

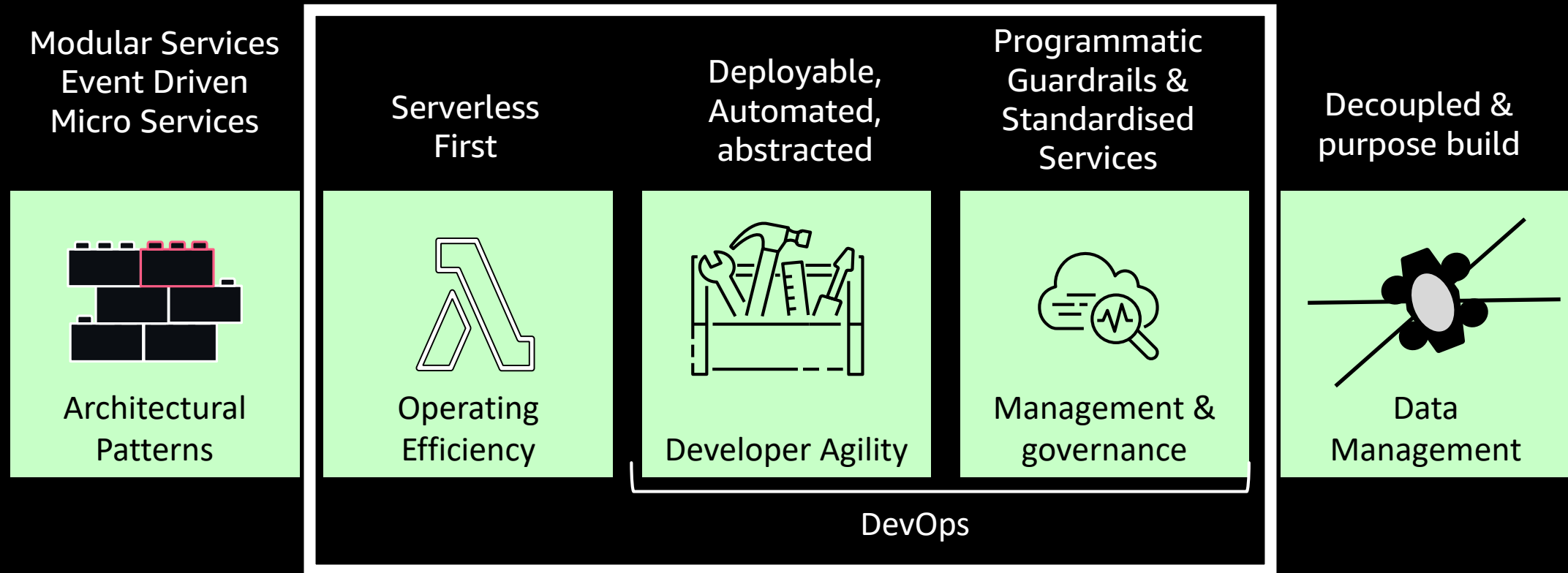
Streamline your Cloud Native Application Deployment using AWS SAM

Urmila Raju (She/Her)

Senior Solutions Architect, AWS



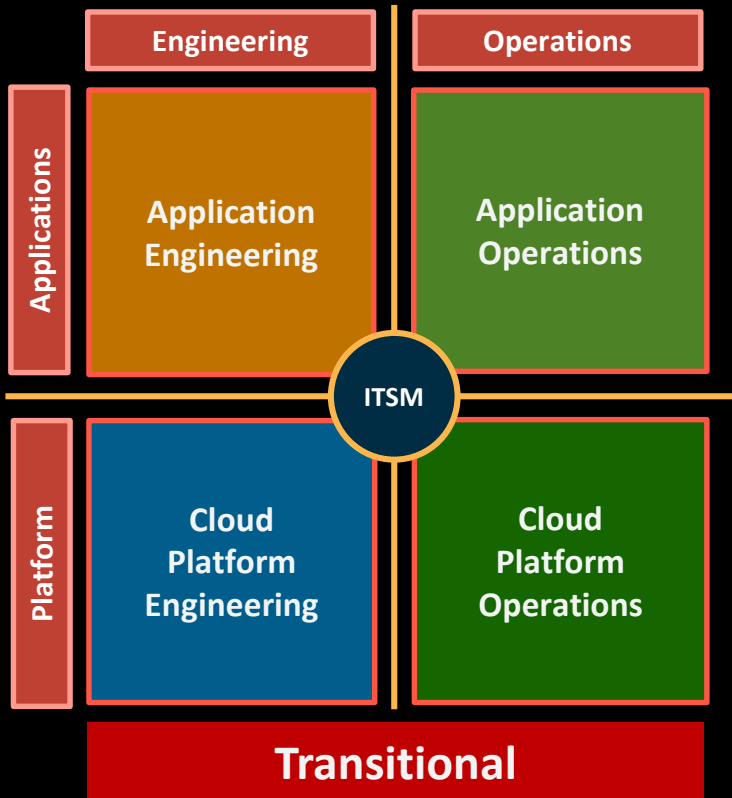
What are modern/ Cloud Native apps...



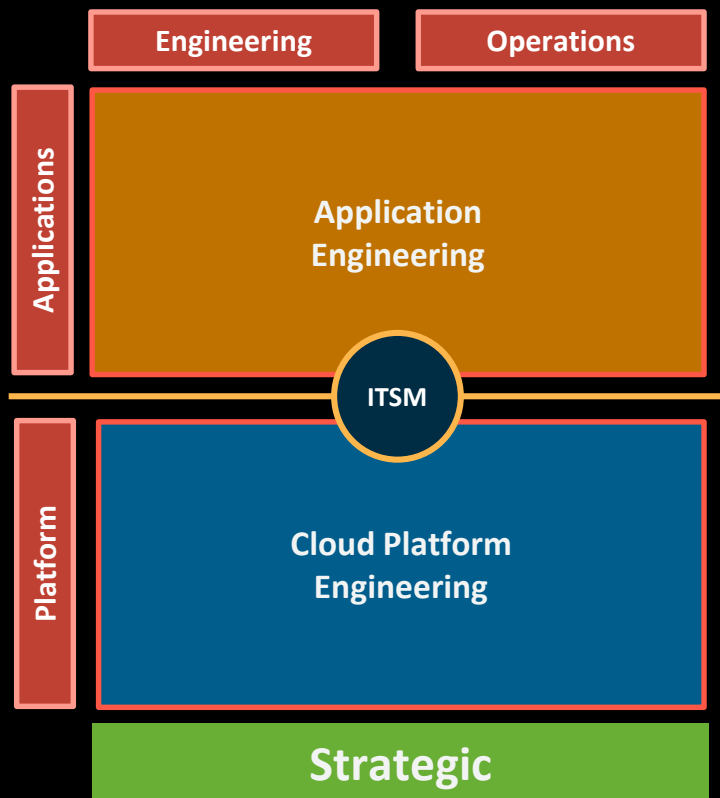
How can we build and
deploy Serverless Apps with
Speed?

Is your Cloud Operating model ready for Modern Apps

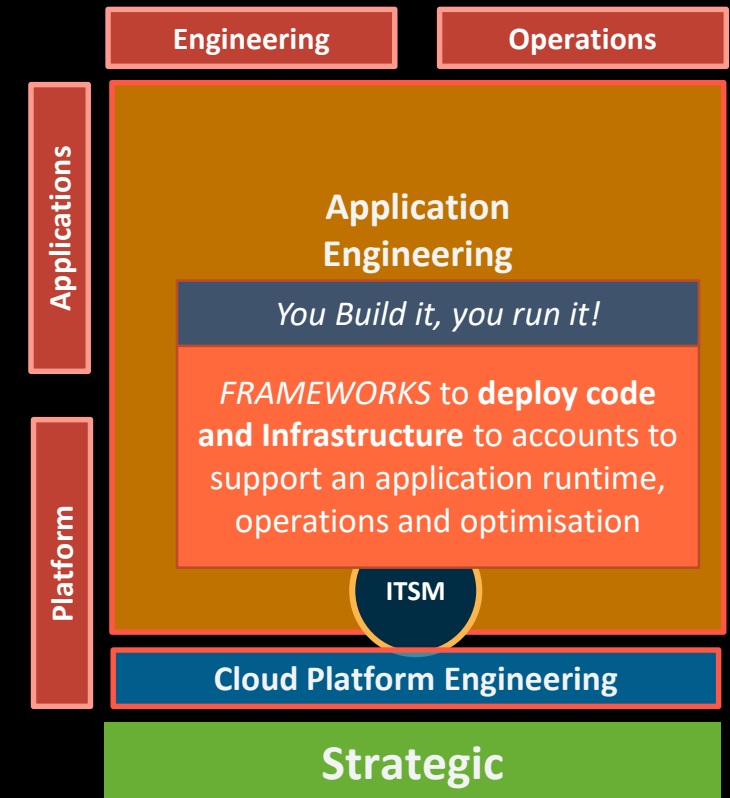
“Traditional Operations”



“Distributed DevOps”



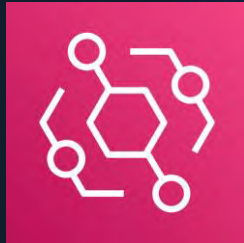
“Decentralized DevOps”



Serverless Apps - Code and Infrastructure are tightly coupled



AWS
Lambda



Amazon
EventBridge



AWS Step
functions

So, how do we choose the right
'Infrastructure as Code' tool?

IaC tools – Which is right for you?

- AWS Native



AWS CloudFormation



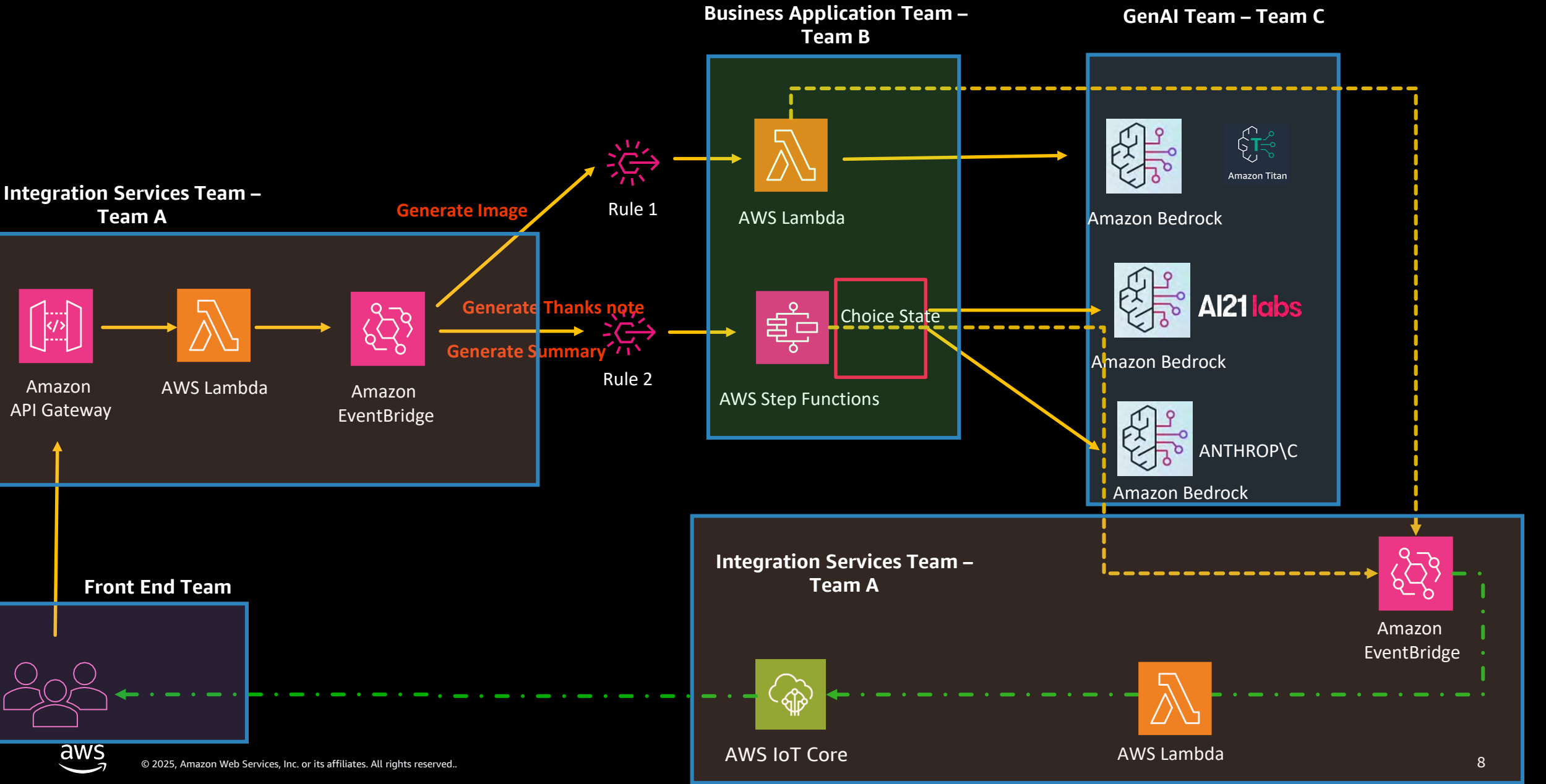
AWS Cloud
Development Kit (AWS
CDK)

- Open source

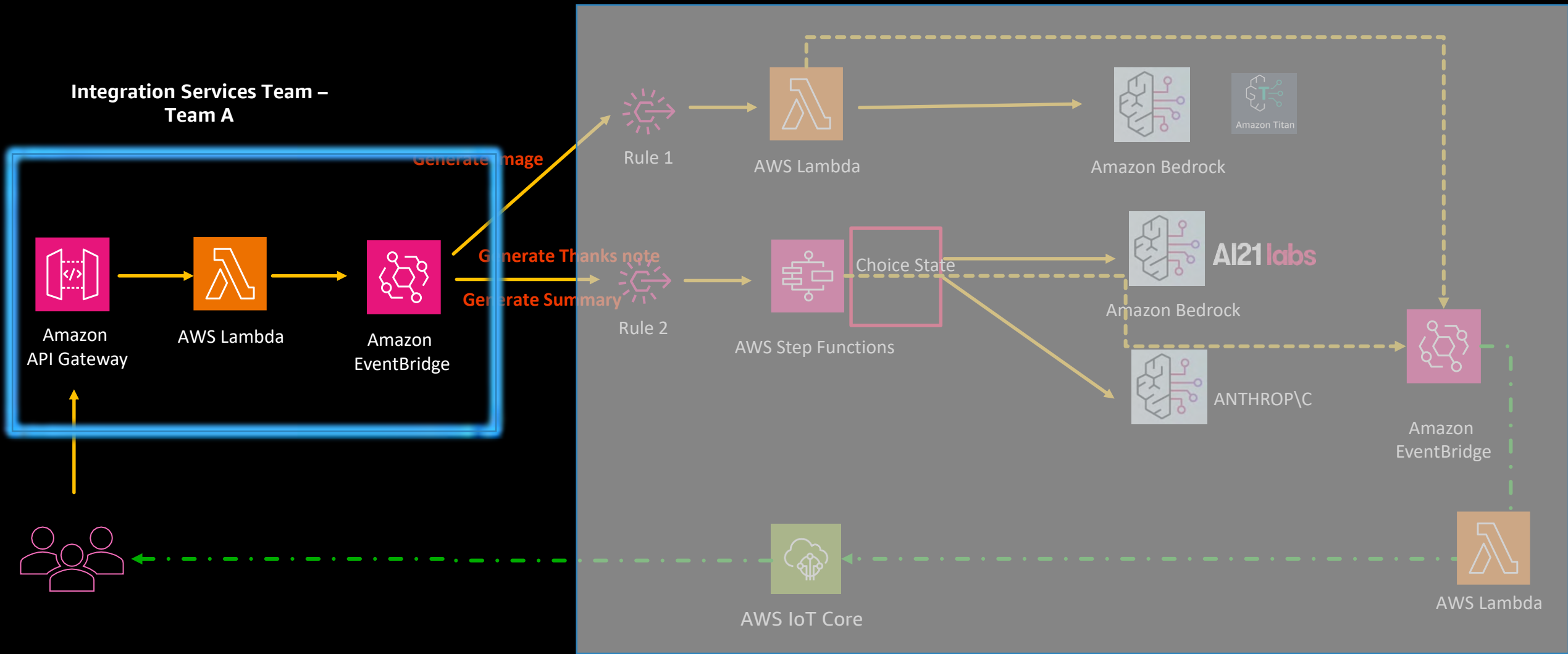


A practical guide to AWS SAM using a Serverless GenAI app example

Serverless GenAI App – Text Summary and Image Generation



Lets start using AWS SAM....



AWS SAM has two parts

Transform templates

Shorthand syntax expresses resources and event source mappings; it provides infrastructure as code (IaC) for serverless applications

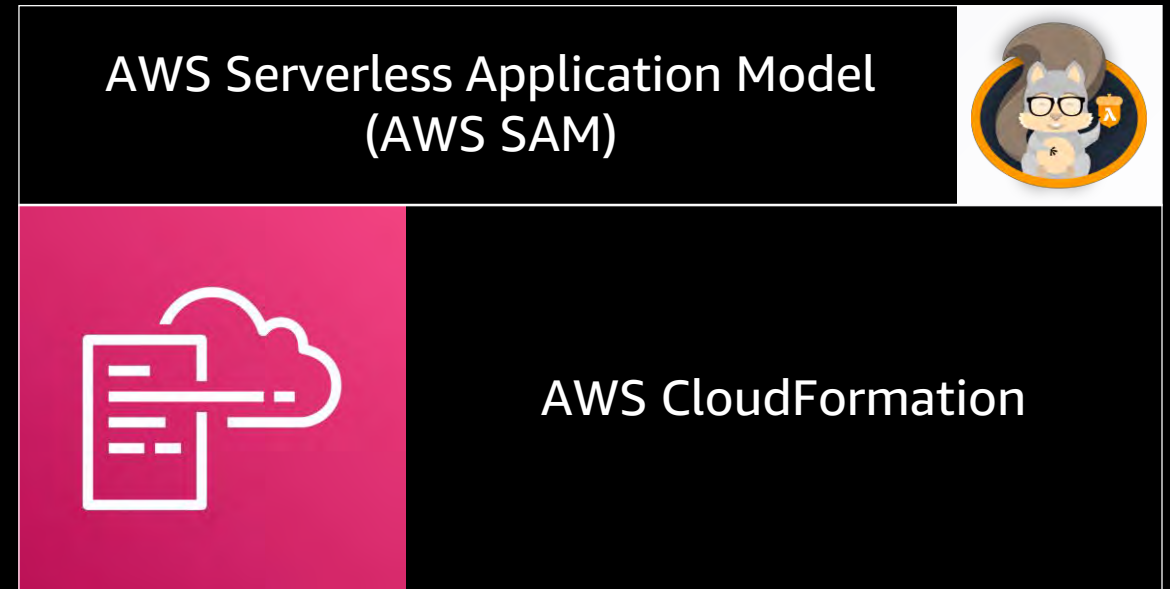
AWS Command Line Interface (AWS CLI)

Provides tooling for local development with rapid iteration, debugging, build, packaging, and deployment for serverless applications



AWS SAM transform templates

Built on top of
AWS CloudFormation



AWS SAM transform templates

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetProductsFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: src/
      Handler: app.handler
      Runtime: python3.9
      Policies:
        - DynamoDBReadPolicy:
            TableName: !Ref ProductTable
    Events:
      GetProductsEvent:
        Type: Api
        Properties:
          Path: /products
          Method: get
  ProductTable:
    Type: AWS::Serverless::SimpleTable
```



AWS Lambda function



AWS IAM role

AWS SAM serverless resources

9 serverless
resource
types

- AWS::Serverless::Api
- AWS::Serverless::Application
- AWS::Serverless::Connector
- AWS::Serverless::Function
- AWS::Serverless::GraphQLApi
- AWS::Serverless::HttpApi
- AWS::Serverless::LayerVersion
- AWS::Serverless::SimpleTable
- AWS::Serverless::StateMachine
- SAM Version 2016-10-31

AWS SAM transform templates

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetProductsFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: src/
      Handler: app.handler
      Runtime: python3.9
      Policies:
        - DynamoDBReadPolicy:
            TableName: !Ref ProductTable
    Events:
      GetProductsEvent:
        Type: Api
        Properties:
          Path: /products
          Method: get
  ProductTable:
    Type: AWS::Serverless::SimpleTable
```



AWS IAM role

AWS SAM serverless resources – IAM policies

75+

managed templates,
with more being added
s12d.com/sam-policies

Policy template table

The following is a table of the available policy templates.

Policy Template	Description
AcmGetCertificatePolicy	Gives a permission to read a certificate from AWS Certificate Manager.
AMIDescribePolicy	Gives permission to describe Amazon Machine Images (AMIs).
AthenaQueryPolicy	Gives permissions to execute Athena queries.
AWSSecretsManagerGetSecretValuePolicy	Gives permission to get the secret value for the specified AWS Secrets Manager secret.
AWSSecretsManagerRotationPolicy	Gives permission to rotate a secret in AWS Secrets Manager.
CloudFormationDescribeStacksPolicy	Gives permission to describe AWS CloudFormation stacks.
CloudWatchDashboardPolicy	Gives permissions to put metrics to operate on CloudWatch dashboards.
CloudWatchDescribeAlarmHistoryPolicy	Gives permission to describe CloudWatch alarm history.
CloudWatchPutMetricPolicy	Gives permission to send metrics to CloudWatch.
CodeCommitCrudPolicy	Gives permissions to create/read/update/delete objects within a specific CodeCommit repository.
CodeCommitReadPolicy	Gives permissions to read objects within a specific CodeCommit repository.
CodePipelineLambdaExecutionPolicy	Gives permission for a Lambda function invoked by CodePipeline to report the status of the job.
CodePipelineReadOnlyPolicy	Gives read permission to get details about a CodePipeline pipeline.

AWS SAM transform templates

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetProductsFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: src/
      Handler: app.handler
      Runtime: python3.9
      Policies:
        - DynamoDBReadPolicy:
            TableName: !Ref ProductTable
    Events:
      GetProductsEvent:
        Type: Api
        Properties:
          Path: /products
          Method: get
  ProductTable:
    Type: AWS::Serverless::SimpleTable
```



AWS Lambda function



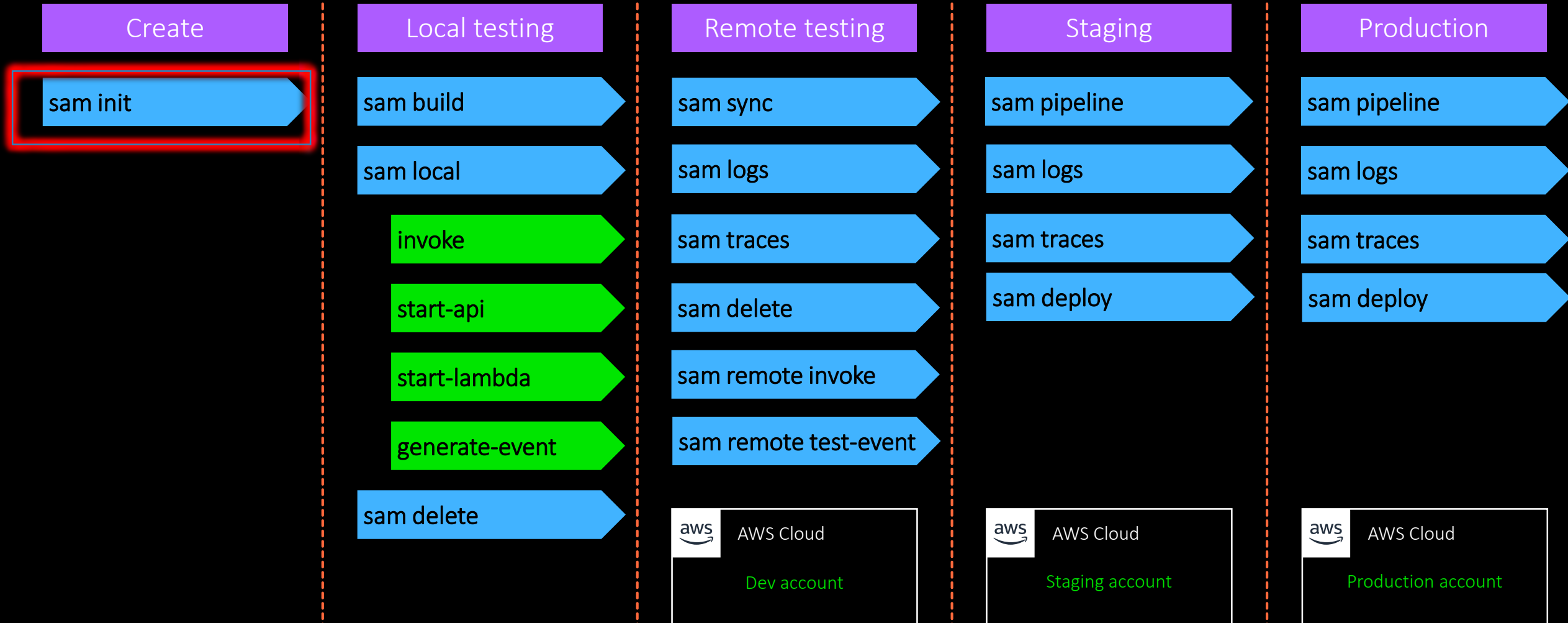
Amazon API Gateway

Lambda function event sources

19
function event
source types
supported in
AWS SAM

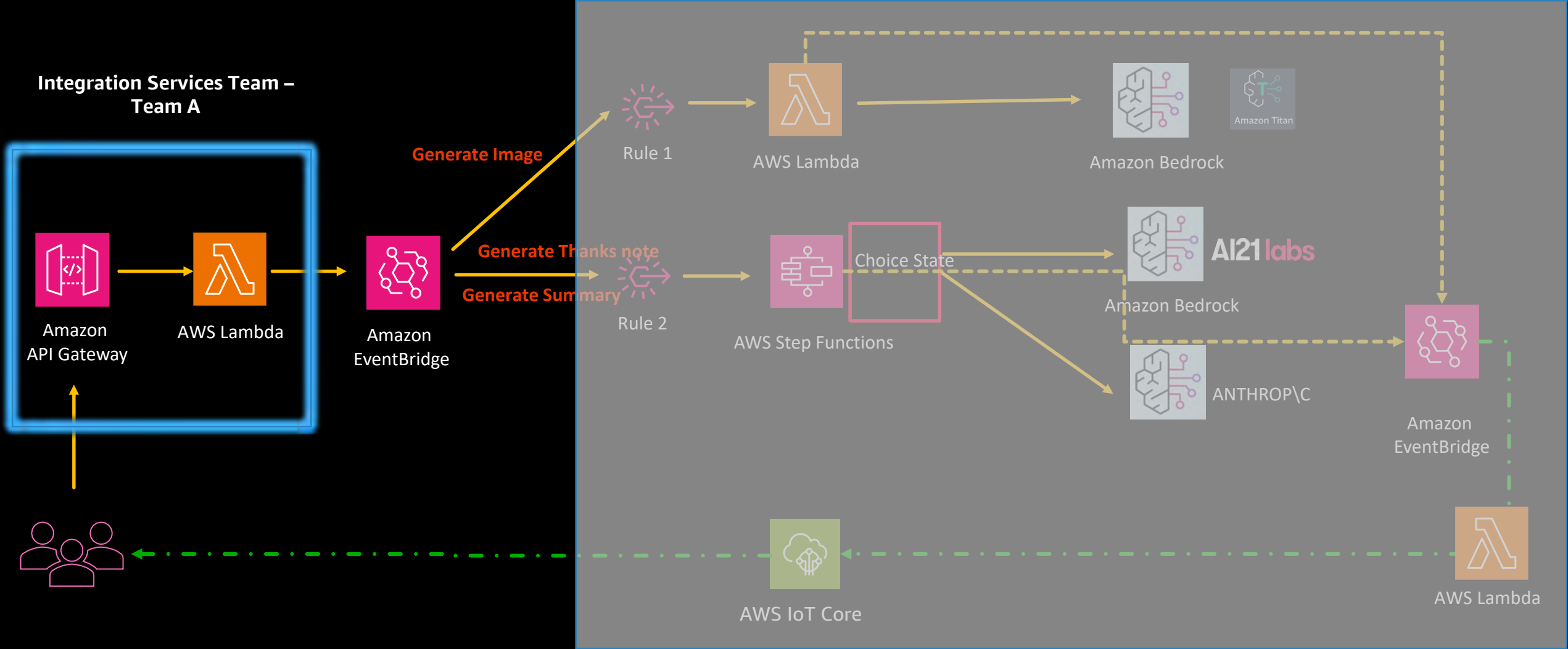
- Amazon S3
- Amazon SNS
- Amazon Kinesis
- Amazon DynamoDB
- Amazon SQS
- Api
- HttpApi
- Schedule
- ScheduleV2
- CloudWatchEvent
- CloudWatchLogs
- IoTRule
- Alexa skills
- Amazon Cognito
- EventBridgeRule
- Amazon MSK
- SelfManagedKafka
- Amazon MQ
- Amazon DocumentDB

The development cycle

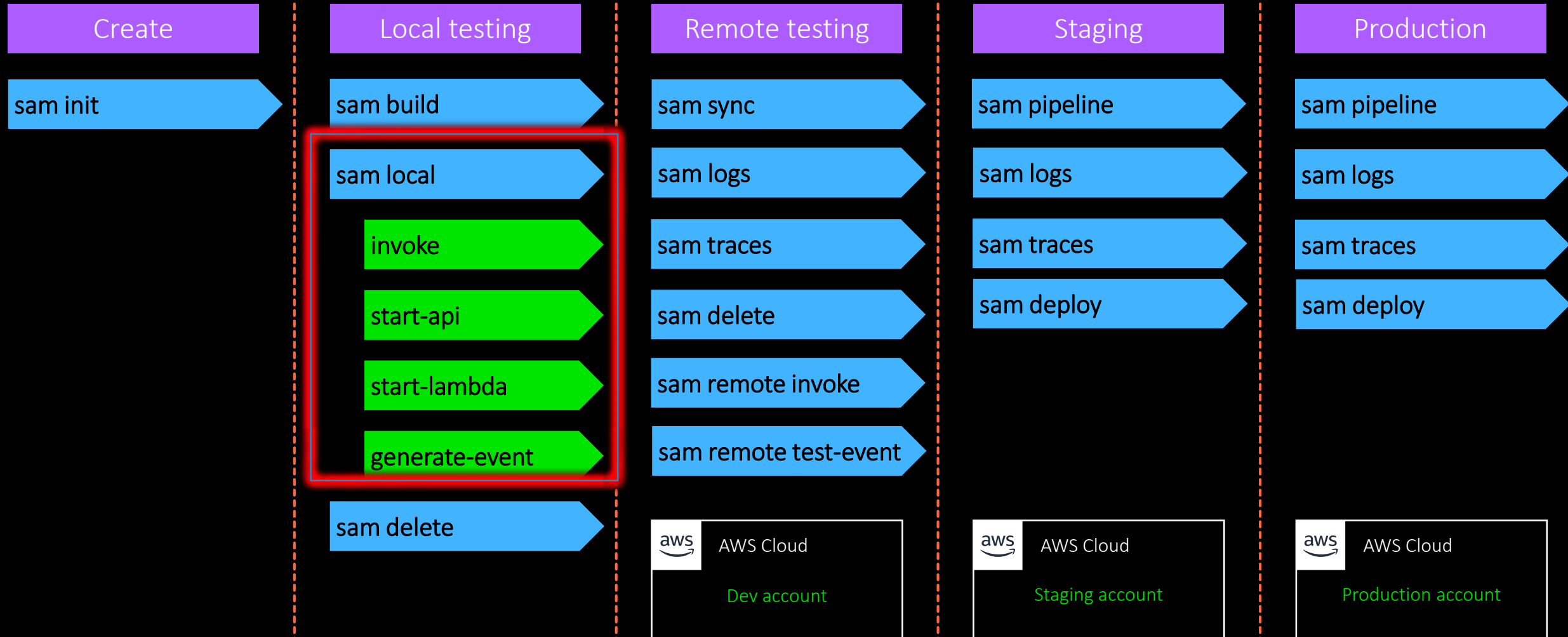


Integration Modules

Integration Services Team –
Team A



The development cycle



AWS SAM local

Invokes a Lambda function directly emulating a service emitting an event and the Lambda service reacting.



Amazon S3



AWS Step Functions



Amazon Lex



Amazon DynamoDB



AWS Lambda



Amazon Cognito



Amazon SageMaker



AWS Batch



Amazon CloudWatch

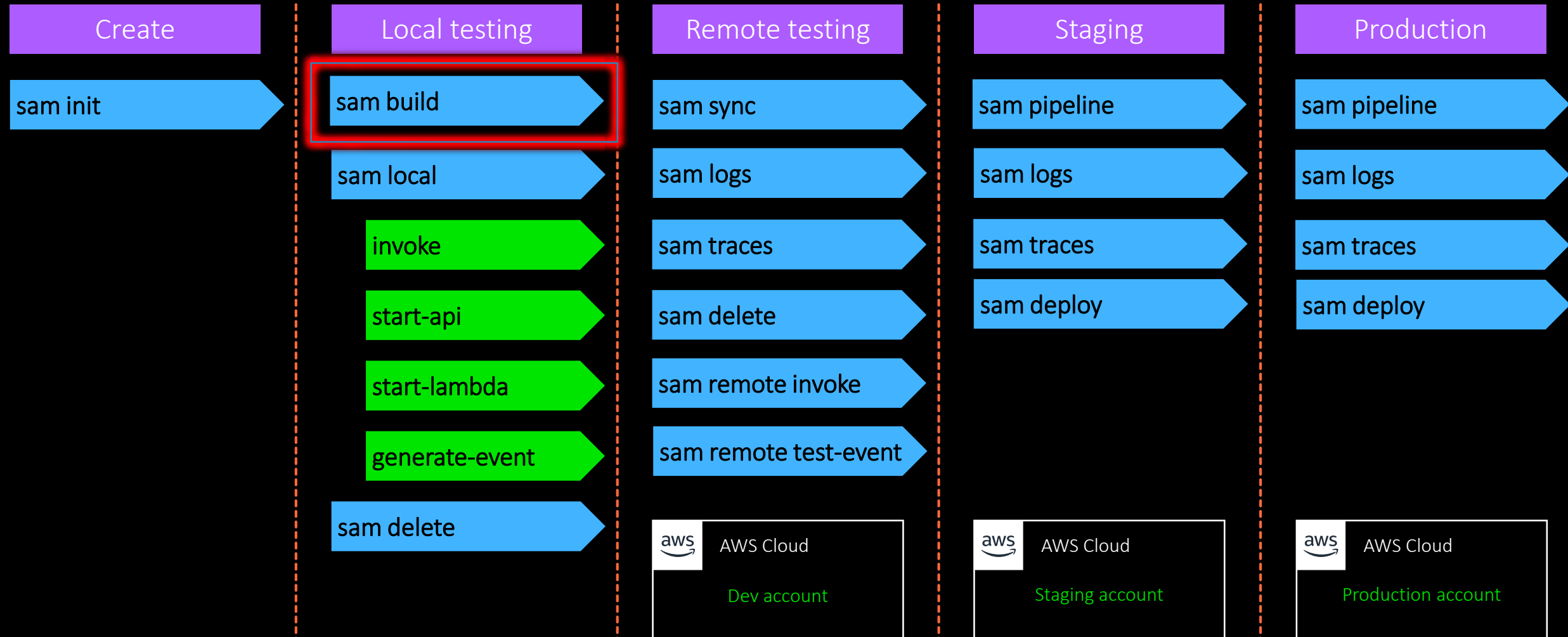
CLI flags

`--event -e`

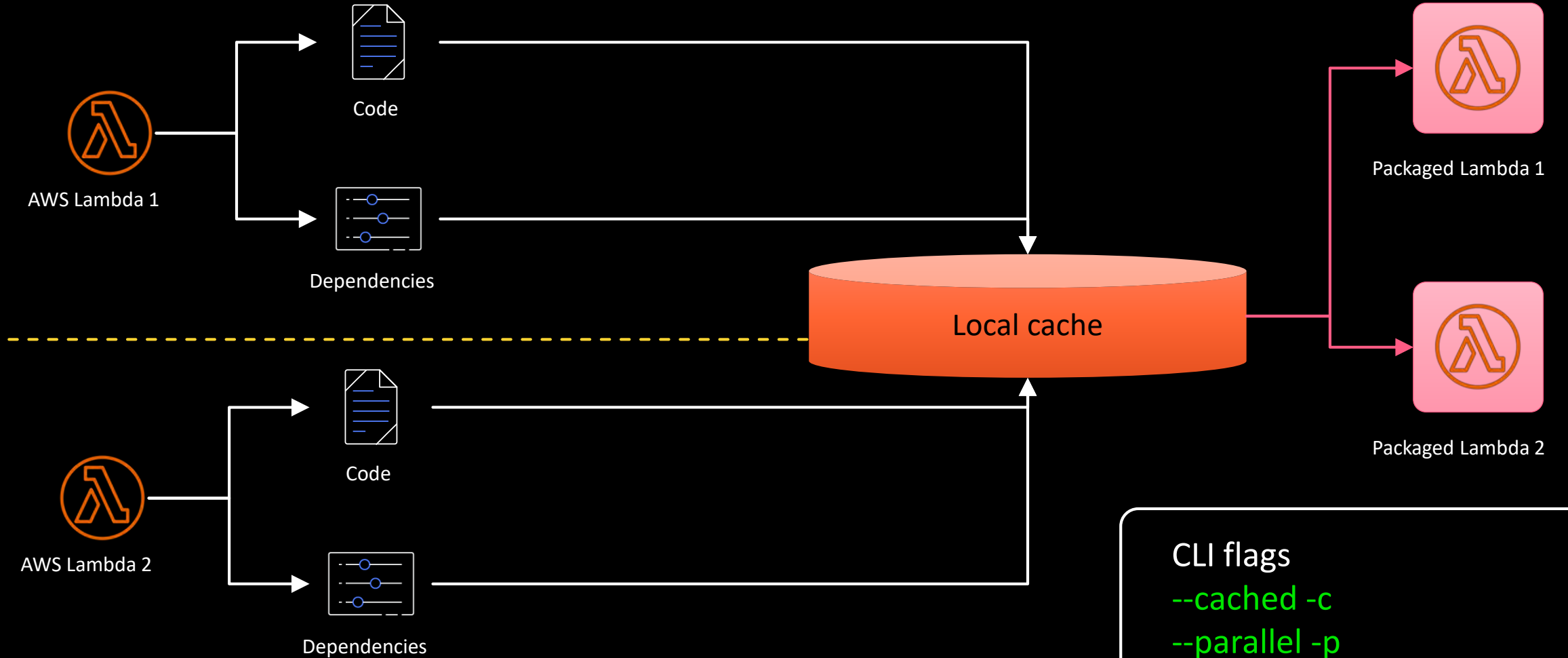
`--shutdown`



The development cycle



AWS SAM Accelerate – sam build



CLI flags

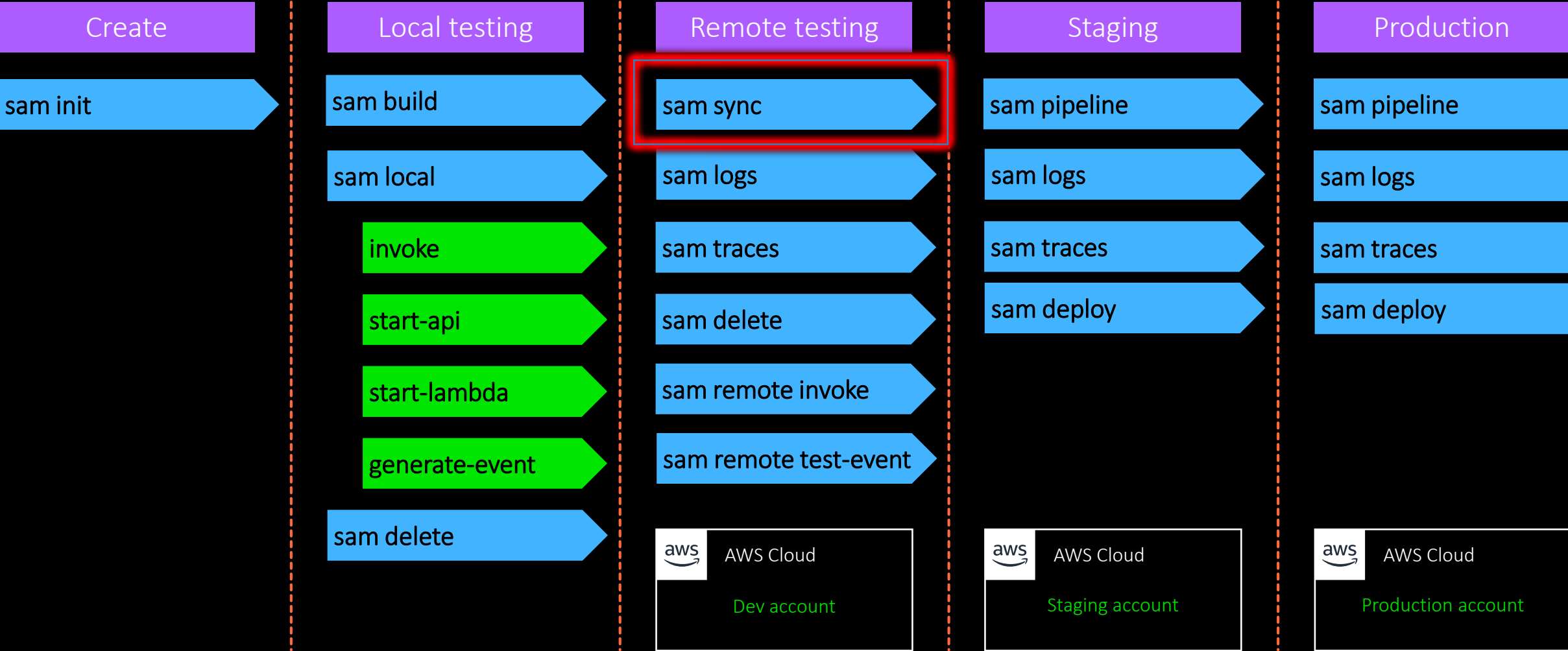
`--cached -c`

`--parallel -p`

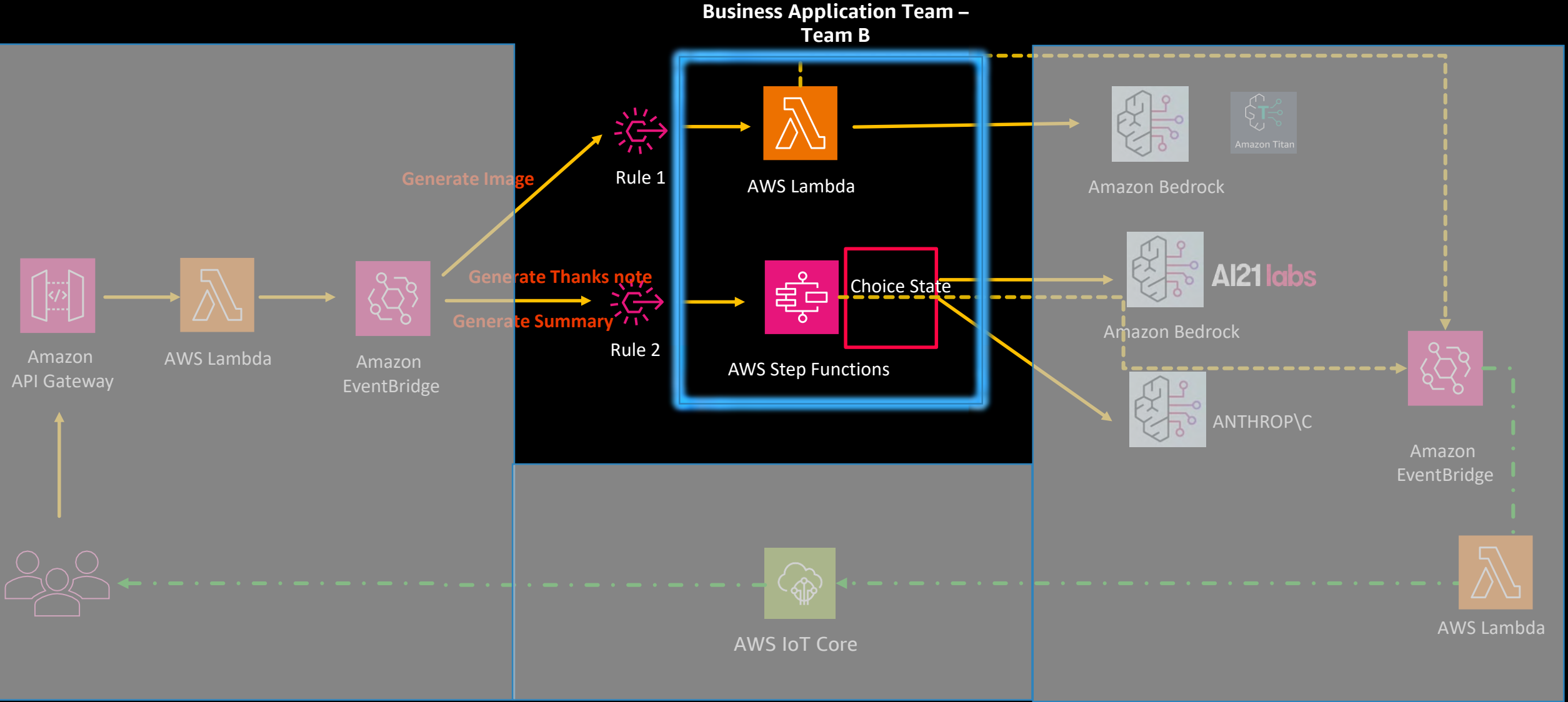
`--use-container -u`



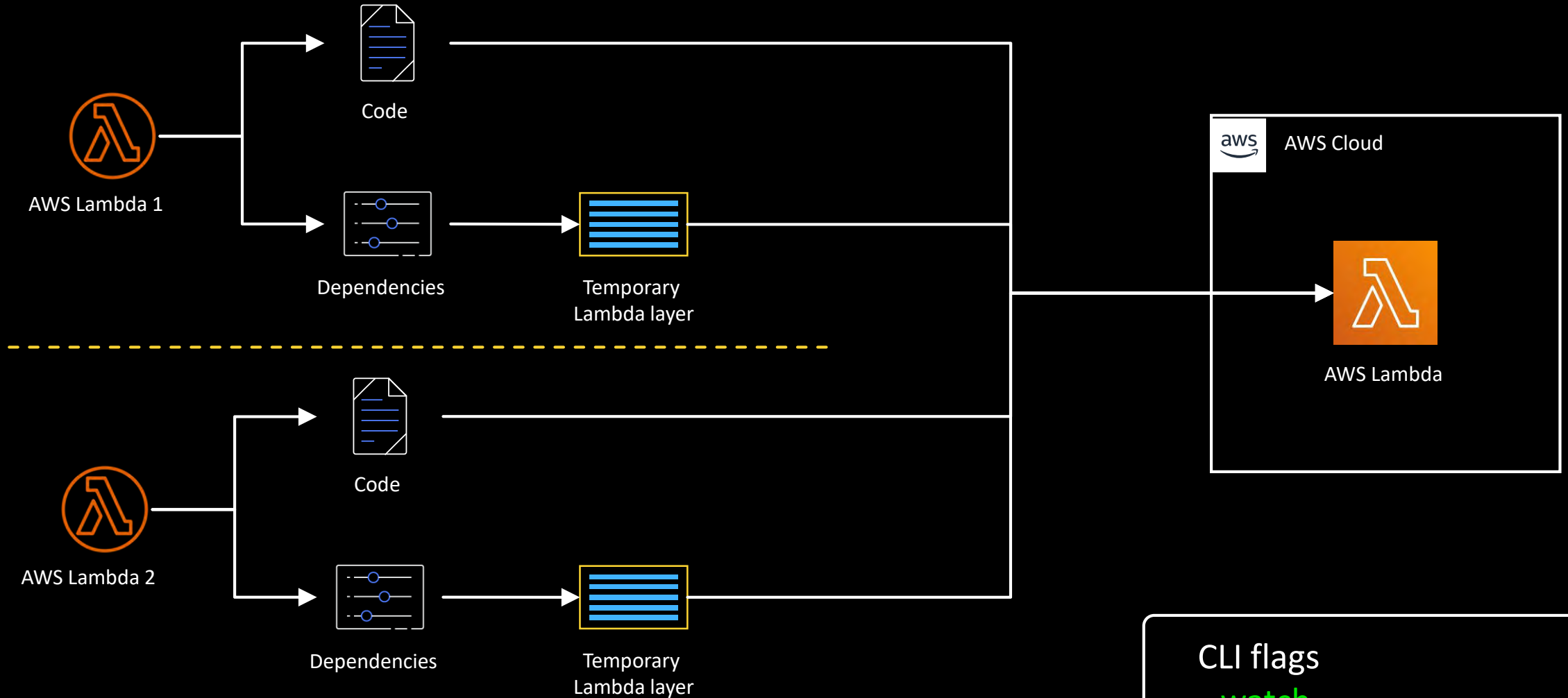
The development cycle



Business Function Modules

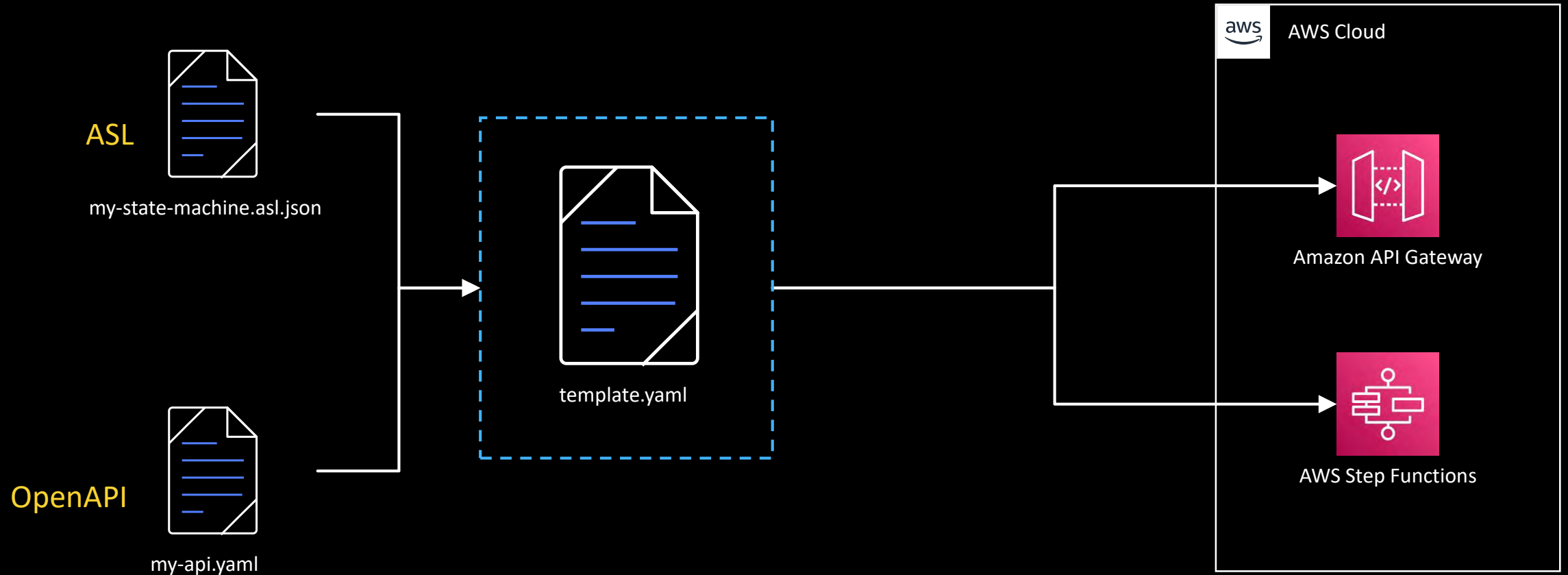


AWS SAM Accelerate – sam sync



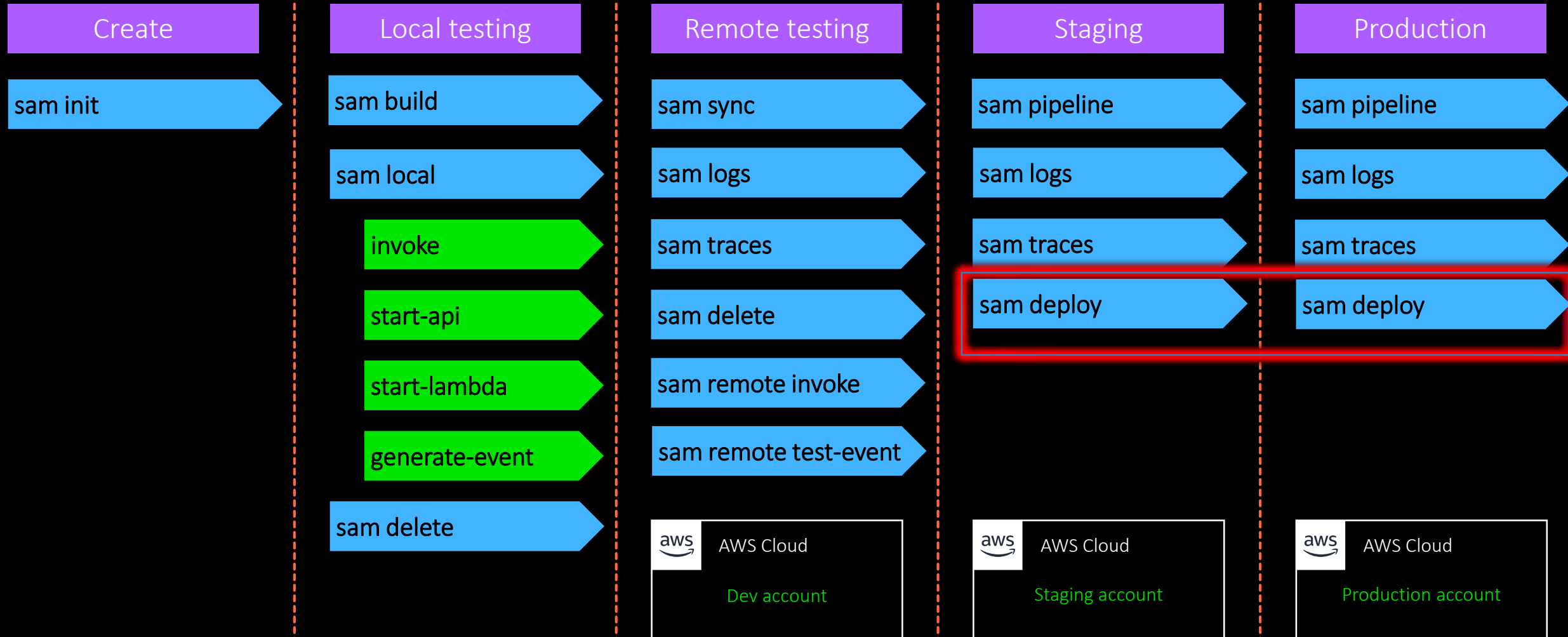
CLI flags
--watch

AWS SAM Accelerate – sam sync

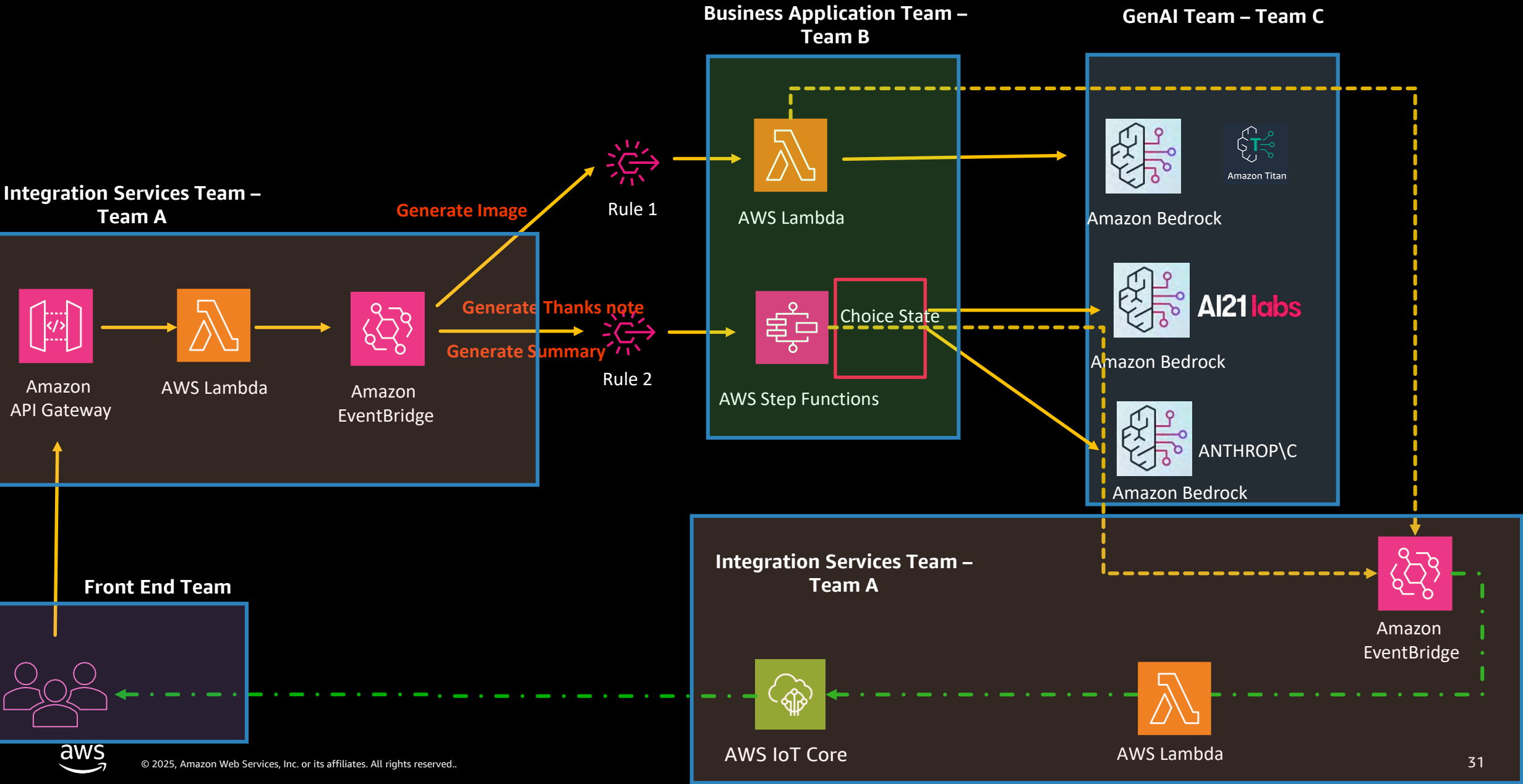


CLI flags
--watch

The development cycle



Serverless GenAI App – Text Summary and Image Generation



AWS SAM Accelerate - Nested stack

TeamAStack

Type: AWS::Serverless::Application

Properties:

Location: ./teamA.yaml

Parameters:

AppId: 112d43

TeamBStack

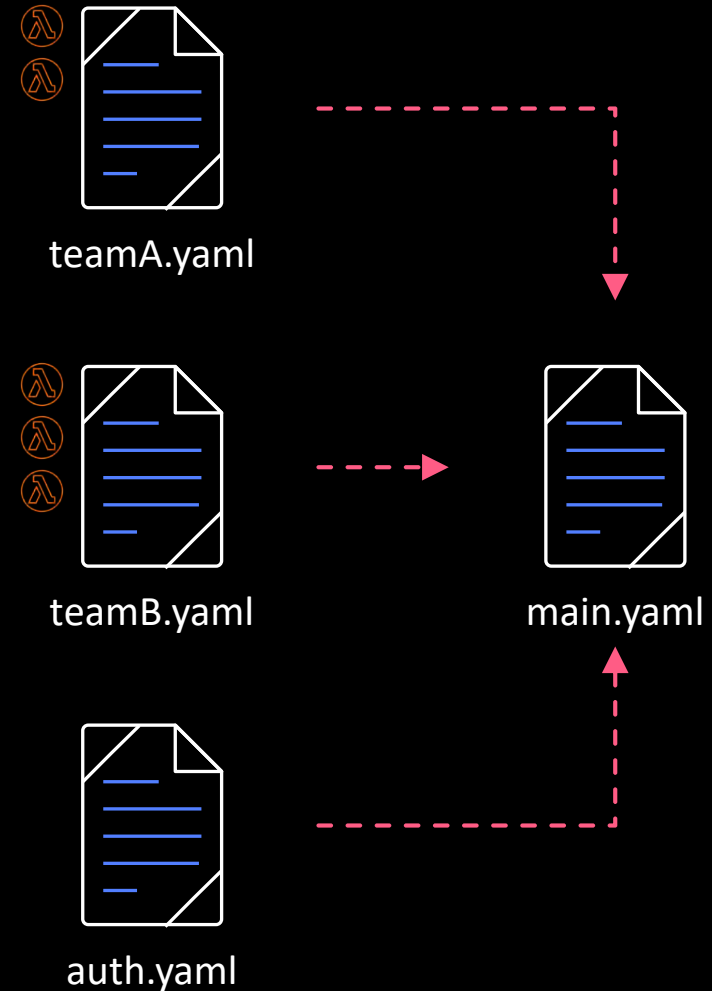
Type: AWS::Serverless::Application

Properties:

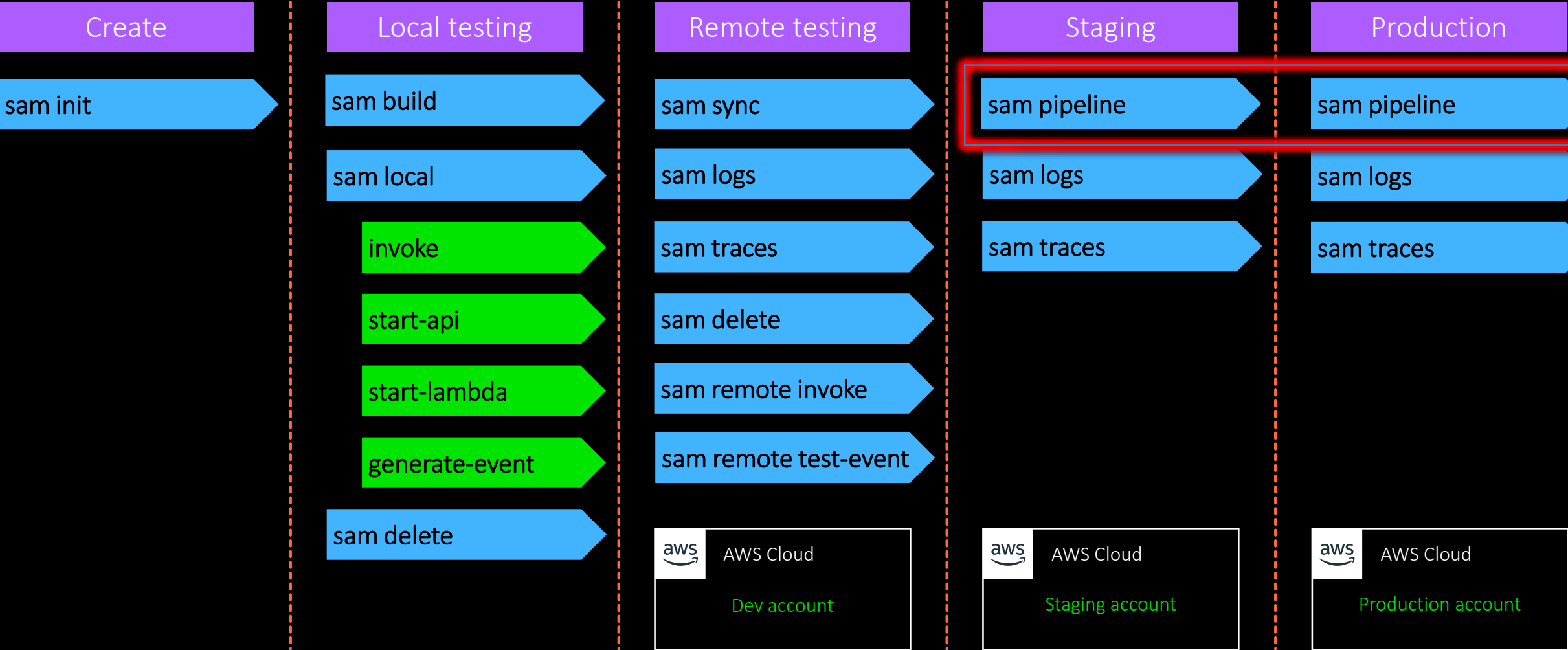
Location: ./TeamB.yaml

Parameters:

AppId: 112d43

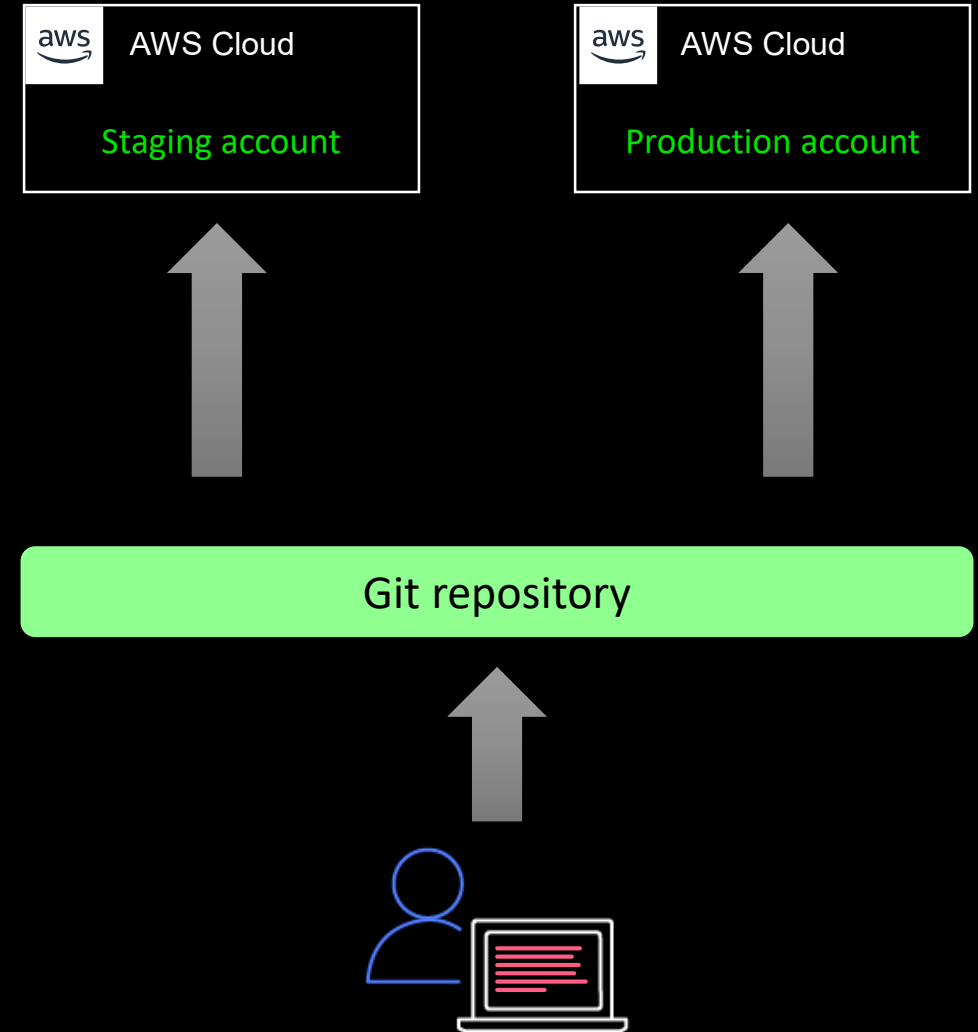


The development cycle



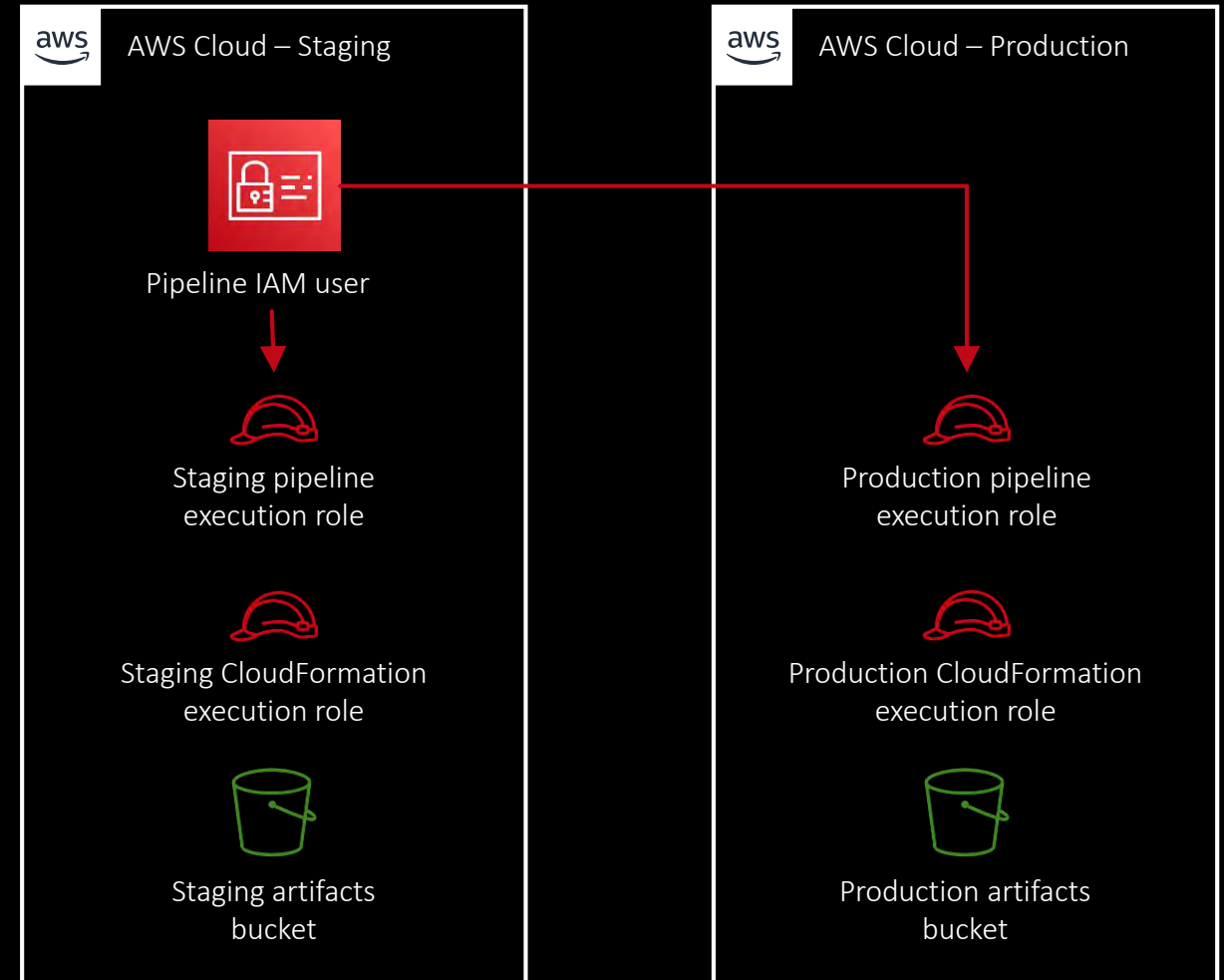
AWS SAM pipeline

- Generates the resource and configuration required to build a continuous integration and continuous delivery (CI/CD) pipeline
- Supports
 - AWS CodePipeline
 - GitHub Actions
 - Jenkins CI
 - GitLab
 - BitBucket

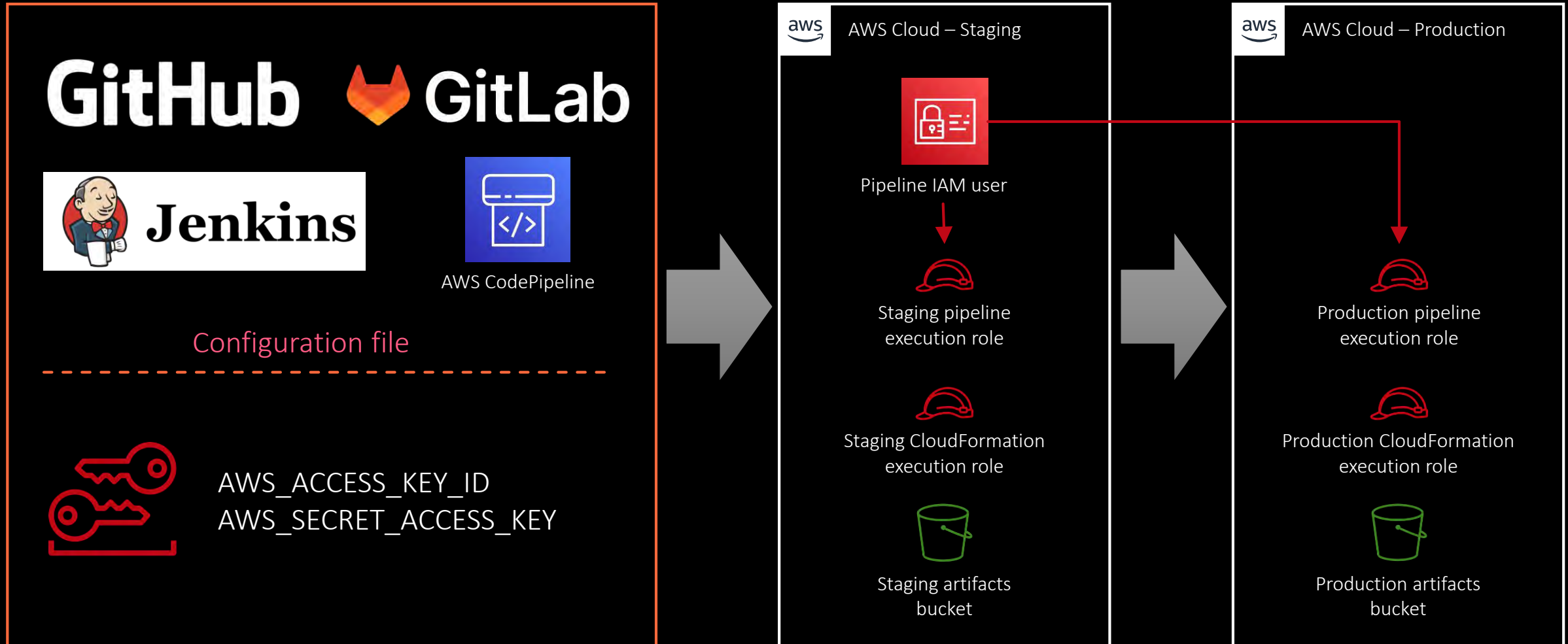


AWS SAM pipeline bootstrap

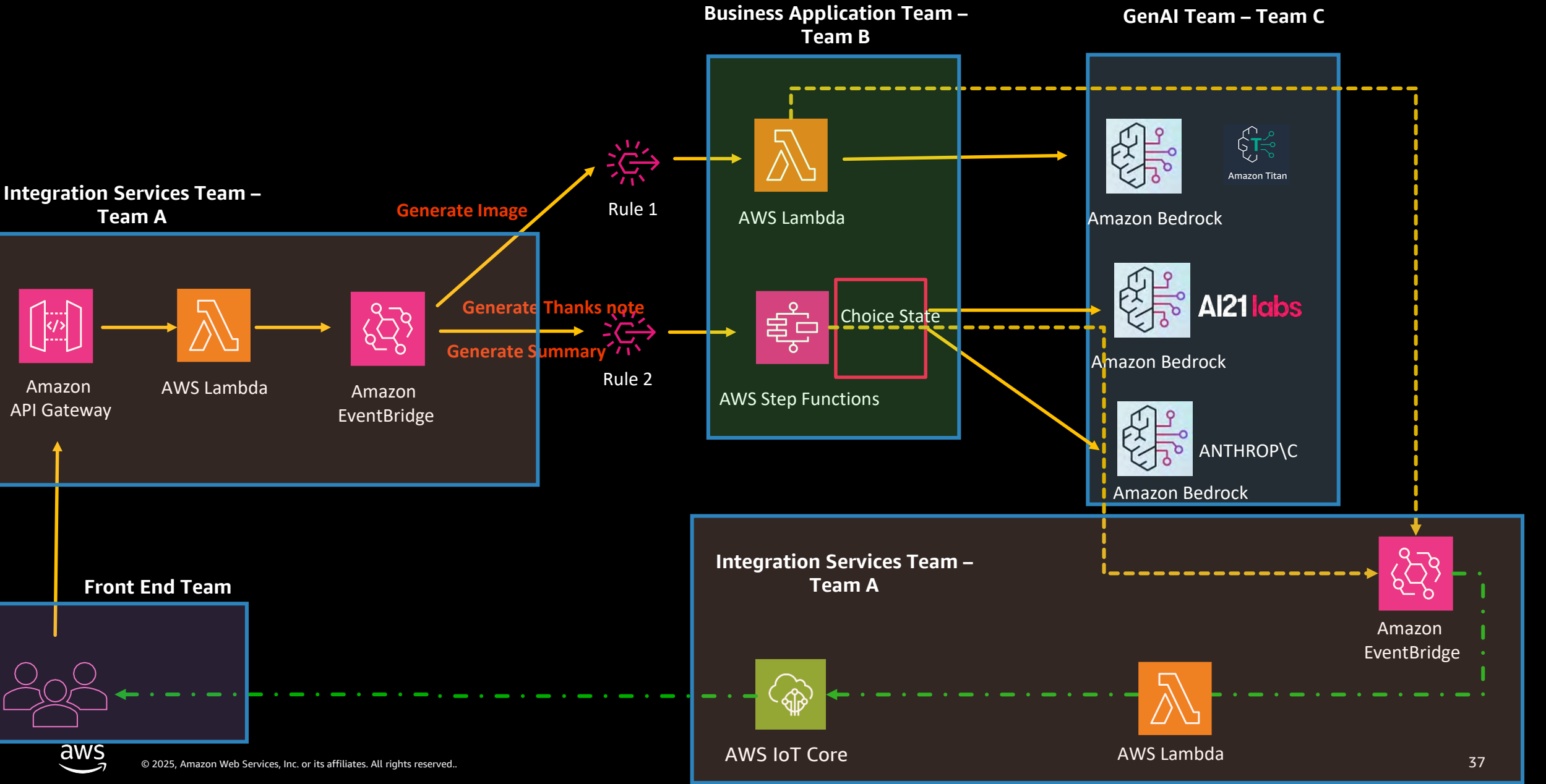
Creates the AWS resources and permissions required to deploy application artifacts from your code repository into your AWS environments



AWS SAM pipeline



Serverless GenAI App – Text Summary and Image Generation



Useful Resources



[ServerlessLand
Patterns](#)



[Sessions With
Sam - YouTube](#)

Thank You

Urmila Raju (She/Her)
Senior Solutions Architect, AWS

Connect with me on LinkedIn

