AWS MLOps Framework (Single account)

This solution is an extendable framework that provides a standard interface for managing ML pipelines for AWS ML services and third-party services. To deploy this solution using the available AWS CloudFormation template, select **Deploy with AWS**.
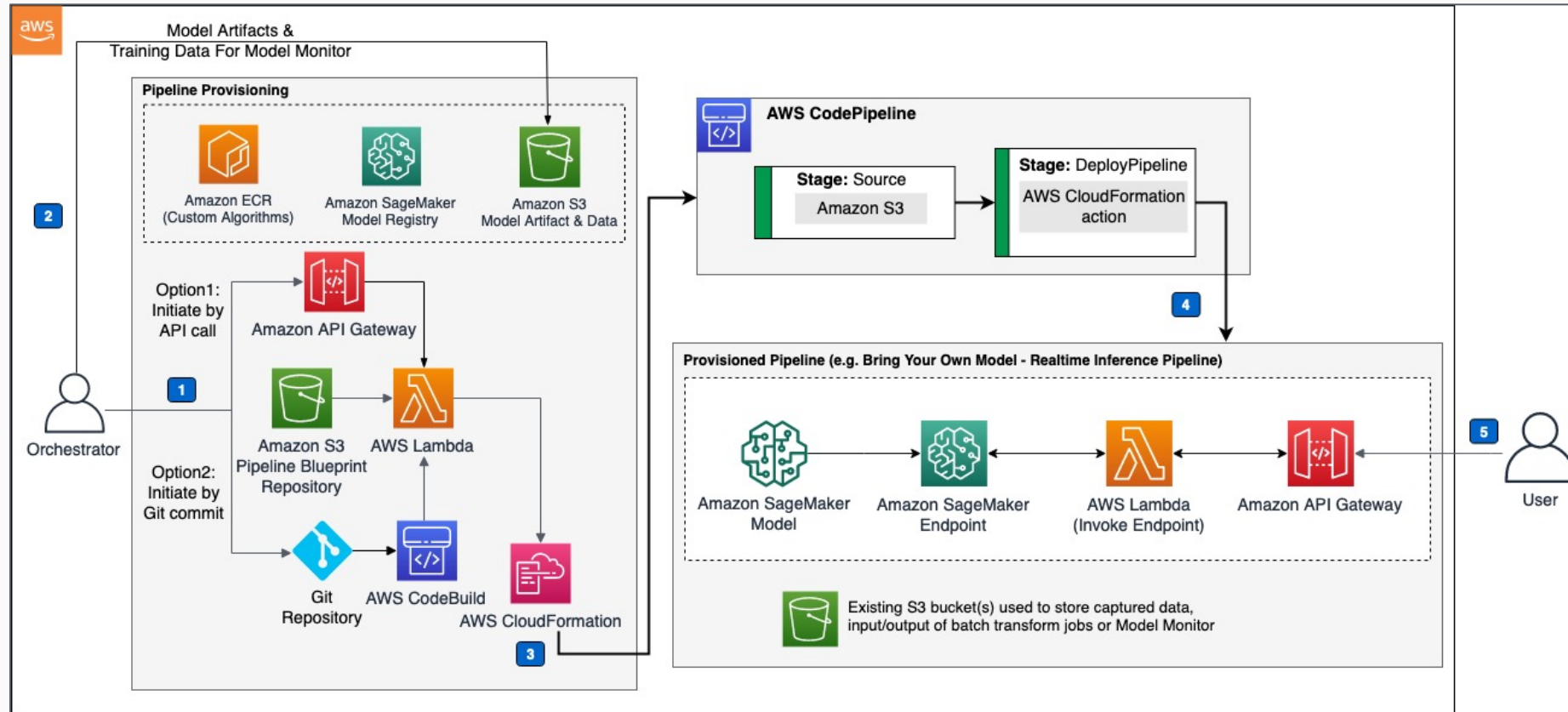

1 The Orchestrator (solution owner or DevOps engineer) launches the solution in the AWS account and selects the desired options (for example, using Amazon SageMaker Registry, or providing an existing S3 bucket).

2 The Orchestrator uploads the required assets for the target pipeline (for example, model artifact, training data, and/or custom algorithm zip file) into the Assets S3 bucket. If Amazon SageMaker Model Registry is used, the Orchestrator (or an automated pipeline) must register the model with the Model Registry.

3 A single account AWS CodePipeline instance is provisioned by either sending an API call to the API Gateway, or by committing the mlops-config.json file to the Git repository. Depending on the pipeline type, the Orchestrator AWS Lambda function packages the target AWS CloudFormation template and its parameters/configurations using the body of the API call or the mlops-config.json file, and uses it as the source stage for the AWS CodePipeline instance.

4 The DeployPipeline stage takes the packaged CloudFormation template and its parameters/configurations and deploys the target pipeline into the same account.

5 After the target pipeline is provisioned, users can access its functionalities. An Amazon Simple Notification Service (Amazon SNS) notification is sent to the email provided in the solution's launch parameters.

Deploy with AWS

Deployable AWS Reference Implementation

AWS MLOps Framework (Multi-account)

This solution is an extendable framework that provides a standard interface for managing ML pipelines for AWS ML services and third-party services. To deploy this solution using the available AWS CloudFormation template, select **Deploy with AWS**.
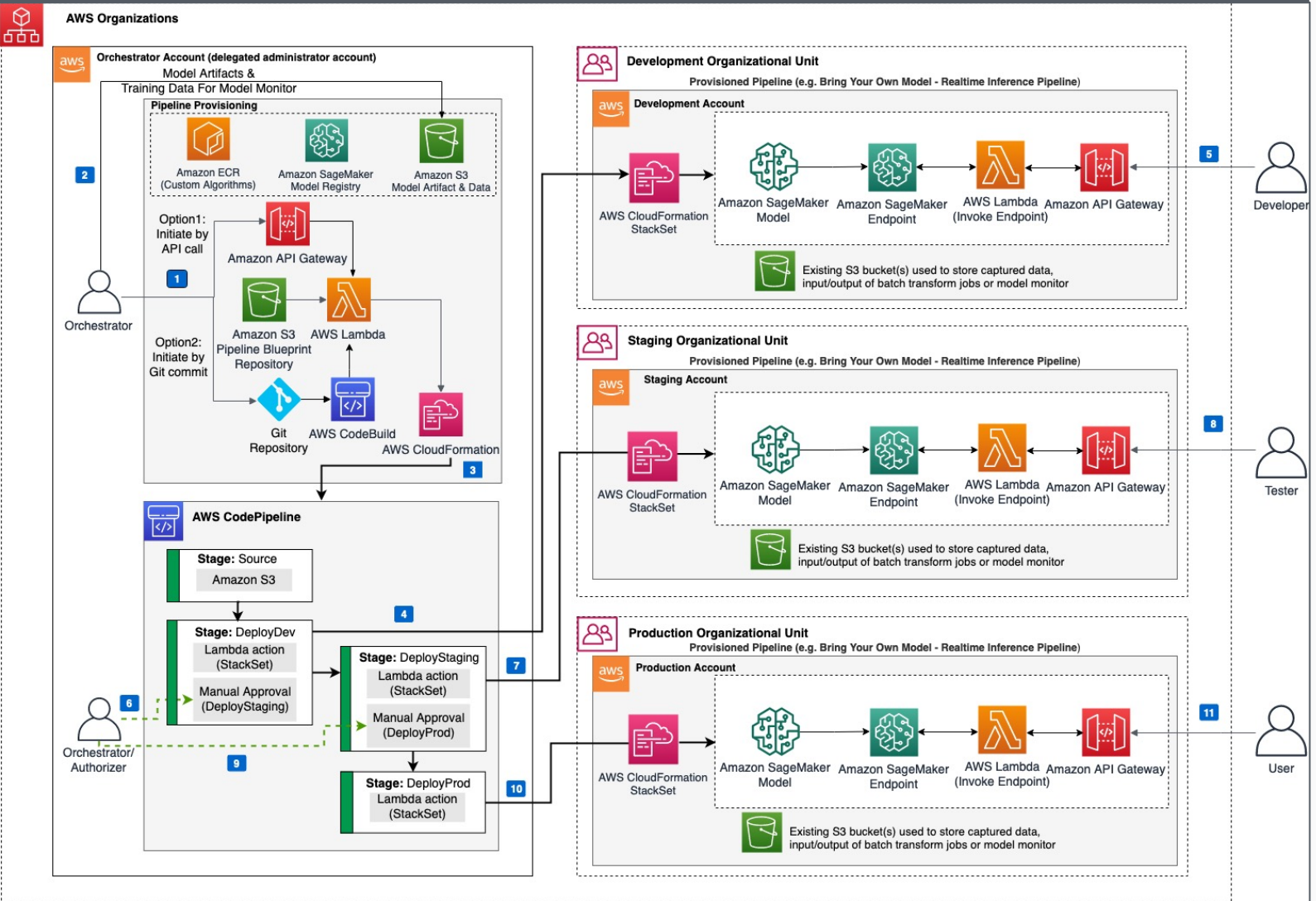
1. The Orchestrator (solution owner or DevOps engineer with admin access to the Orchestrator account) provides the AWS Organizations information (for example, development, staging, and production organizational unit IDs and account numbers). They also specify the desired options (for example, using Amazon SageMaker Registry, or providing an existing S3 bucket), and then launch the solution in their AWS account.

2. The Orchestrator uploads the required assets for the target pipeline (for example, model artifact, training data, and/or custom algorithm zip file) into the Assets S3 bucket in the AWS Orchestrator account. If Amazon SageMaker Model Registry is used, the Orchestrator (or an automated pipeline) must register the model with the Model Registry.

3. A multi-account AWS CodePipeline instance is provisioned by either sending an API call to the API Gateway, or by committing the mlops-config.json file to the Git repository. Depending on the pipeline type, the Orchestrator AWS Lambda function packages the target AWS CloudFormation template and its parameters/configurations for each stage using the body of the API call or the mlops-config.json file, and uses it as the source stage for the AWS CodePipeline instance.

4. The DeployDev stage takes the packaged CloudFormation template and its parameters/configurations and deploys the target pipeline into the development account.

5. After the target pipeline is provisioned into the development account, the developer can then iterate on the pipeline.

6. After the development is finished, the Orchestrator (or another authorized account) manually approves the DeployStaging action to move to the DeployStaging Stage.

7. The DeployStaging stage deploys the target pipeline into the staging account, using the staging configuration.

8. Testers perform different tests on the deployed pipeline.

9. After the pipeline passes quality tests, the Orchestrator can approve the DeployProd action.

10. The DeployProd stage deploys the target pipeline (with production configurations) into the production account.

11. Finally, the target pipeline is live in production. An Amazon SNS notification is sent to the email provided in the solution's launch parameters.

Reviewed for technical accuracy September 27, 2021

Deployable AWS Reference Implementation

Deploy with AWS