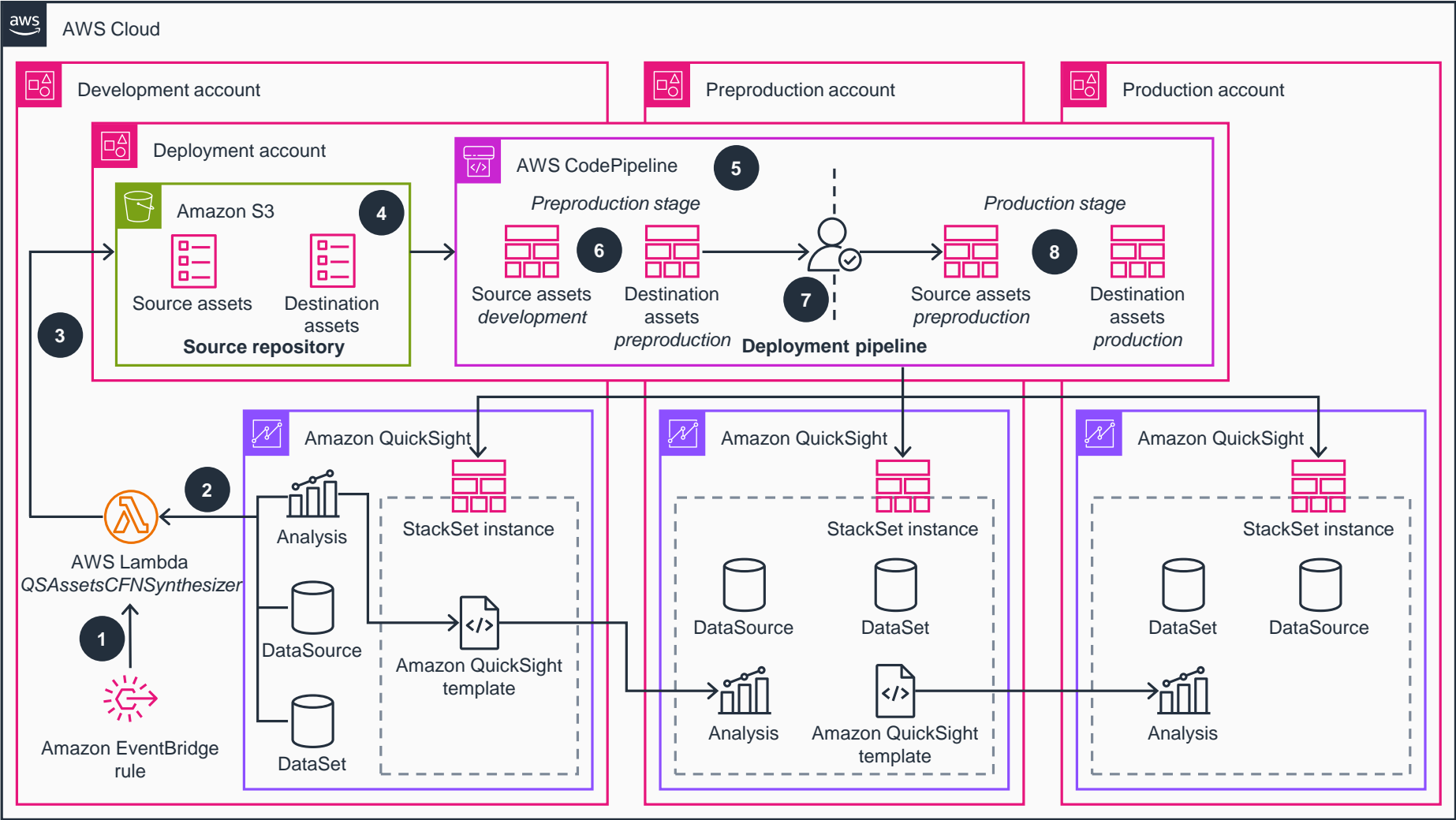


# Guidance for Multi-Account Environments on Amazon QuickSight

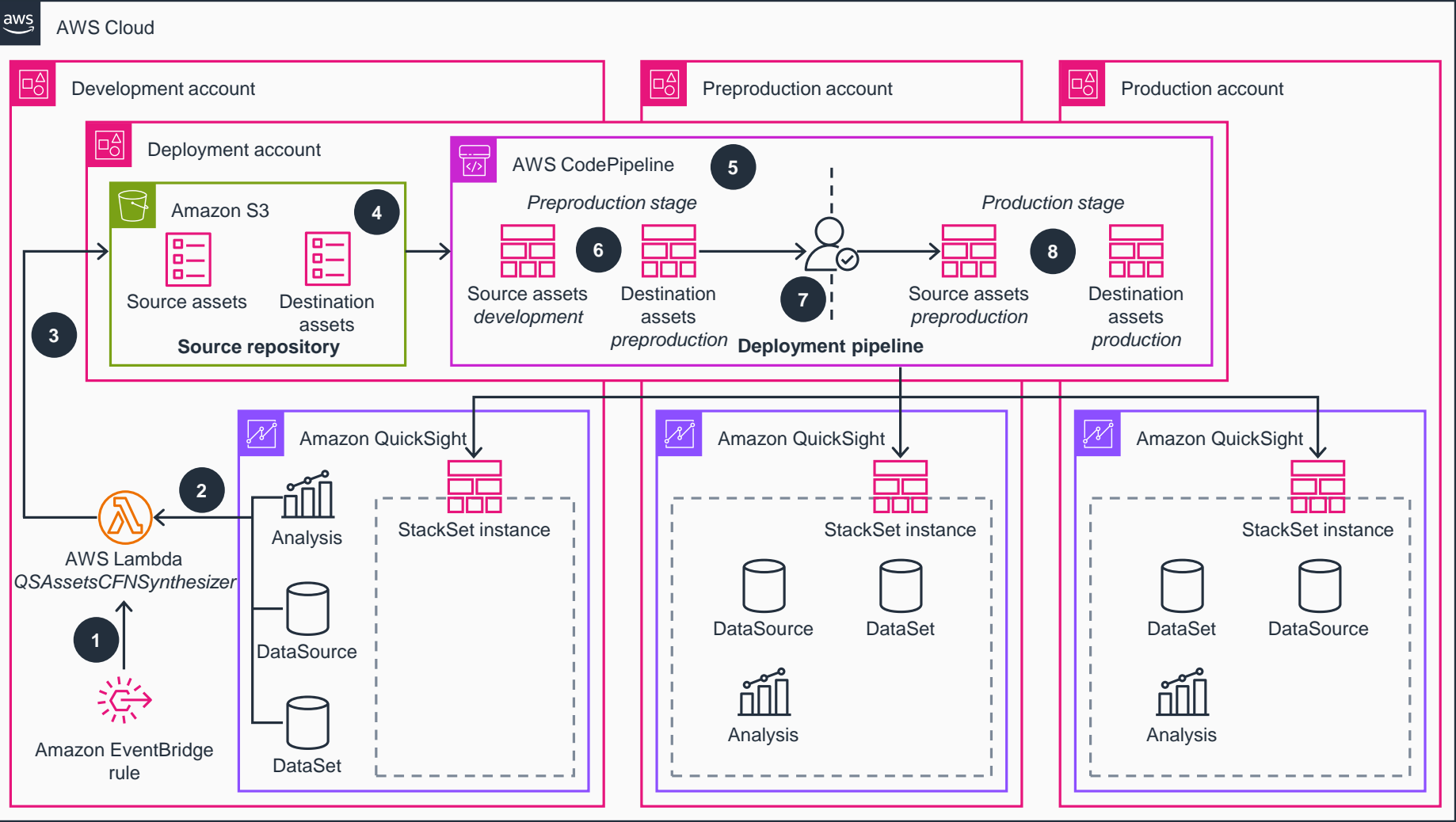
This architecture diagram shows how to build a multi-account environment where assets from different development-cycle phases are hosted in separate accounts and promoted using a continuous integration and continuous delivery (CI/CD) pipeline. There are two deployment modes for this solution, the first one uses an Amazon QuickSight template, the second uses an asset bundle API. This slide displays the QuickSight template. For details on the asset bundle API deployment mode, refer to the next slide.



- 1 An **Amazon EventBridge** rule invokes the **QSAssetsCFNSynthesizer AWS Lambda** function when a new dashboard version is deployed.
- 2 The **Lambda** function describes the **Amazon QuickSight** assets that were created manually in the development account and generates **AWS CloudFormation** templates.
- 3 The **CloudFormation** templates are uploaded to **Amazon Simple Storage Service (Amazon S3)**. Two templates are generated: *source assets* which create an analysis template, and *destination assets*, which create an analysis from the **QuickSight** template and its required datasets and data sources.
- 4 **Amazon S3** is configured as the source stage for **AWS CodePipeline** and acts as the source repository for the pipeline deployments.
- 5 **CodePipeline** is configured with two deployment stages for production and preproduction. The promotion to production is protected with a manual approval to prevent uncontrolled promotion of assets.
- 6 The first stage will deploy the source assets **CloudFormation** template in the development account, which creates a **QuickSight** template in development that models the analysis to be promoted across the environments. Then the destination assets' **CloudFormation** template is deployed in preproduction, creating a **QuickSight** analysis and its dependent assets (such as DataSource and DataSets).
- 7 Deployment to production will be kept on hold with a manual approval until the assets have been reviewed in preproduction.
- 8 Once the assets have been reviewed and approved, the second stage will deploy the source assets template to model the **QuickSight** assets that were previously created in preproduction. The second stage will then deploy the destination assets to create the **QuickSight** analysis and its depending assets in production.

# Guidance for Multi-Account Environments on Amazon QuickSight

This slide displays an asset bundle API deployment mode.



- 1 An **EventBridge** rule invokes the `QSAssetsCFNSynthesizer` **Lambda** function when a new dashboard version is deployed.
- 2 The **Lambda** function uses the **QuickSight** advanced deployment APIs (AssetBundle) to generate a **CloudFormation** template that models the development analysis and all its depending assets (such as DataSource and DataSets).
- 3 The **CloudFormation** templates are uploaded to **Amazon S3**. Two templates are generated: *source* assets, which will be empty in this case, and *destination* assets, which create an analysis from the **CloudFormation** template generated in the previous step.
- 4 **Amazon S3** is configured as the source stage for **CodePipeline** and acts as the source repository for the pipeline deployments.
- 5 **CodePipeline** is configured with two deployment stages for production and preproduction. The promotion to production is protected with a manual approval to prevent uncontrolled promotion of assets.
- 6 The first stage will deploy the source assets **CloudFormation** template in the development account (empty in this deployment mode) and then the destination assets **CloudFormation** template in preproduction, creating a **QuickSight** analysis and its depending assets (such as DataSource and DataSets).
- 7 Deployment to production will be kept on hold with a manual approval until the assets have been reviewed in preproduction.
- 8 Once the assets have been reviewed and approved, the second stage will deploy the source assets template in preproduction (empty in this deployment mode) and then the destination assets to create the **QuickSight** analysis and its depending assets in production.