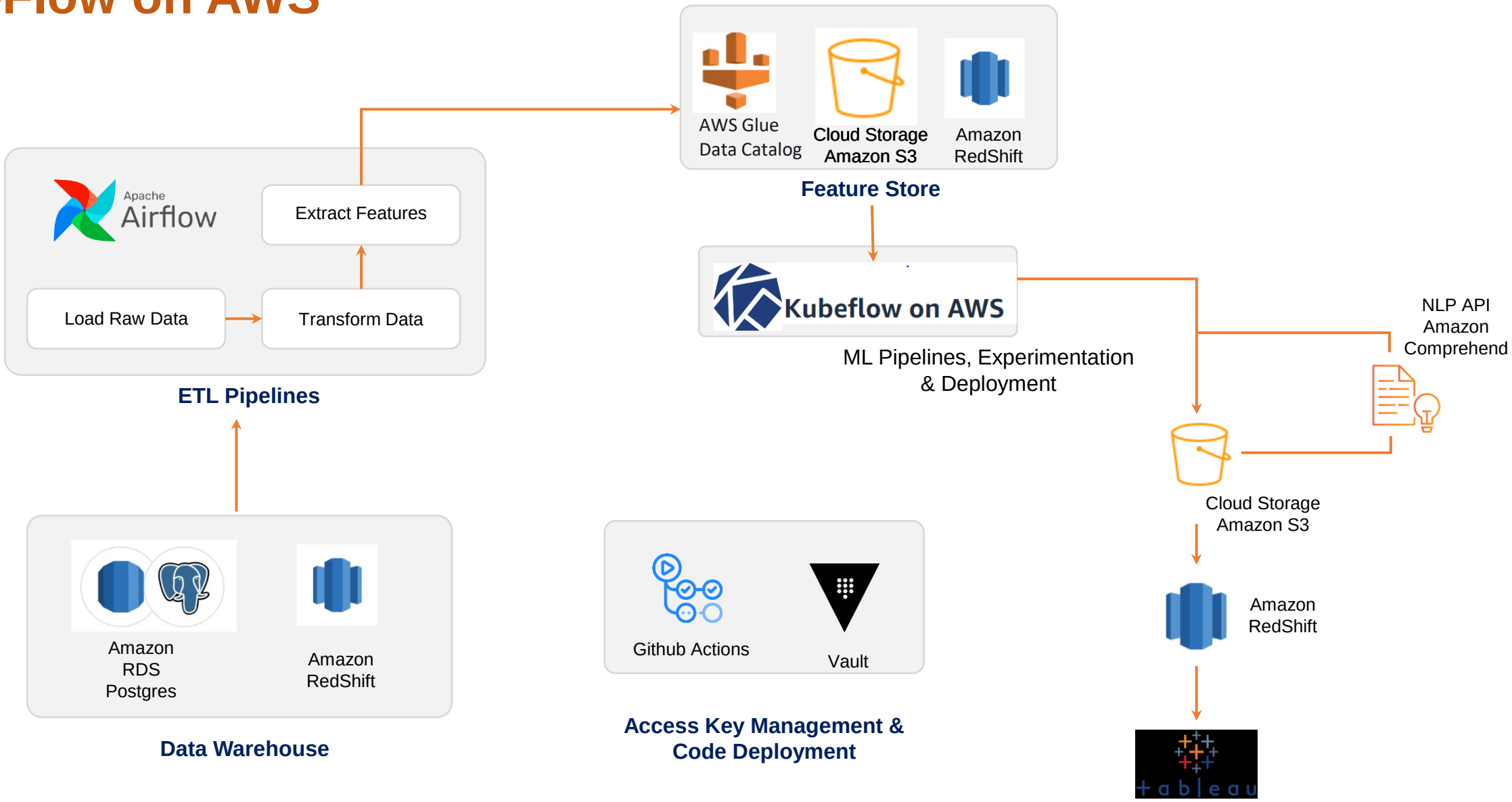
AWS MLOPS Solution Architecture

Propose Architecture KubeFlow on AWS

Option 1



Note: This architecture is based on current analysis of Kubeflow services and may change upon further exploration of Kubeflow

Option 1 - KubeFlow on AWS

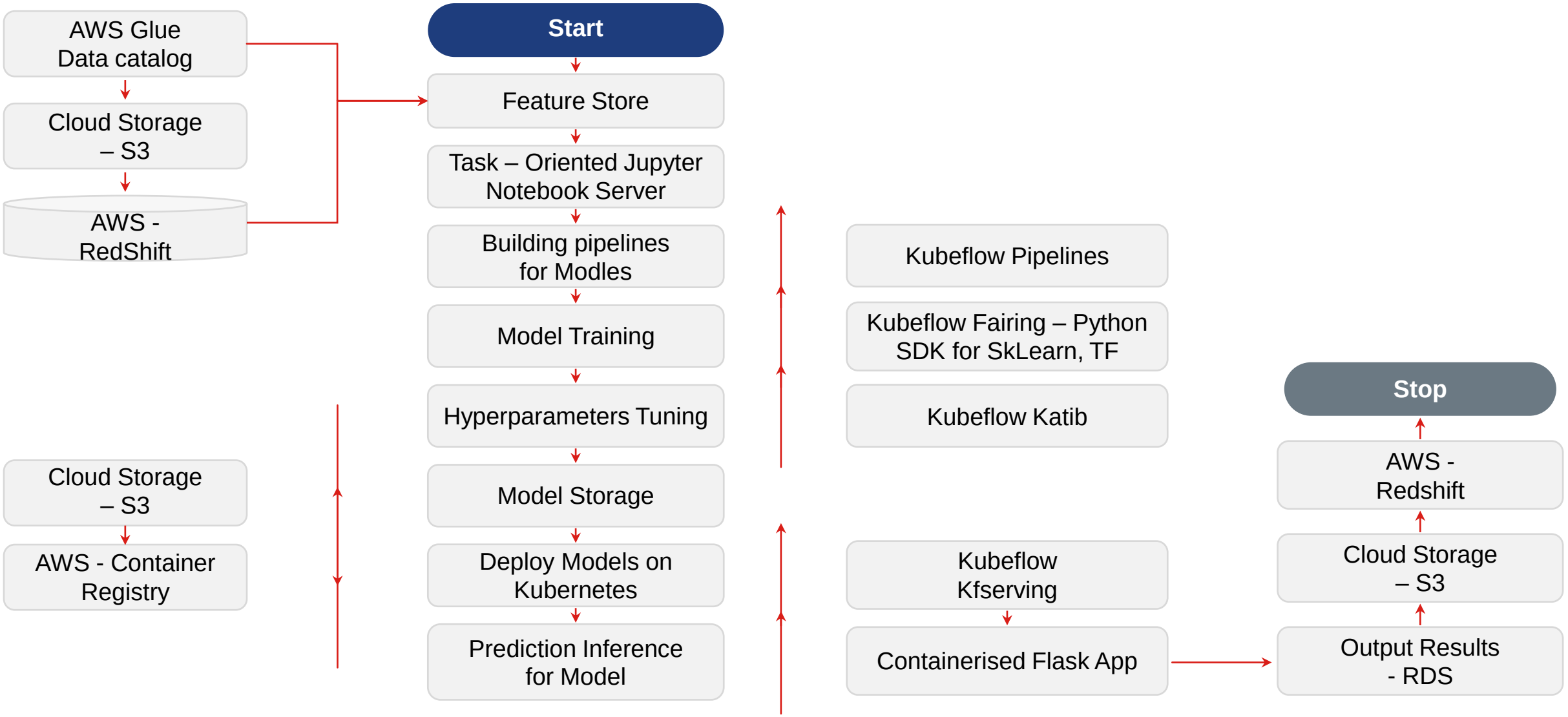
- The Option -1 leverages the GCP Solution used for retailer with Kubeflow as an enabler supported by AWS Services for Data Engineering.
- Use Airflow for ETL Pipeline (Same as GCP approach)
- AWS Services for Data Engineering
  - AWS Services Redshift & Amazon RDS for Datawarehouse
  - AWS RDS – Postgress SQL for Cloud Storage
  - AWS Glue Catalog for Data Catalog requirement
  - AWS S3 for Cloud Storage
  - AWS Comprehend for NLP functionality (Choose API as per Business requirements)

- Use KubeFlow on AWS for ML Pipeline, Experimentation and Deployment

Kubeflow for AWS - Overview

- AWS recently launched Kubeflow v1.4 as part of its own Kubeflow distribution (called Kubeflow on AWS), which streamlines data science tasks and helps build highly reliable, secure, portable, and scalable ML systems with reduced operational overheads through integrations with AWS managed services.
- Kubeflow distribution to build ML systems on top of Amazon Elastic Kubernetes Service (Amazon EKS) to build, train, tune, and deploy ML models for a wide variety of use cases, including computer vision, natural language processing, speech translation, and financial modeling.

AWS Kubeflow Process Flow



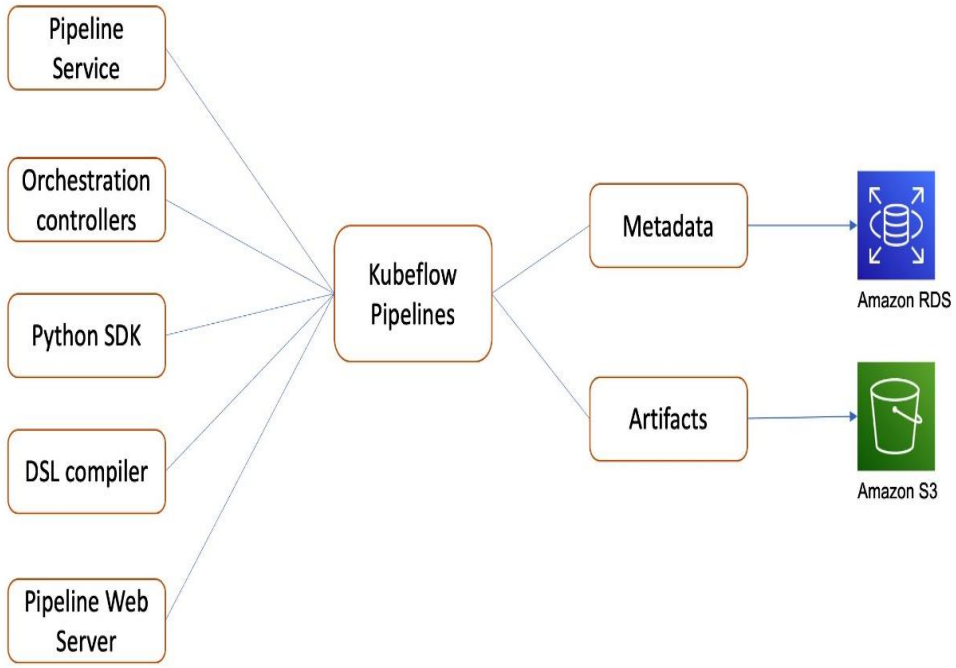
SIGMOID

Kubeflow Pipeline

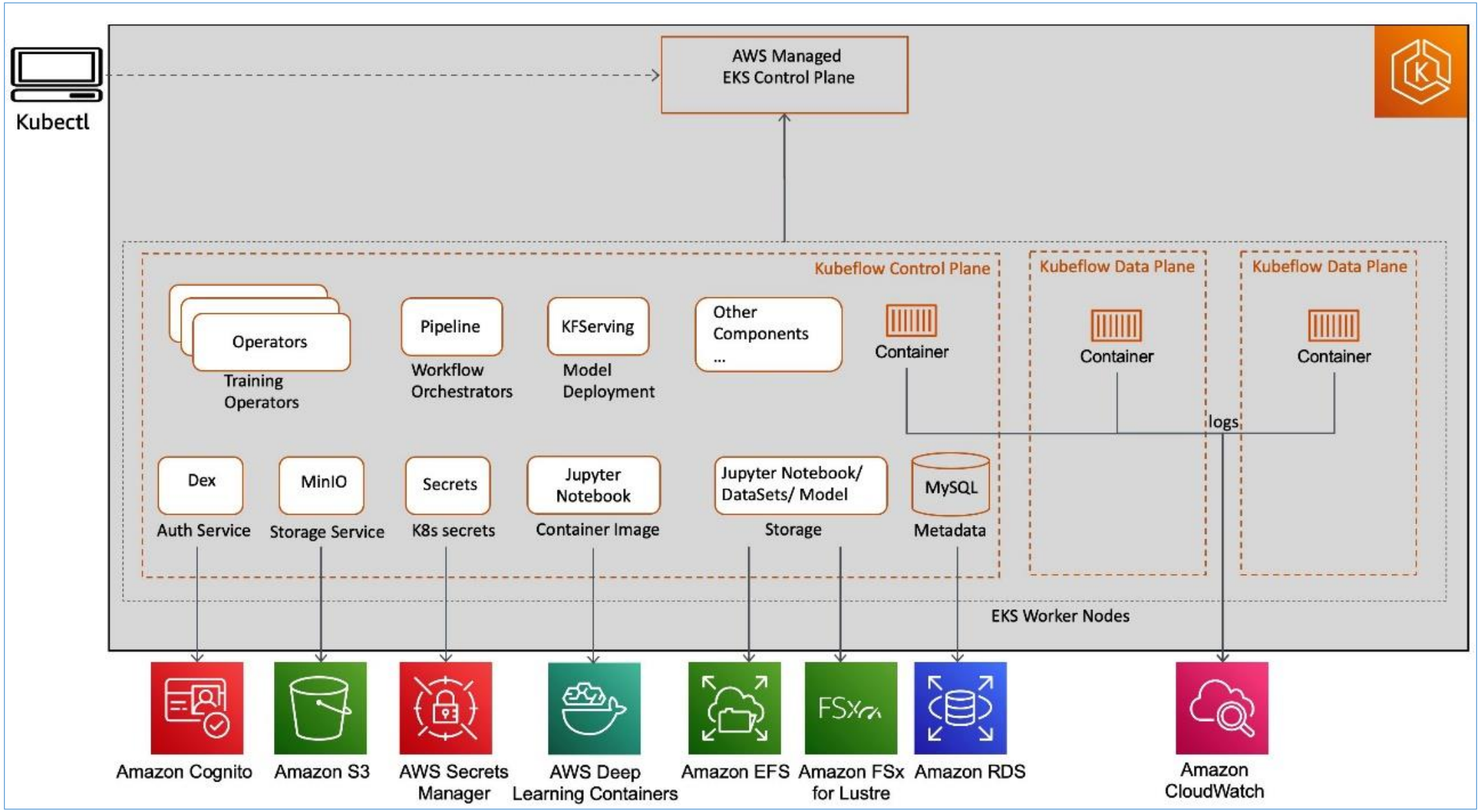
Kubeflow Pipelines includes Python SDK, a DSL compiler to convert Python code into a static config, a Pipelines service that runs pipelines from the static configuration, and a set of controllers to run the containers within the Kubernetes Pods needed to complete the pipeline.

- Kubeflow Pipelines metadata for pipeline experiments and runs are stored in MySQL, and artifacts including pipeline packages and metrics are stored in MinIO.
    - Pipeline metadata in Amazon RDS – Amazon RDS provides a scalable, highly available, and reliable Multi-AZ deployment architecture with a built-in automated failover mechanism and resizable capacity for an industry-standard relational database like MySQL.
    - Pipeline artifacts in Amazon S3 – Amazon S3 offers industry-leading scalability, data availability, security, and performance, and could be used to meet your compliance requirements.
- These integrations help offload the management and maintenance of the metadata and artifact storage from self-managed Kubeflow to AWS managed services, which is easier to set up, operate, and scale.

- Kubeflow Pipelines is a platform for building and deploying portable, scalable ML workflows.
- These workflows can help automate complex ML pipelines using built-in and custom Kubeflow components



AWS Service integrations with Kubeflow Pipeline



Kube Flow – Integration with AWS Services

Application Load Balancer for secure external traffic management over HTTPS

Amazon CloudWatch for persistent log management

AWS Cognito for user authentication with Transport Layer Security (TLS)

AWS Deep Learning Containers for highly optimized Jupyter notebook server images

Amazon Elastic File System (Amazon EFS) or Amazon FSx for Lustre for a simple, scalable, and serverless file storage

Amazon EKS for managed Kubernetes clusters

Amazon Relational Database Service (Amazon RDS) for highly scalable pipelines and a metadata store

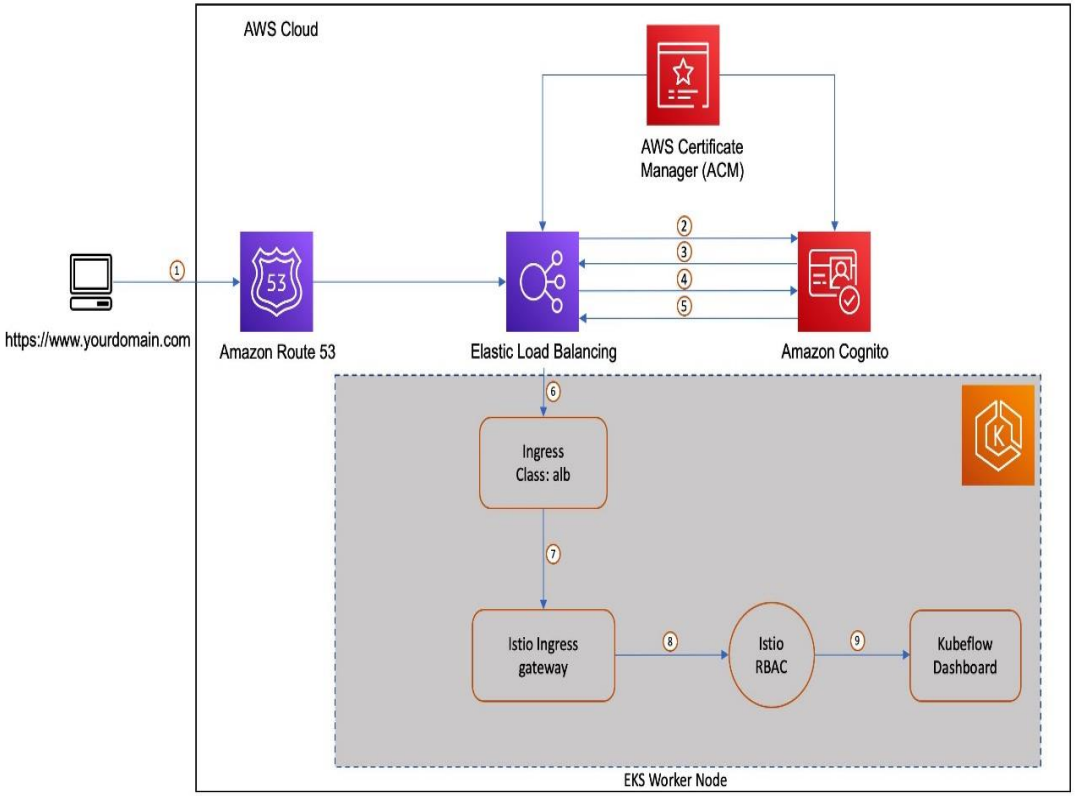
AWS Secrets Manager to protect secrets needed to access your applications

Amazon Simple Storage Service (Amazon S3) for an easy-to-use pipeline artifacts store

Decouple critical parts of the Kubeflow control plane from Kubernetes, providing a secure, scalable, resilient, and cost-optimized design.

Secure authentication of Kubeflow users with Amazon Cognito

- Application Load Balancer (ALB) for external traffic management
- AWS Certificate Manager (ACM) to support TLS
- IAM roles for service accounts (IRSA) for fine-grained access control at the Kubernetes Pod level
- AWS Key Management Service (AWS KMS) for data encryption key management
- AWS Shield for DDoS protection

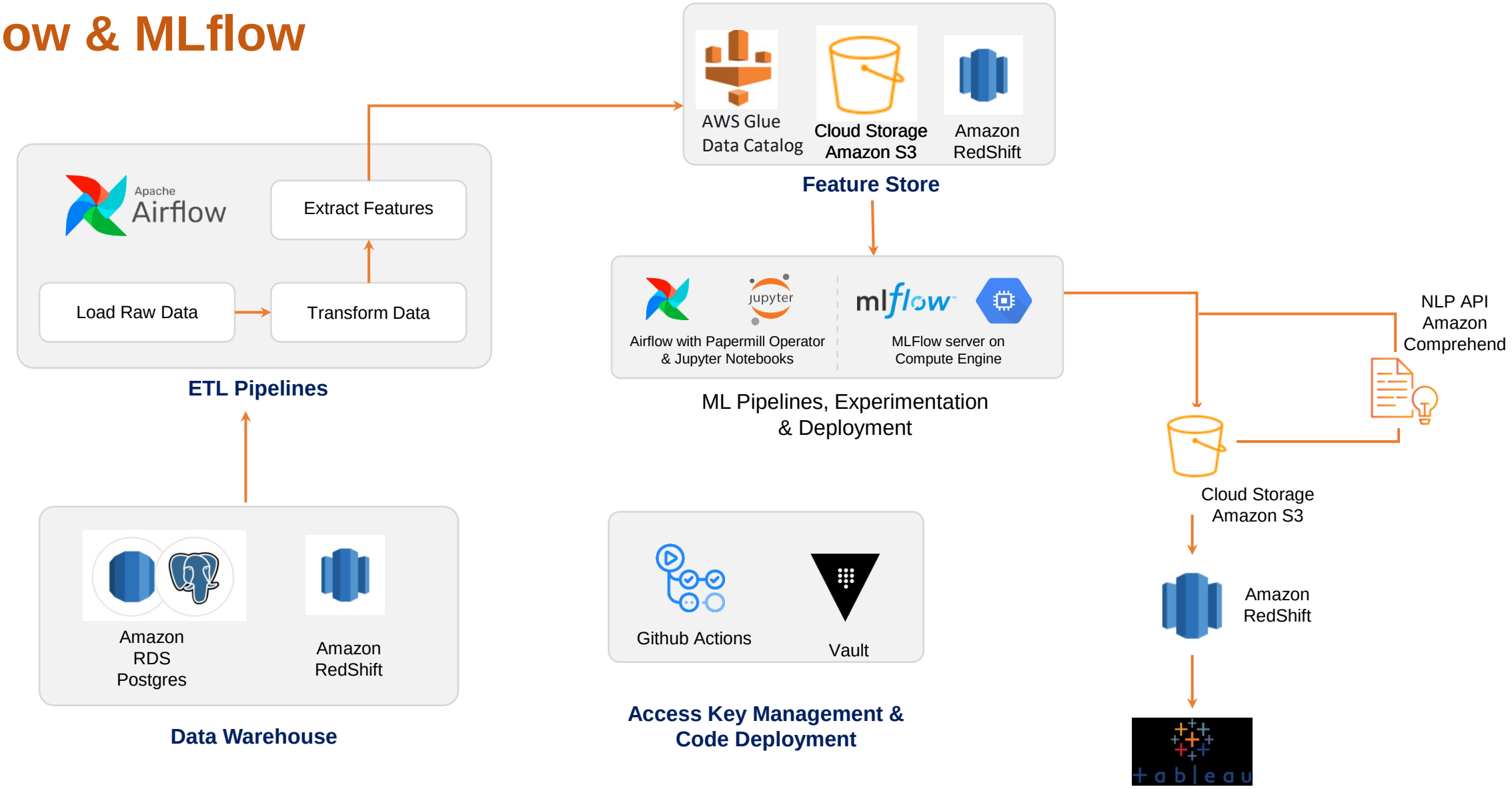


KubeFlow – Deployment Options

- Deployment with Amazon Cognito
- Deployment with Amazon RDS and Amazon S3
- Deployment with Amazon Cognito, Amazon RDS, and Amazon S3
- Vanilla deployment

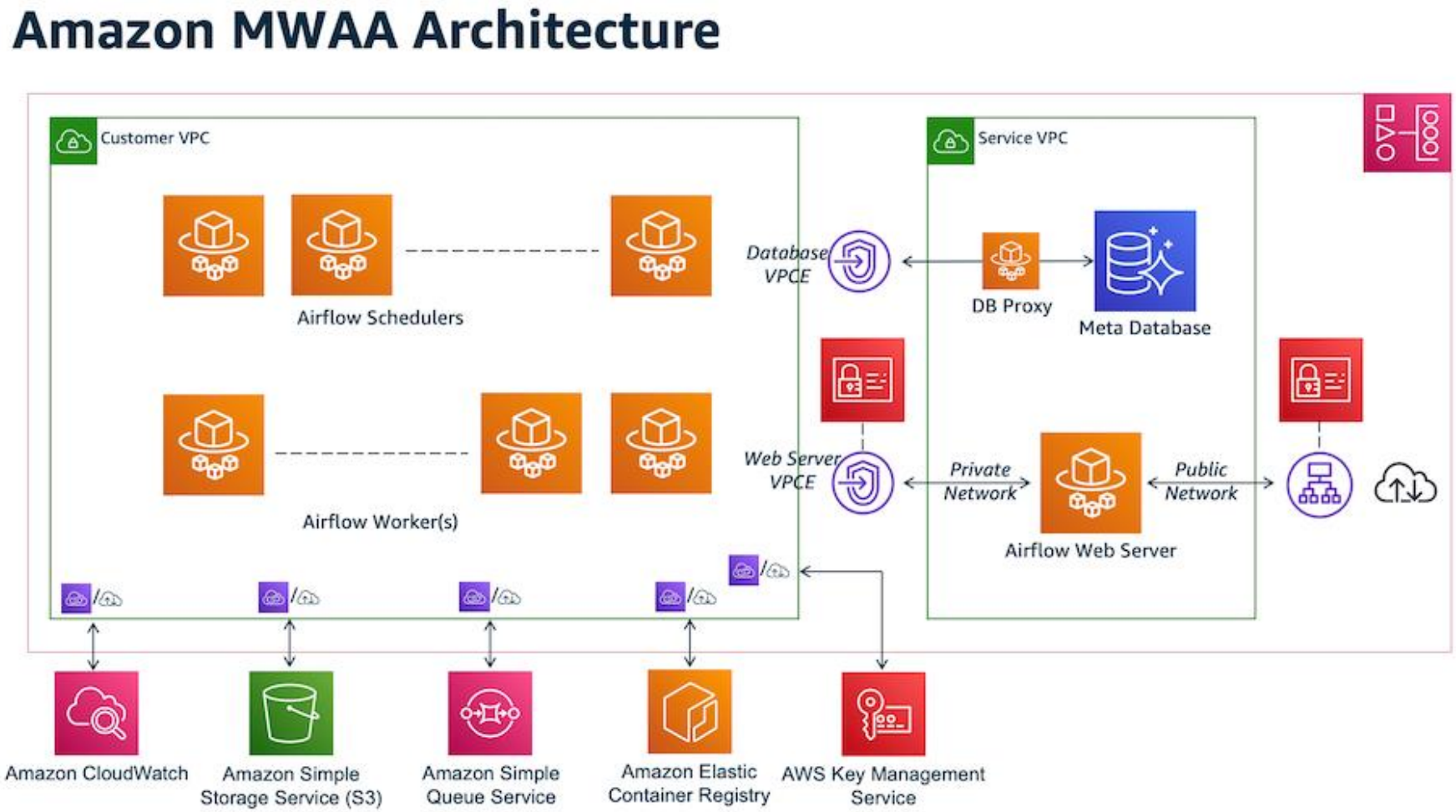
Proposed Architecture – Airflow & MLflow

Option 2

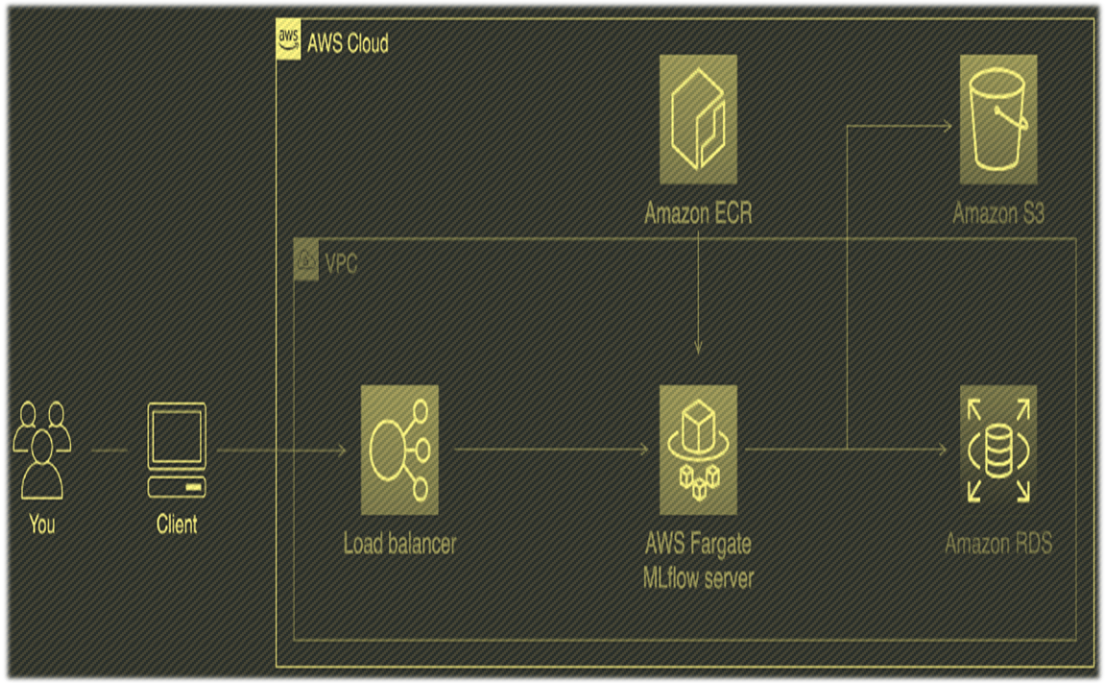


Note: This architecture is based on current analysis of Kubeflow services and may change upon further exploration of Kubeflow

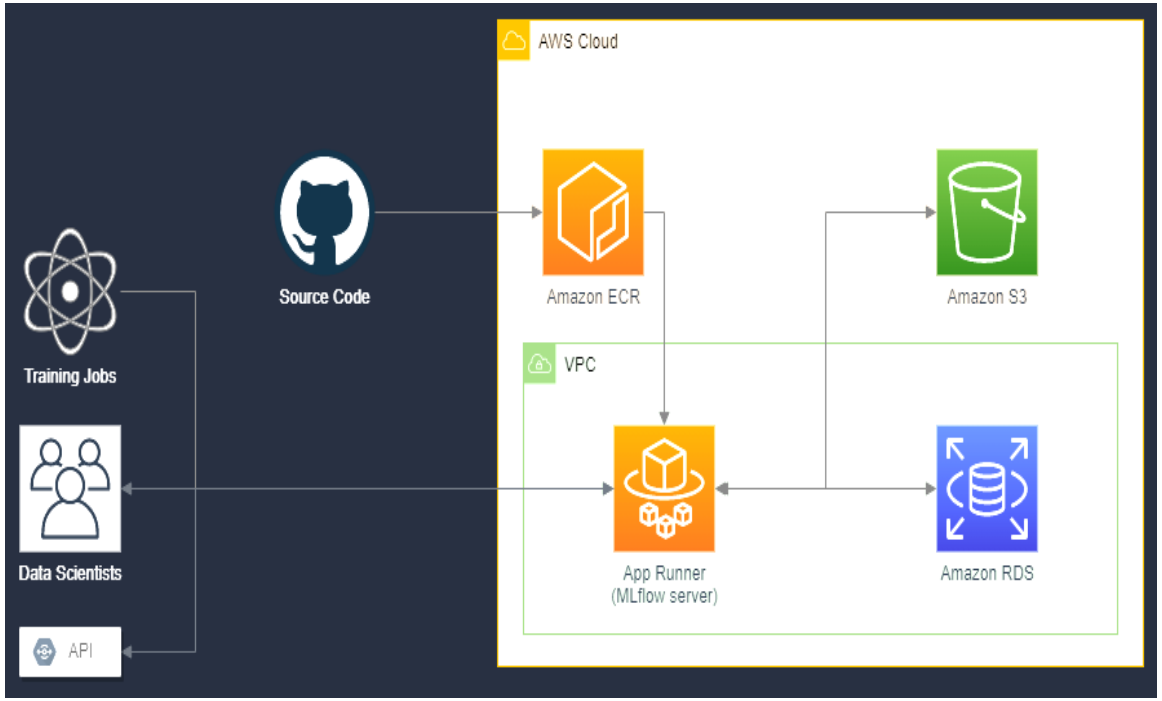
Managed Workflows for Apache Airflow On AWS



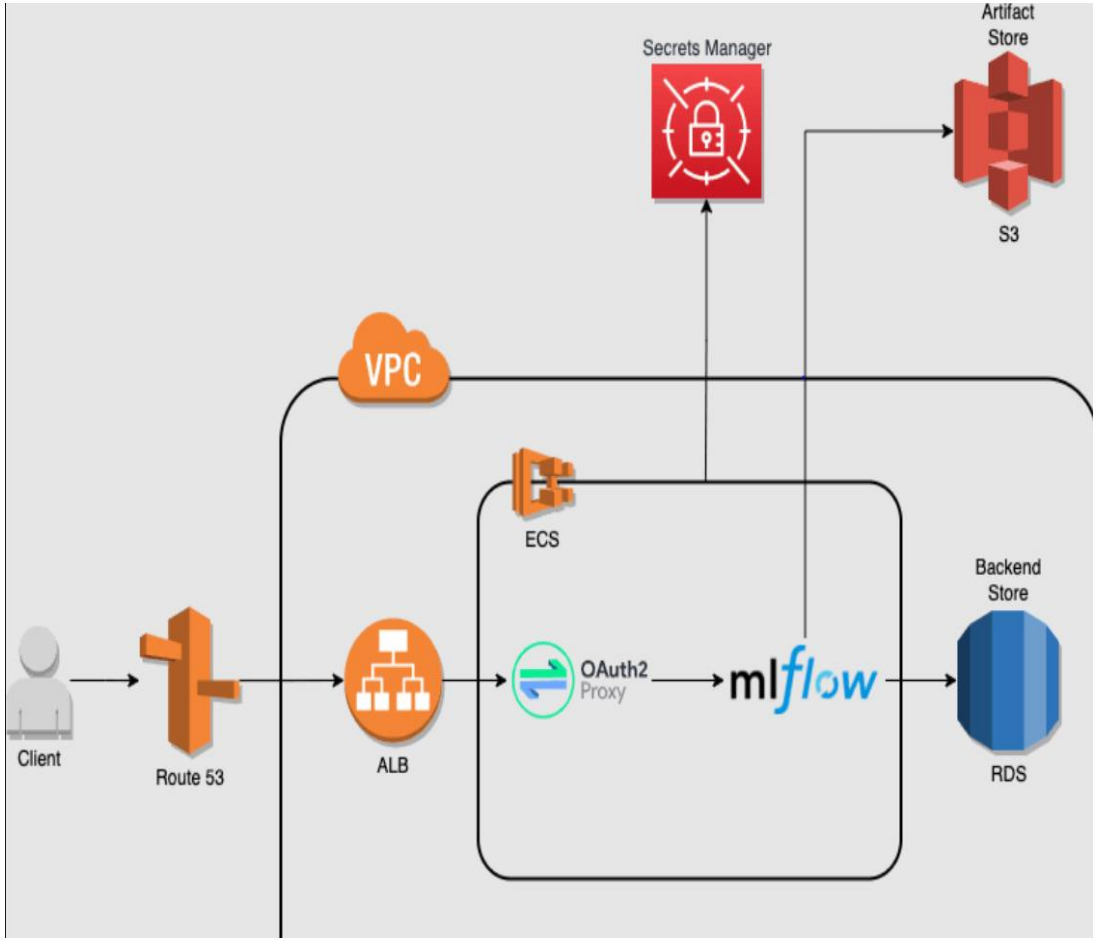
Deploying MLflow on AWS Fargate



Deploying MLflow on App Runner



Deploying secure MLflow on AWS



- The main MLFlow infrastructure components are:

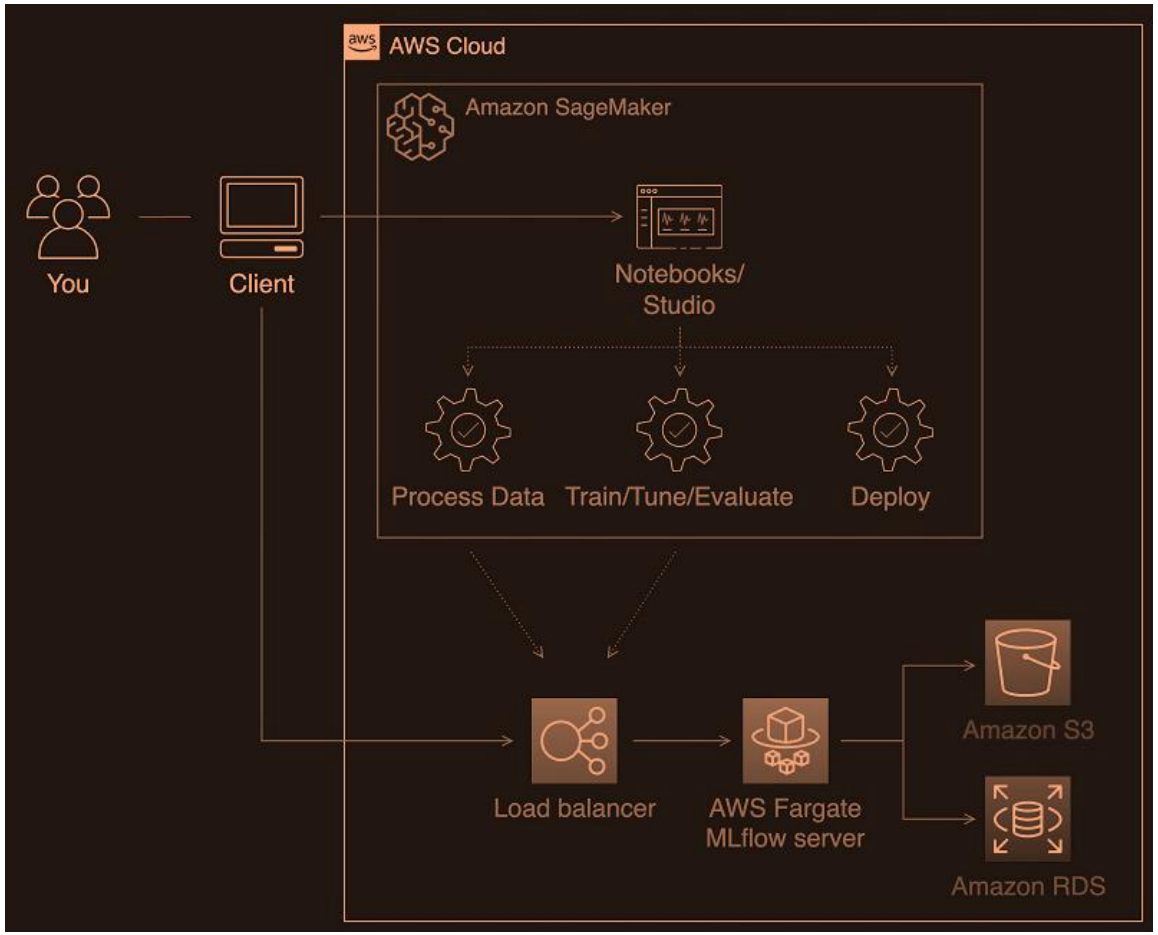
- MLFlow Tracking Server, which exposes API for logging parameters, metrics, experiments, metadata and UI for visualizing the results.
- Amazon Aurora Serverless used as the backend store where MLFlow stores metadata about experiments and runs i.e. metrics, tags and parameters.
- AWS S3 used as the artefact store where MLFlow stores artefacts, e.g. models, data files.
- **Oauth2-proxy** protects MLFlow endpoints using OAuth2 compatible providers, e.g. Google.
- The other AWS components provide a runtime/compute environment (Elastic Container Service, ECS), routing (Application Load Balancer, ALB, and Route 53 as a DNS service) and security (Secrets Manager and Virtual Private Cloud, VPC).

Propose Architecture

Option - 3

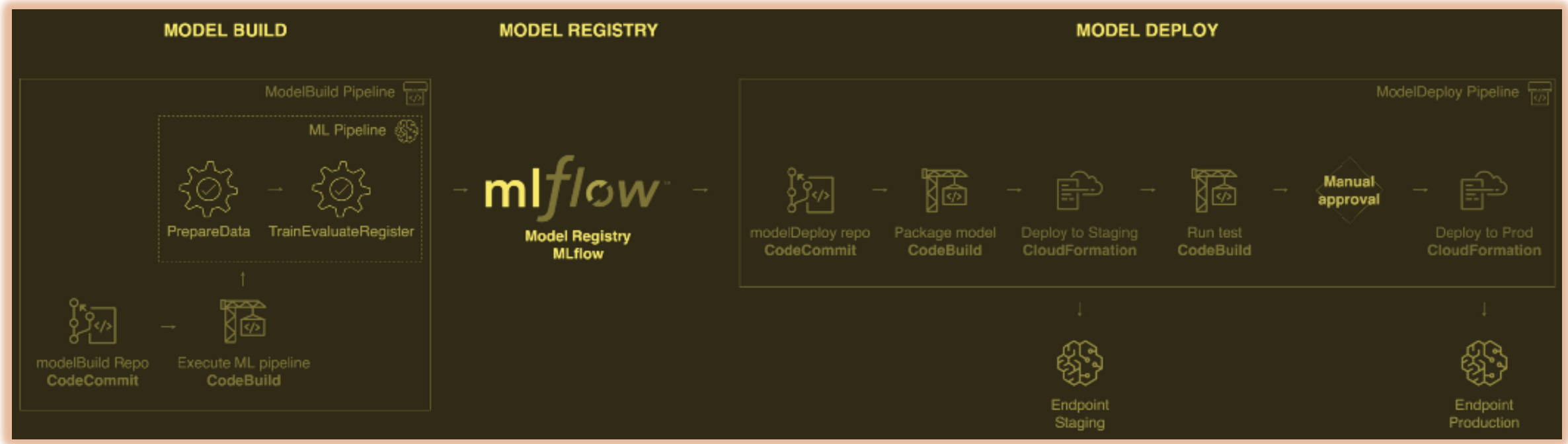
MLOps with MLFlow and Amazon SageMaker Pipelines

- SageMaker Pipelines combines ML workflow orchestration, model registry, and CI/CD into one umbrella so you can quickly get your models into production
- Create an MLOps project for model building, training, and deployment to train an ML model and deploy it into a SageMaker Endpoint.
- Upon updating the modelBuild side of the project it will log models into the MLflow model registry, and the modelDeploy side so it can ship them to production.



AWS MLflow Architecture

- Deploy MLflow on AWS and launch an MLOps project in SageMaker.
- Update the **modelBuild** pipeline so we can log models into our MLflow model registry.
- Deploy the MLflow models into production with the **modelDeploy** pipeline.

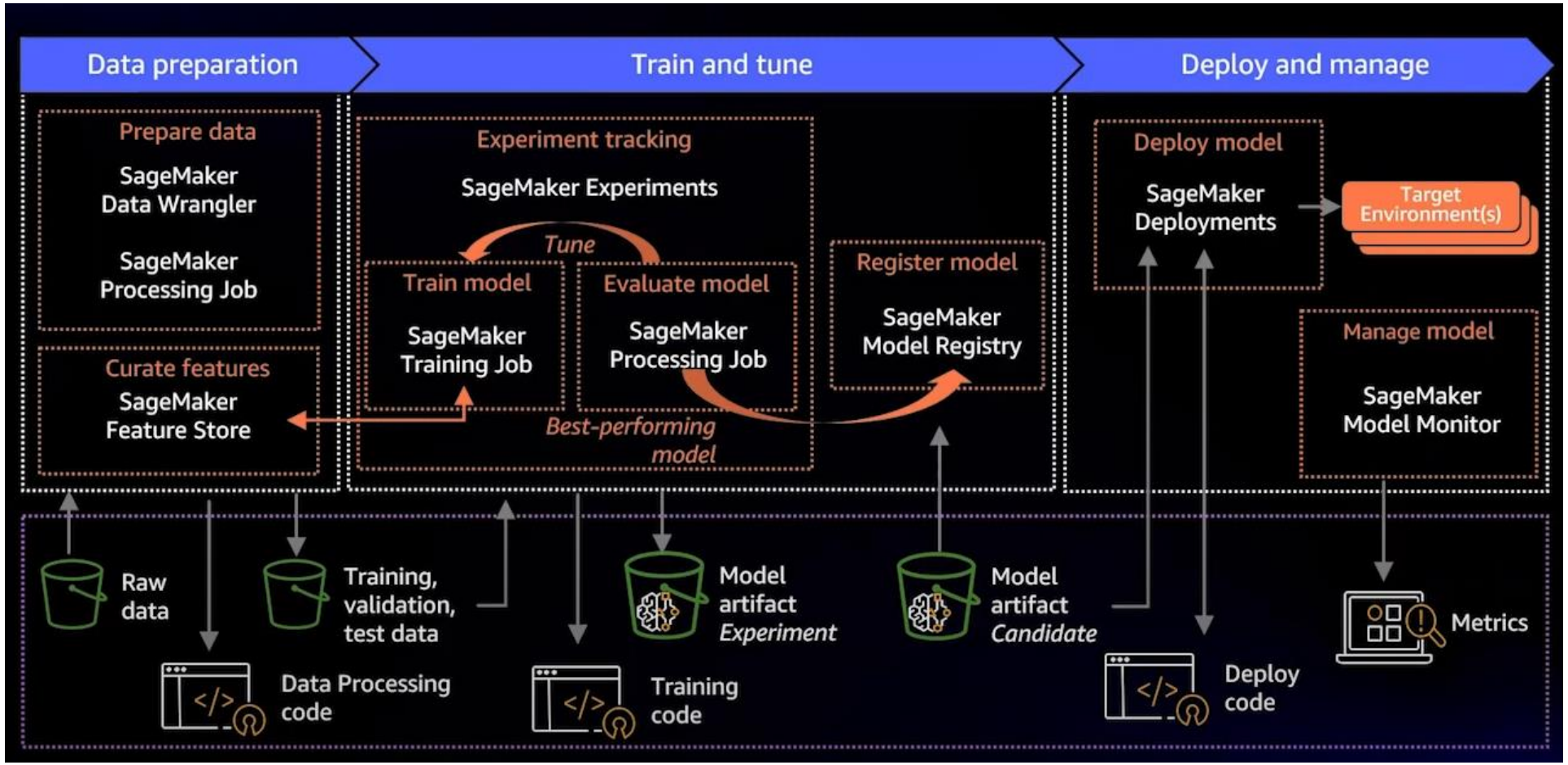


Options – Comparison

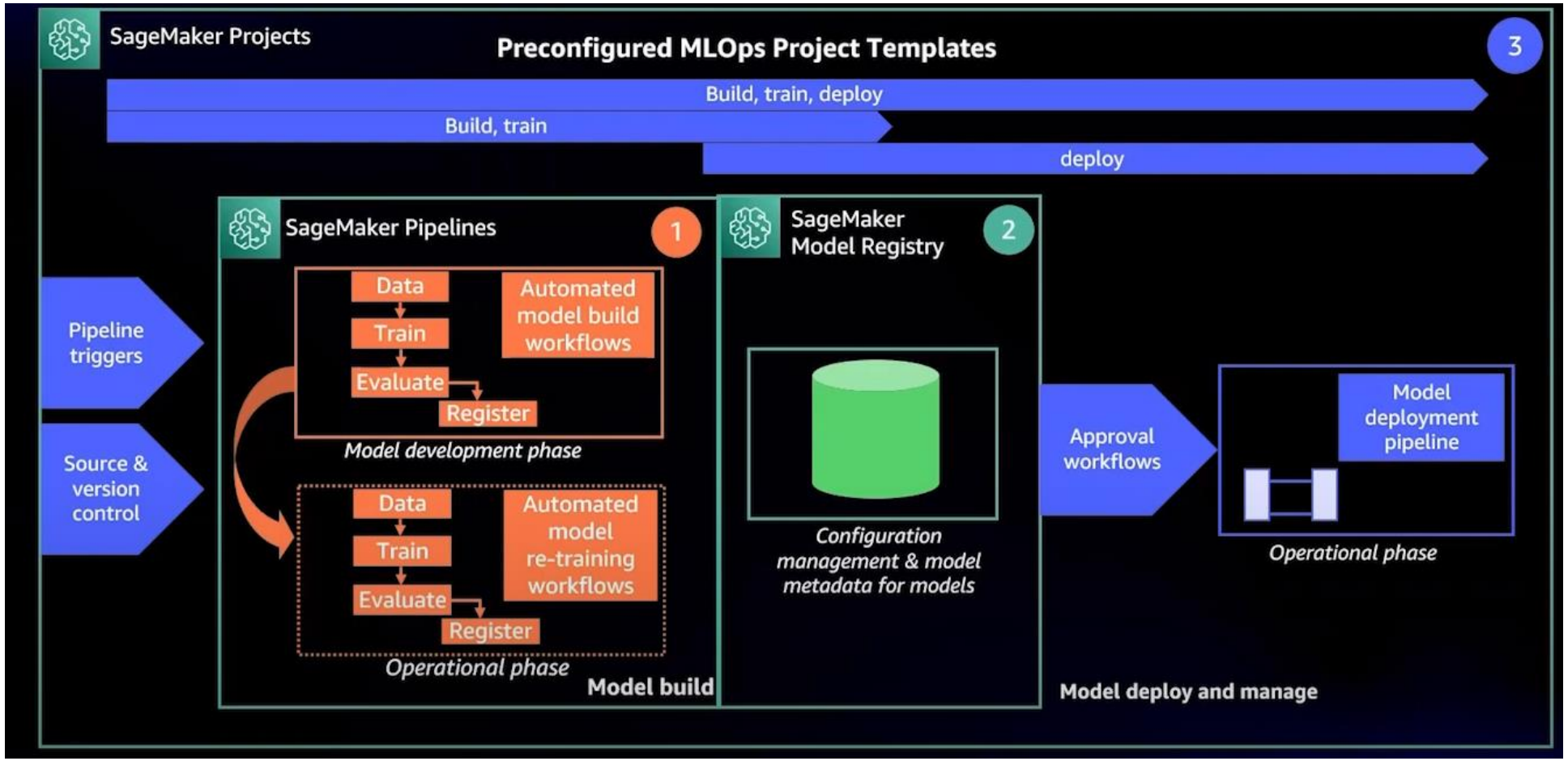
In Progress

MLOPS
AWS SageMaker

Miscellaneous Option



Sage Maker Pipelines : Components



Appendix – A

Set up an Airflow Environment on AWS

- The attached document explains the process steps to configure Airflow as a Managed service.

In Progress

Appendix – B

AWS - MLOPS

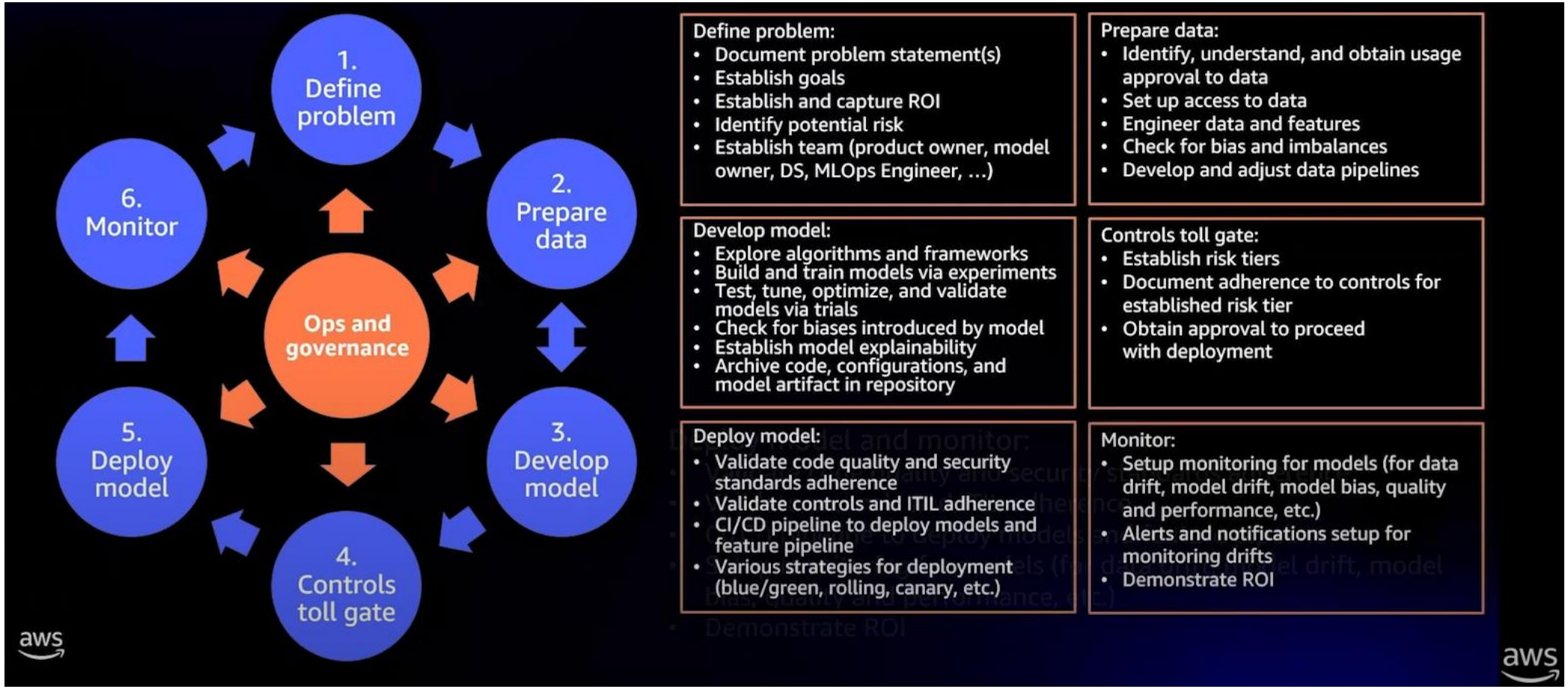
LIFE CYCLE

DEFINE PROBLEM
& PREPARE DATA

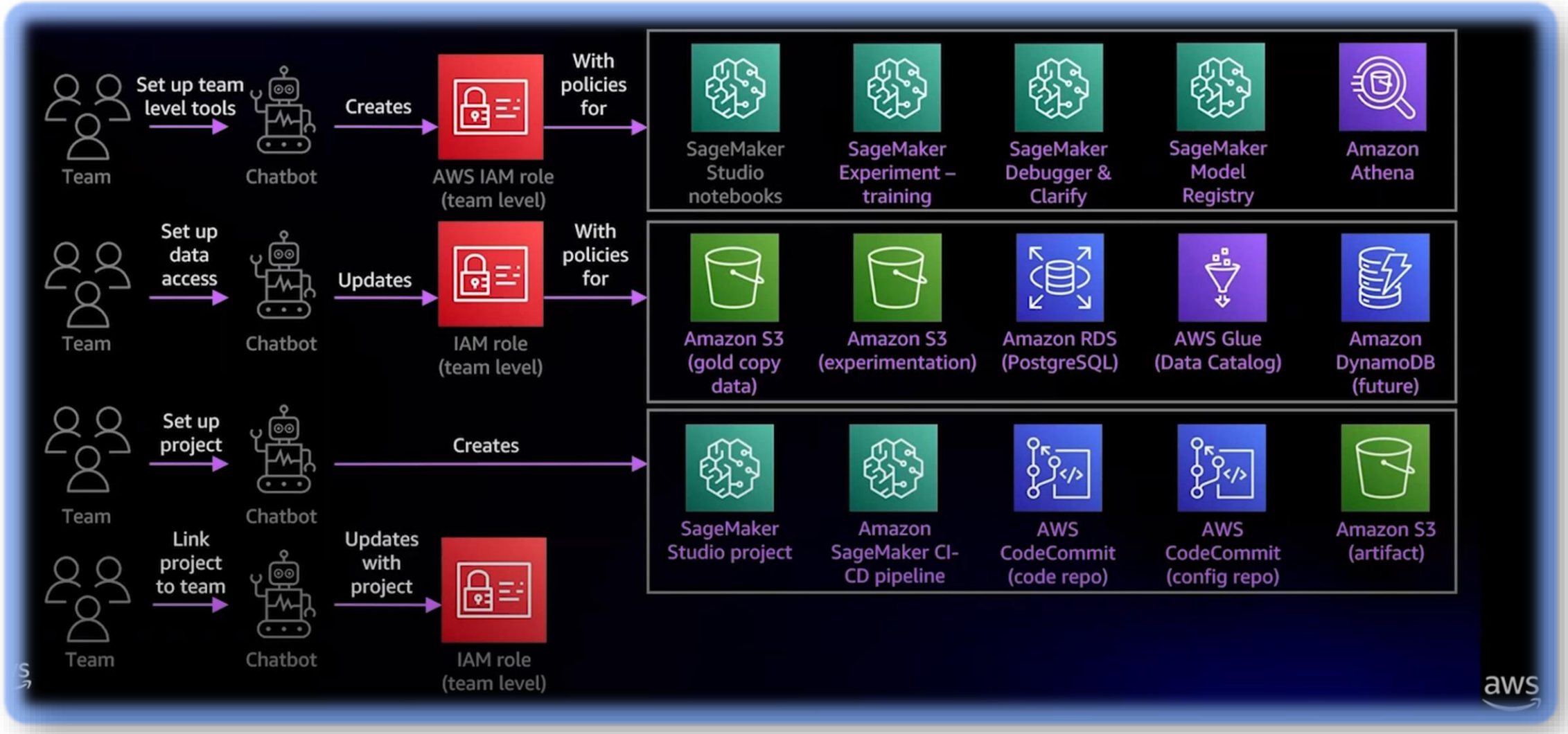
DEVELOP PHASE

CONTROL , DEPLOY
& MONITOR

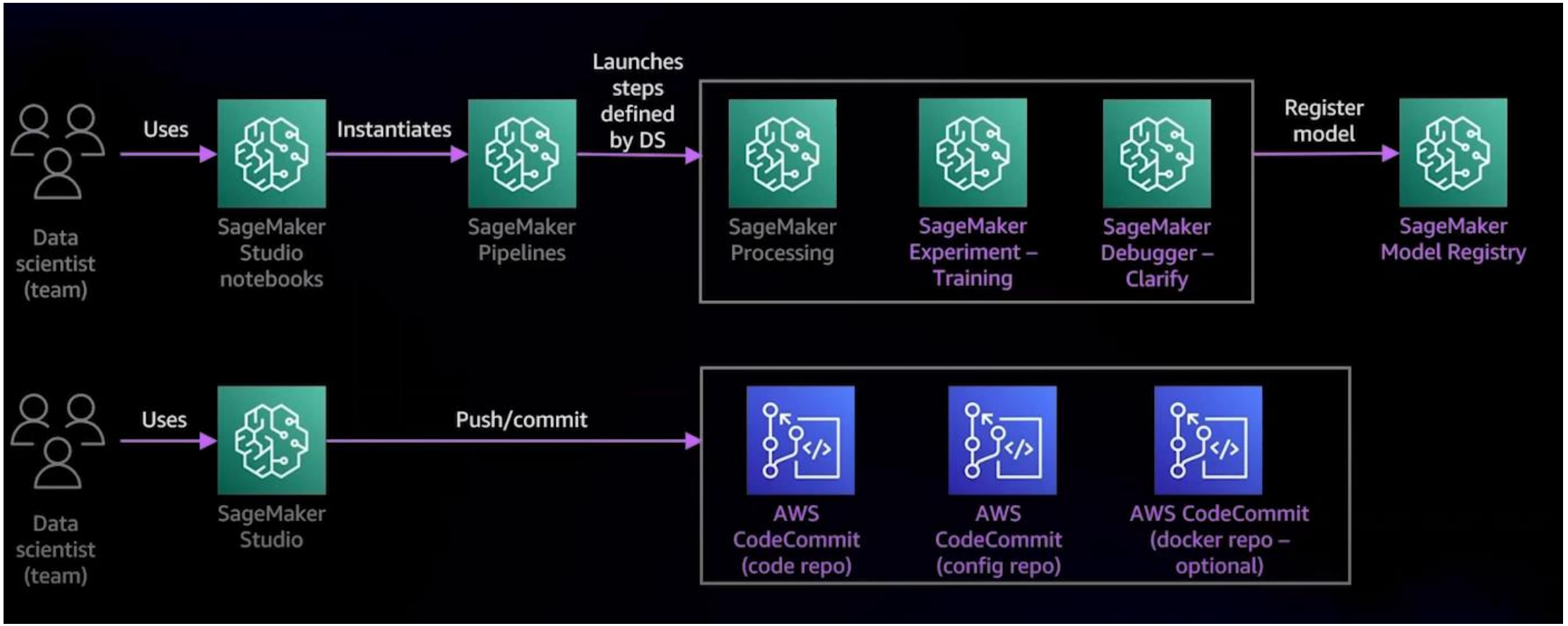
MLOPS - Lifecycle



MLOPS – Define Problem & Prepare Data



MLOPS : Develop Phase



MLOPS : Control, Deploy, & Monitor Phase