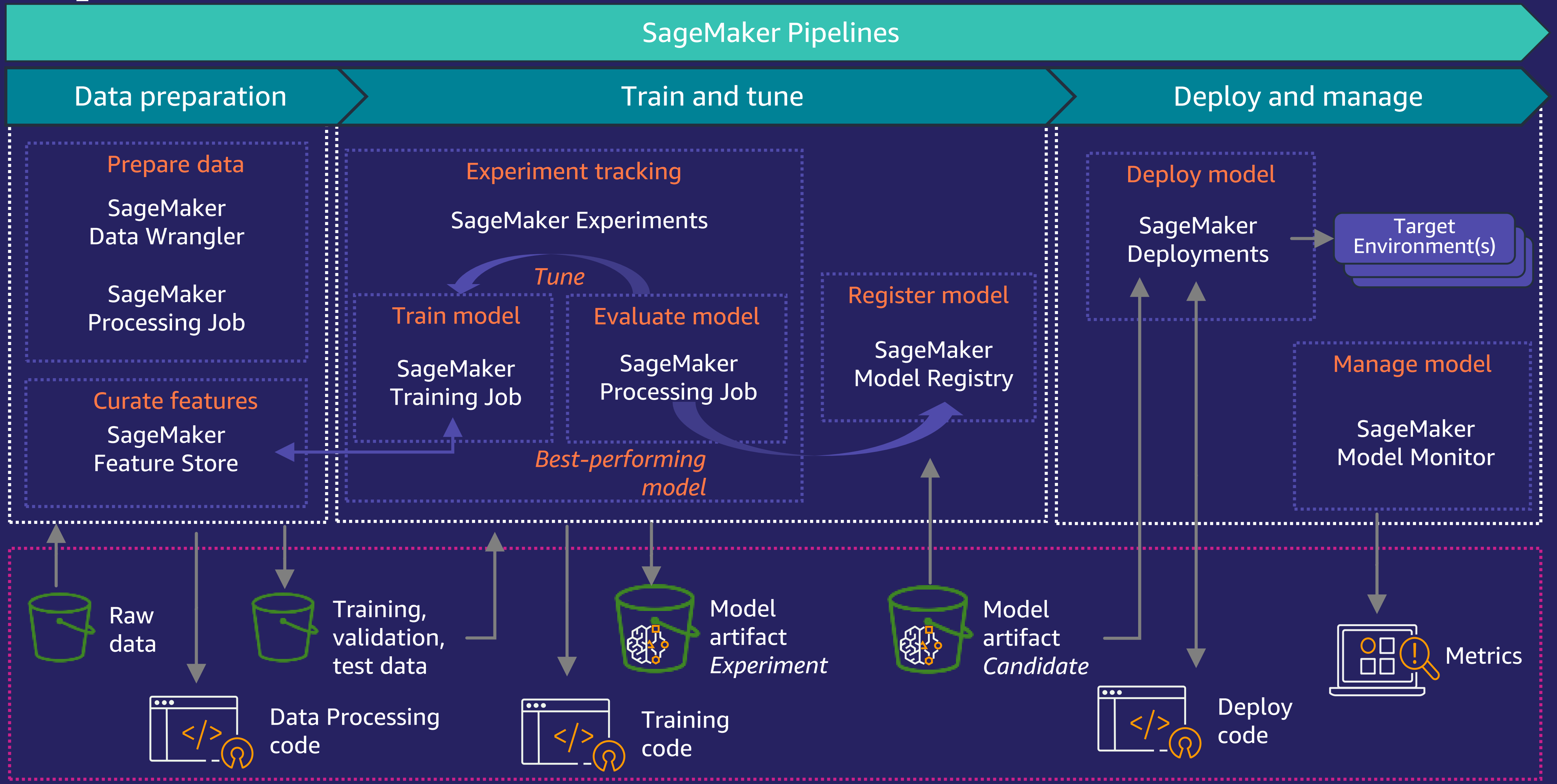
Native integration with SageMaker

Apache Airflow
- SageMaker Operators in Apache Airflow (managed Airflow service)

Amazon Managed Workflows for Apache Airflow

Kubeflow & Kubernetes
- SageMaker Components for Kubeflow Pipelines
- SageMaker Operators for Kubernetes

Amazon SageMaker MLOps-ready features and capabilities



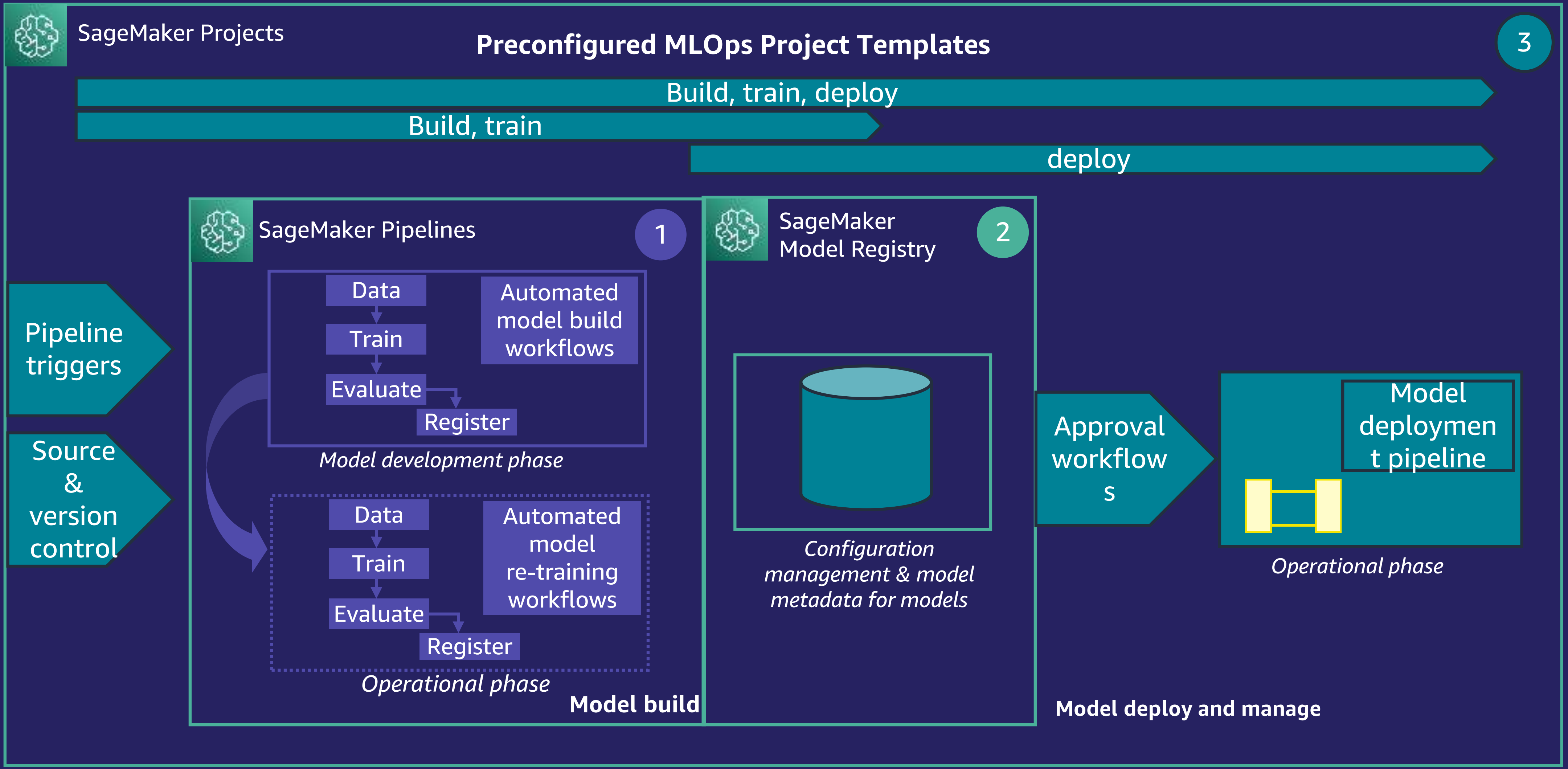
SageMaker Pipelines

Python SDK for quickly and easily building ML workflows

Catalog models to manage models at scale and trigger automated deployment workflows

Built-in support for CI/CD and end-to-end lineage tracking

SageMaker Pipelines: components



Demo – Amazon SageMaker: Projects and Pipelines

Games24x7 Introduction

GAMES | 24 SEVEN

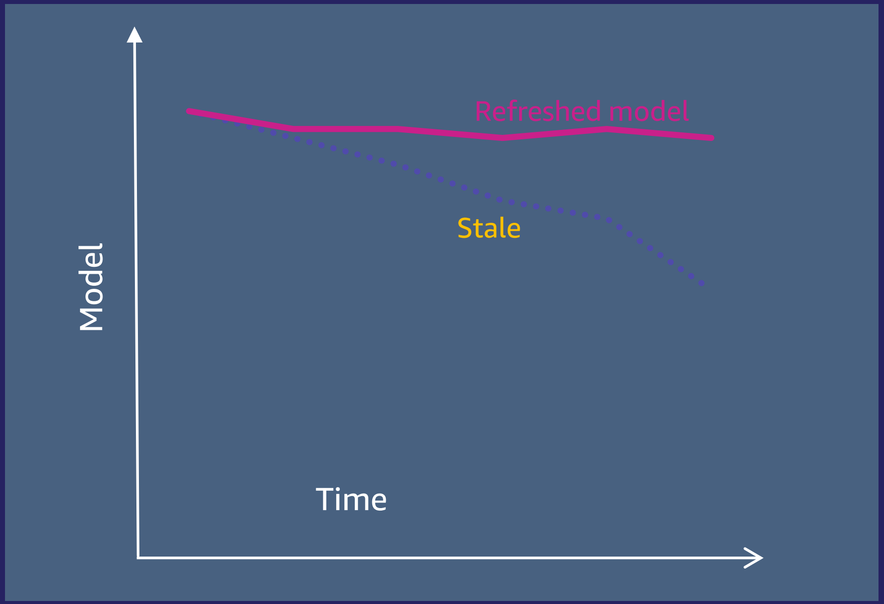
Games24x7 Challenges

- Loss of productivity due to overhead of managing the ML platform.
- Too many tools and interfaces to process data for Machine learning.
- Difficult and slow collaboration between different teams.
- The scale at which models are experimented and deployed by each Data Scientist.
- Tracking multiple models in production and routinely monitor their performance.
- No automation to understand data drift or model drift and re-training of the model is not deterministic.

Operationalizing Model Monitoring with Amazon SageMaker Model Monitor

What happens after the model deployment ?

Deploying a model is not the end. You need to continuously monitor models in production and iterate.



Model accuracy degrade over time

Bias/change in feature attributions

Concept drift due to divergence of data

Amazon SageMaker Model Monitor
Continuous monitoring of models in production

Automatic data collection

Data collected from endpoints is stored in Amazon S3

Continuous monitoring

Define a monitoring schedule and detect changes in quality against a pre-defined baseline

Flexible Monitoring Rules

Use built-in rules to detect drift or write your own rules for custom analysis

Visual data analysis

See monitoring results, data statistics, and violation reports in Amazon SageMaker Studio; Analyze in Notebooks

CloudWatch integration

Metrics emitted to Amazon CloudWatch make it easy to alarm and automate corrective actions

Monitoring Types Supported

- Model Monitor supports monitoring and detection of following types of drift

Data Quality

- **Detect divergence in data**
  - Real time data capture from endpoints
  - Define Baseline
  - Pre-built container for analysis
  - Support custom analysis
  - Type, Num Present, Num Missing
  - Mean, Sum, Std_Dev
  - KLL Sketch

Model Quality

- **Detect quality degradation over time**
  - Merge predictions with ground truth
  - Compare predictions to ground truth
  - Generate reports and violations
  - MAE, RMSE, F1

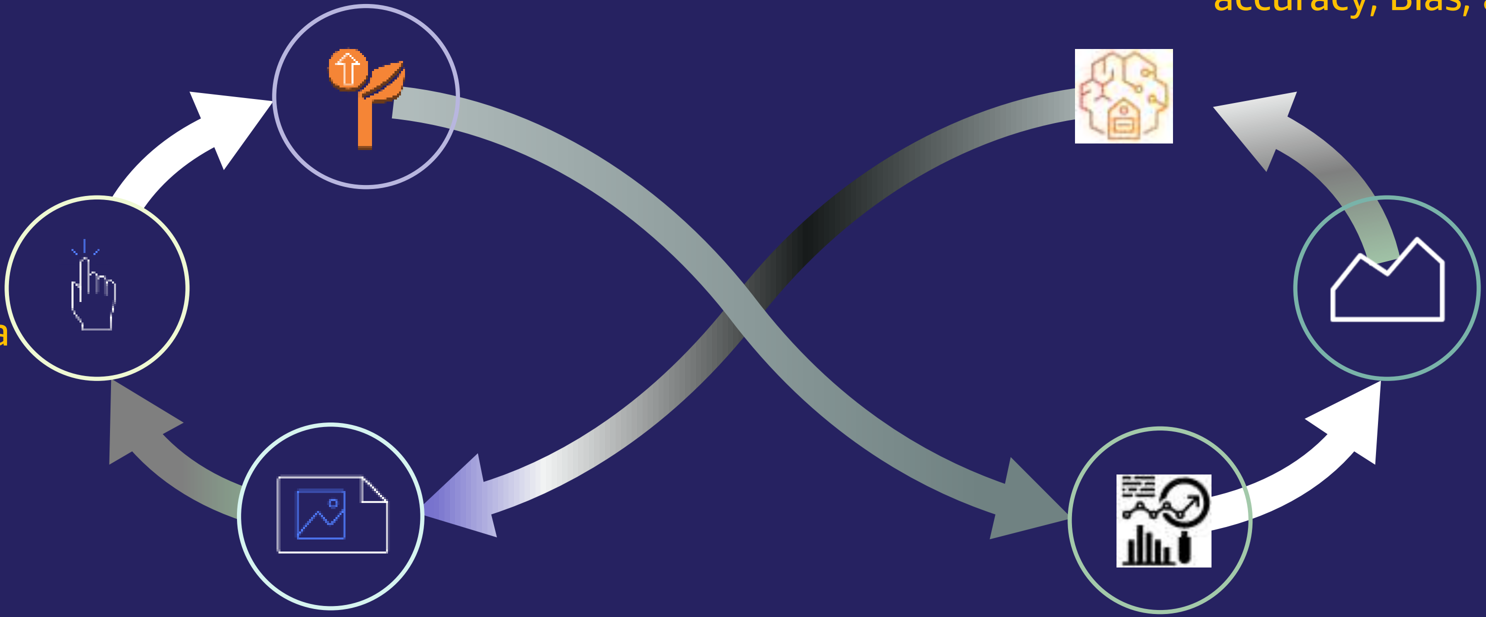
Model Bias

- **Feature Attribution**
  - How much each feature contributed to predictions
  - Shapley Values

Model Explainability

- **Track model balance**
  - Overfitting
  - Underfitting
  - Class Imbalance
  - DPL
  - KL Divergence
  - LP-Norm

Model Monitor – how it works



Deploy Trained Model
Enable Data Capture

Schedule Monitoring Jobs
Detect and monitor Data quality, Model accuracy, Bias, and Explainability drift

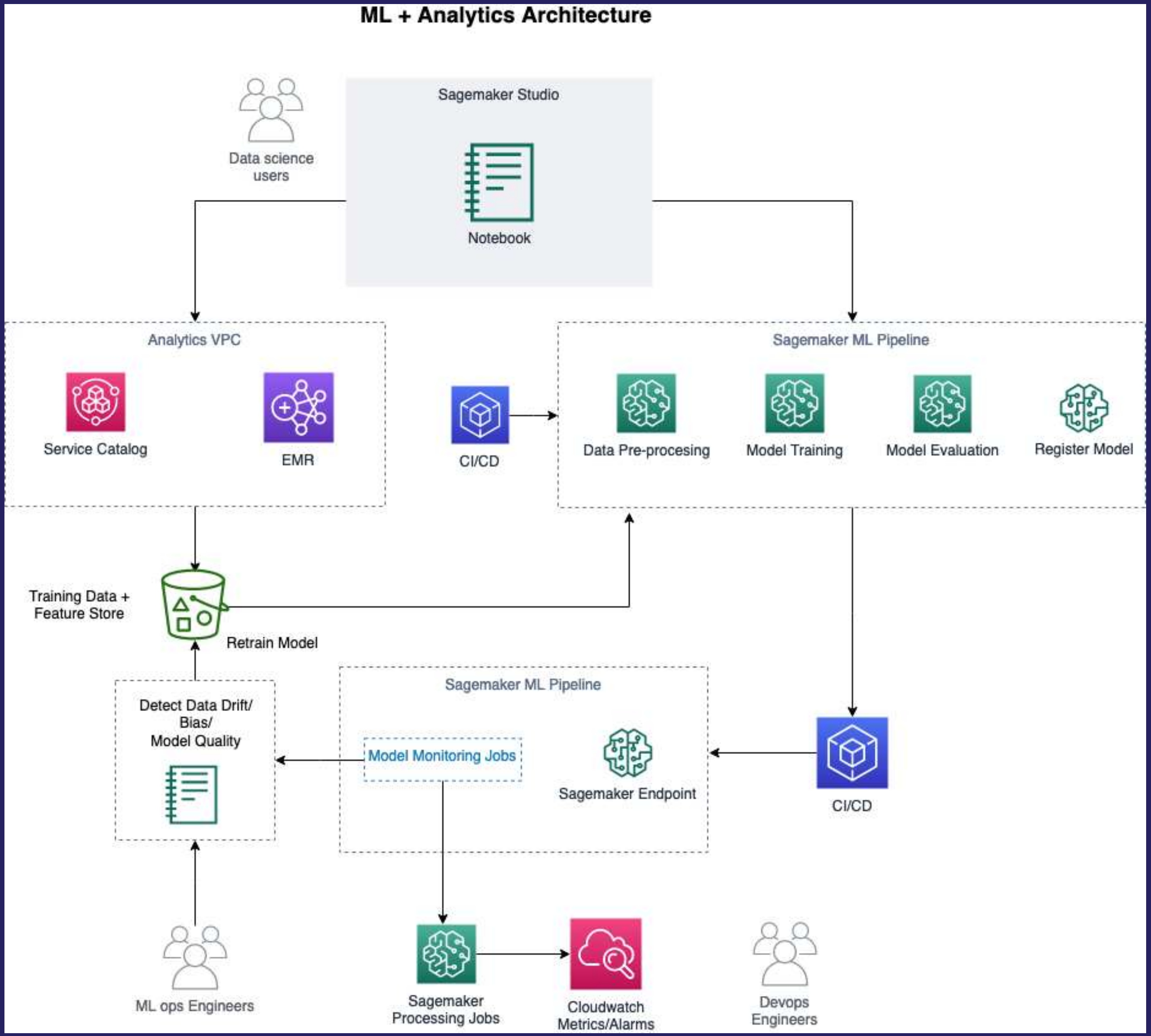
Generate Baseline
Statistics and Constraints

Collect ground truth
Merge with predictions

View drift monitoring results
Reports in S3, Visualize metrics in Studio, Analyze in Notebooks

CloudWatch Alerts
Retrain models / Update data

ML Platform architecture – In progress



Way Forward

- Self service platform for model/notebook as a service with EMR/Spark/Hive/Presto using SageMaker Studio.
- Monitoring and detecting drift in our models using SageMaker Studio.
- Standardize MLOps practices as we scale.

Further resources

Further resources

AWS White papers

- AWS MLOPS: https://d1.awsstatic.com/whitepapers/mlops-continuous-delivery-machine-learning-on-aws.pdf
- AWS Well-Architected Framework for Machine Learning: https://docs.aws.amazon.com/wellarchitected/latest/machine-learning-lens/wellarchitected-machine-learning-lens.pdf
- Sagemaker Workshop:
- https://catalog.us-east-1.prod.workshops.aws/workshops/63069e26-921c-4ce1-9cc7-dd882ff62575/en-US/
- Deep Learning on AWS: https://d1.awsstatic.com/whitepapers/Deep_Learning_on_AWS.pdf
- Amazon SageMaker Total Cost of Ownership: https://pages.awscloud.com/rs/112-TZM-766/images/Amazon_SageMaker_TCO_uf.pdf

AWS MLOPS Framework

https://aws.amazon.com/solutions/implementations/aws-mlops-framework/

Further resources

Self-paced workshops & repositories

- MLOPS across 4 personas: https://github.com/imyoungyang/myAWSStudyBlog/tree/master/ml-ops-poc
- Data Science on AWS (ML end-to-end pipeline): https://github.com/data-science-on-aws/workshop
- Amazon SageMaker MLops, with classic CI/CD tools: https://github.com/awslabs/amazon-sagemaker-mlops-workshop
- Basic SageMaker MLOps: https://github.com/aws-samples/mlops-amazon-sagemaker-devops-with-ml
- Serverless ML pipeline: https://github.com/dylan-tong-aws/aws-serverless-ml-pipeline
- Operationalizing the ML pipeline: https://operational-machine-learning-pipeline.workshop.aws/
- Safe MLOps deployment pipeline: https://mlops-safe-deployment-pipeline.workshop.aws/
- MLOps and integrations: https://mlops-and-integrations.workshop.aws/

Thank you!