Serverless Backend for Frontend on AWS

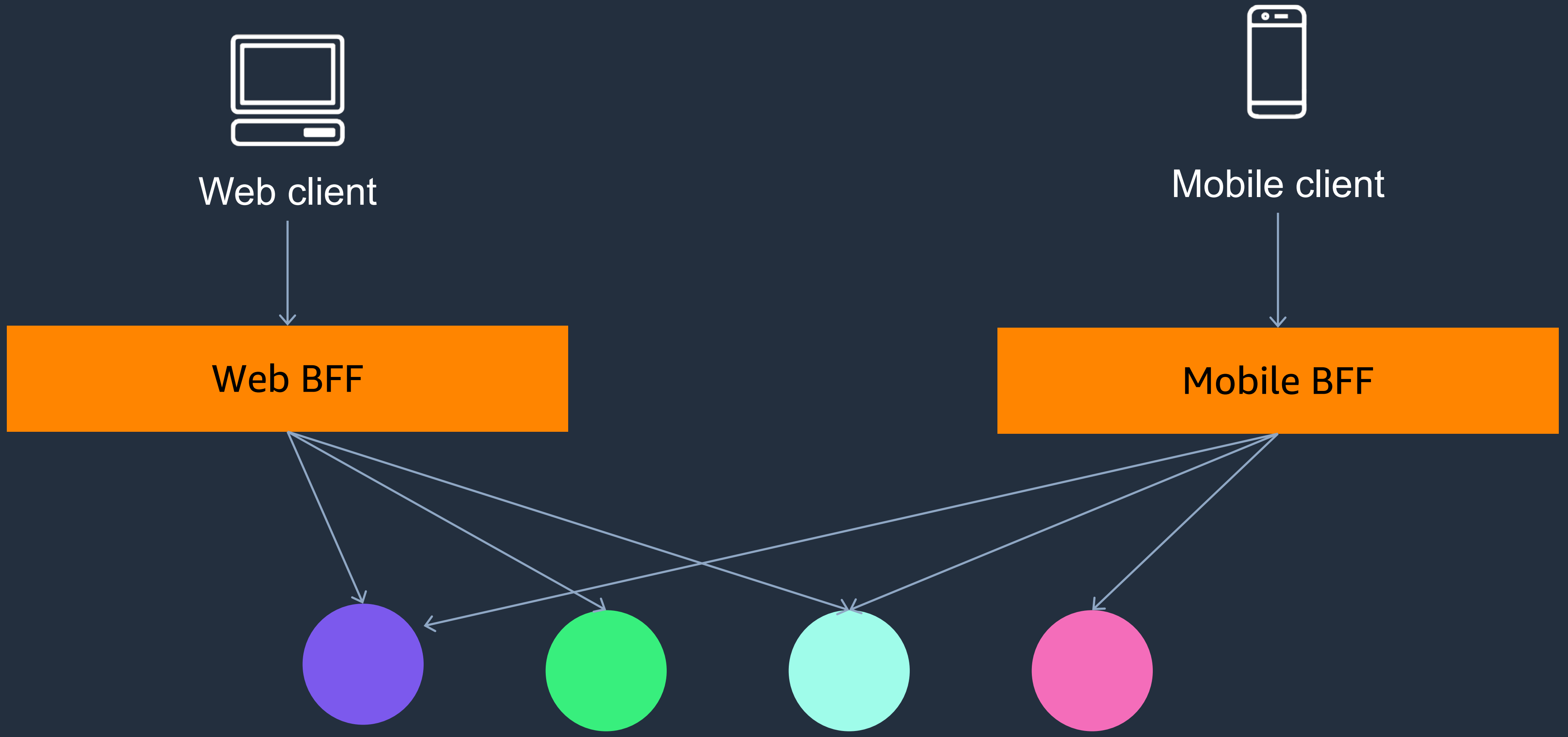
Roman Boiko

Serverless Specialist Solutions Architect
AWS

> "Backends For Frontends solve a pressing concern for mobile development when using microservices."

Sam Newman

Quick recap of the pattern



Build BFF with AWS Serverless services

What does serverless mean?

No infrastructure provisioning, no management

Automatic scaling

Pay for use

Highly available and secure

Serverless Portfolio

APPLICATION PRIMITIVES – COMPUTE AND DATASTORES

Amazon S3

AWS Lambda

AWS Fargate

Amazon DynamoDB

Amazon Aurora Serverless

Amazon Kinesis

APPLICATION INTEGRATION

Amazon SNS

Amazon API Gateway

AWS Step Functions

Amazon EventBridge

Amazon SQS

AWS AppSync

Developer Tools

AWS CloudFormation

AWS Cloud9

AWS CodePipeline

AWS Config

AWS CloudTrail

Amazon CloudWatch

AWS X-Ray

AWS Serverless Application Repository

SECURITY AND ADMINISTRATION

AWS IAM

AWS SSO

Amazon GuardDuty

Amazon Inspector

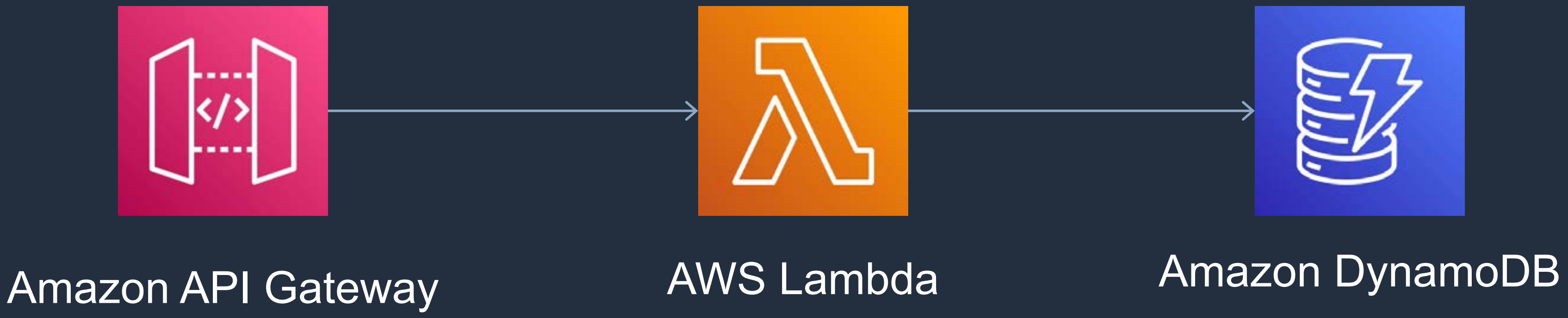
Amazon VPC

AWS WAF

AWS Shield

Amazon Cognito

How to start

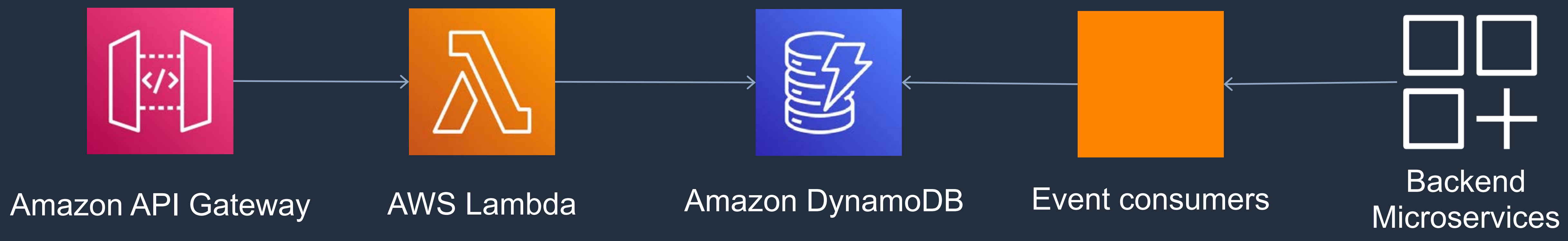


Amazon API Gateway

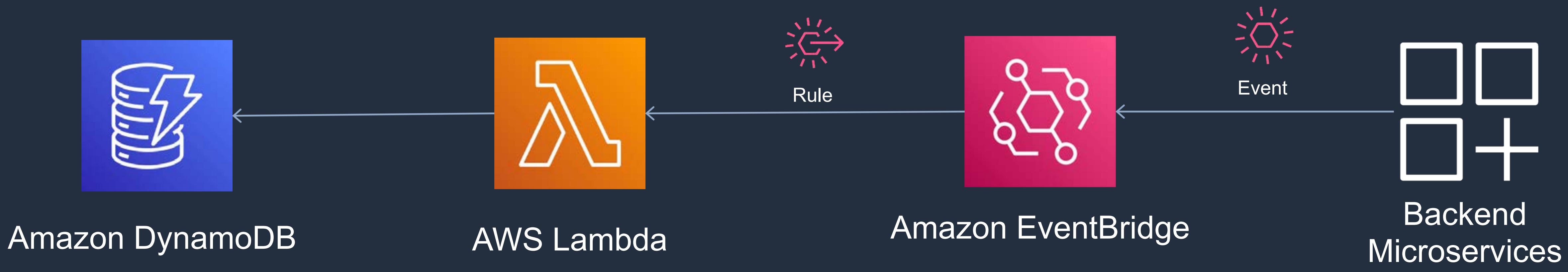
AWS Lambda

Amazon DynamoDB

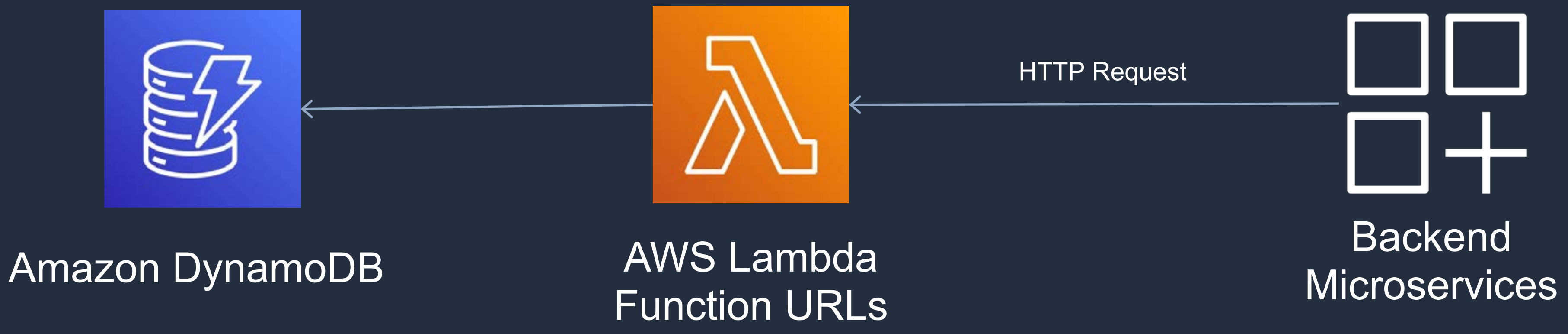
How to get the data – BFF event consumers



Event consumers – with Amazon EventBridge



Event consumers – webhook



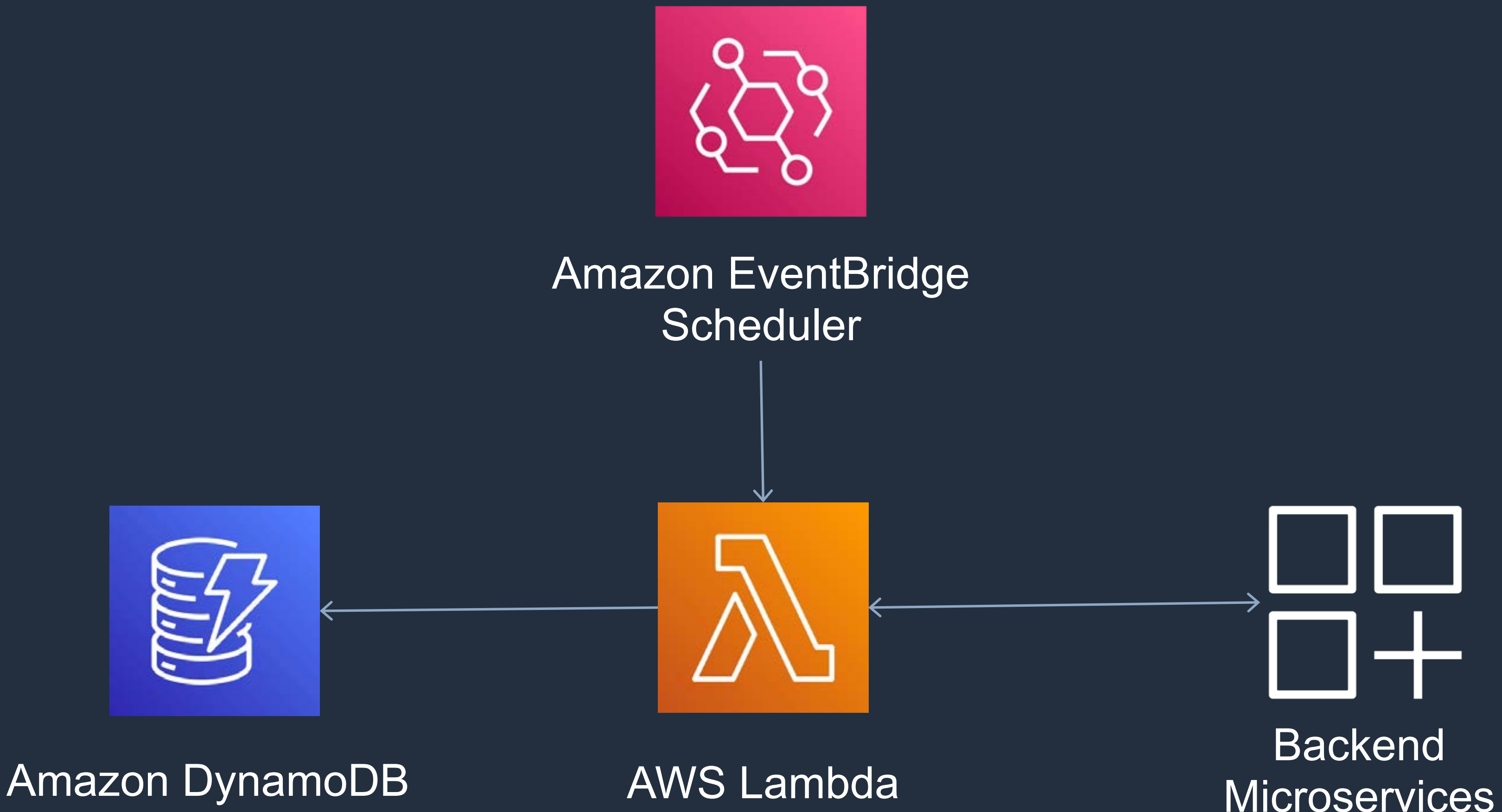
Amazon DynamoDB

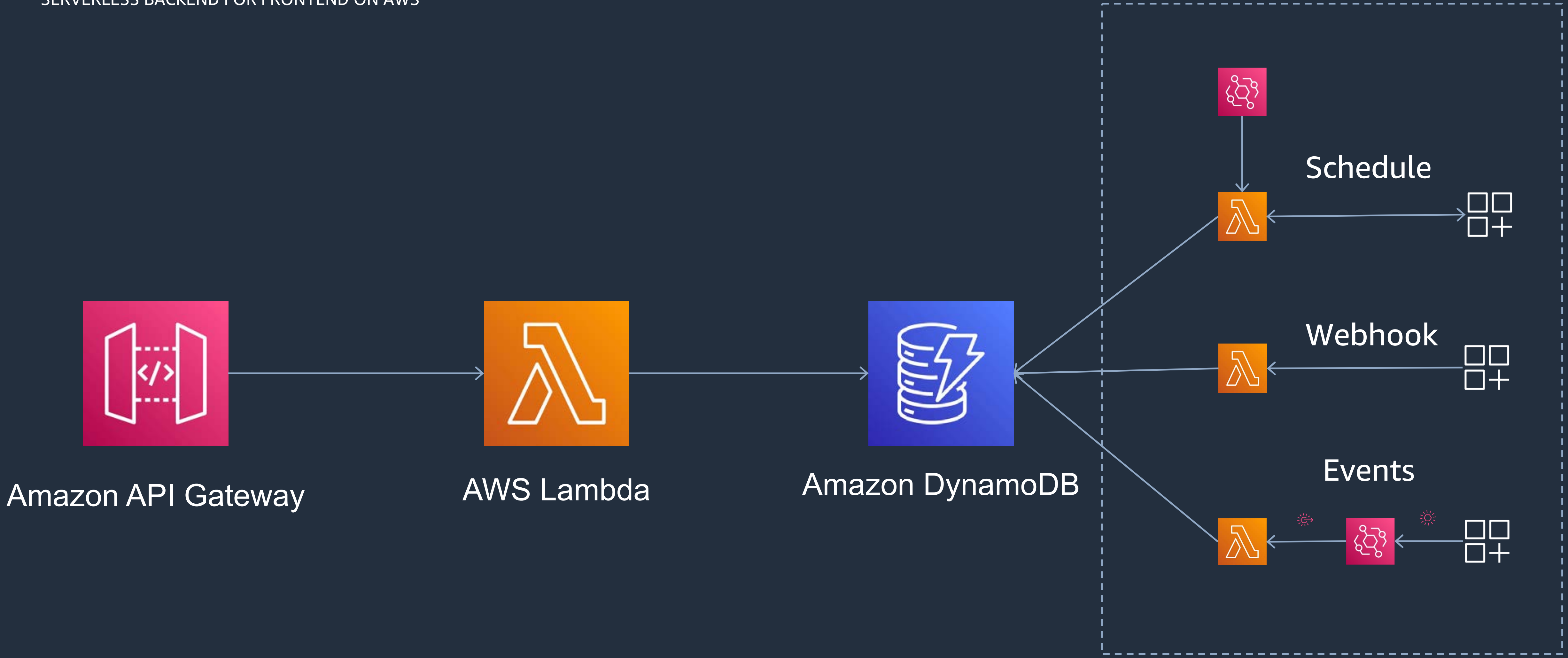
AWS Lambda Function URLs

HTTP Request

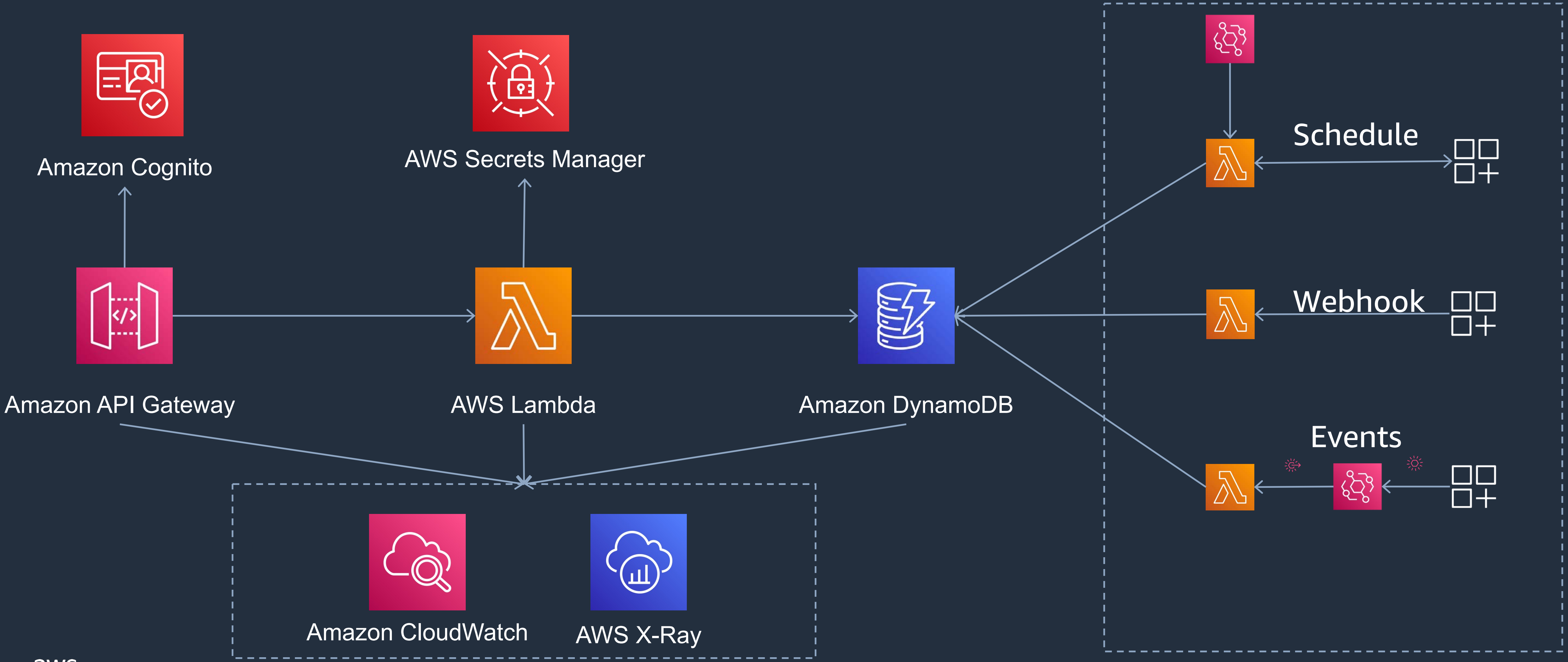
Backend Microservices

Event consumers – polling




Amazon API Gateway
AWS Lambda
Amazon DynamoDB
Schedule
Webhook
Events

Security and Observability



What is
AWS Step Functions?

What is AWS Step Functions?

A serverless workflow orchestration service offered by AWS.

A workflow built on AWS Step Functions…

…is built using a state machine.

…is composed of steps called states.

…moves from one state to another via a state transition.

…is written using Amazon States Language or ASL (think of it as the workflow assembly language).

…can be used to orchestrate multiple AWS services.

Step Functions benefits

Build and update apps quickly

AWS Step Functions lets you build visual workflows that enable fast translation of business requirements into technical requirements. You can build applications in a matter of minutes, and when needs change, you can swap or reorganize components without customizing any code.

Write less code

AWS Step Functions manages the logic of your application for you, and implements basic primitives such as branching, parallel execution, and timeouts. This removes extra code that may be repeated in your microservices and functions.

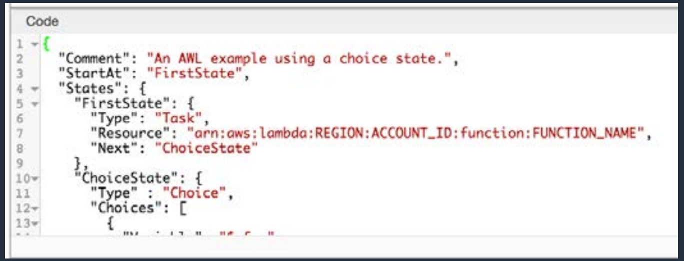
Improve resiliency

AWS Step Functions manages state, checkpoints and restarts for you to make sure that your application executes in order and as expected. Built-in try/catch, retry and rollback capabilities deal with errors and exceptions automatically.

Visual workflows

Define

Amazon States Language
JSON



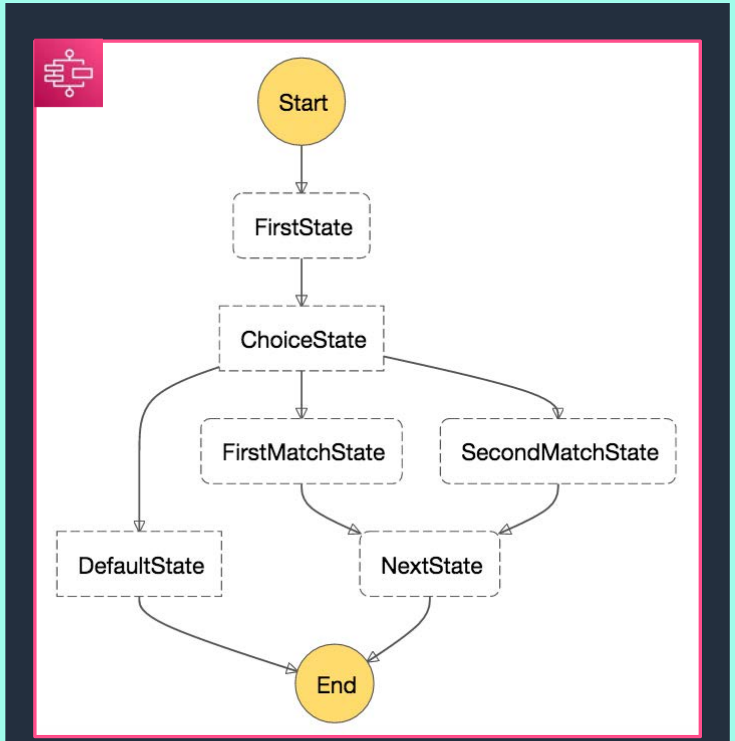
Code

```
{
  "Comment": "An AWL example using a choice state.",
  "StartAt": "FirstState",
  "States": {
    "FirstState": {
      "Type": "Task",
      "Resource": "arn:aws:lambda:REGION:ACCOUNT_ID:function:FUNCTION_NAME",
      "Next": "ChoiceState"
    },
    "ChoiceState": {
      "Type" : "Choice",
      "Choices": [
        {
```

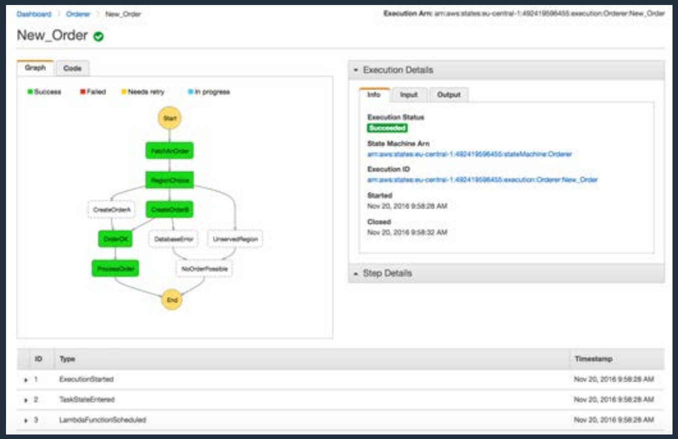
CDK
TypeScript, JavaScript, Python, Java, C#

Data Science SDK
Python

Visualize



Execute and Monitor



Execution Arn: arn:aws:states:eu-central-1:492419596455:execution:Orderer:New_Order

New_Order ✓

Execution Details

Execution Status
Succeeded

State Machine Arn
arn:aws:states:eu-central-1:492419596455:stateMachine:Orderer

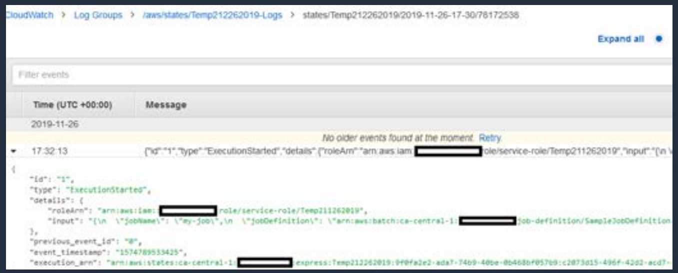
Execution ID
arn:aws:states:eu-central-1:492419596455:execution:Orderer:New_Order

Started
Nov 20, 2016 9:58:28 AM

Closed
Nov 20, 2016 9:58:32 AM

Step Details

ID	Type	Timestamp
1	ExecutionStarted	Nov 20, 2016 9:58:28 AM
2	TaskStateEntered	Nov 20, 2016 9:58:28 AM
3	LambdaFunctionScheduled	Nov 20, 2016 9:58:28 AM


CloudWatch > Log Groups > /aws/states/Temp212262019-Logs > states/Temp212262019/2019-11-26-17-30/78172538
Expand all
Filter events

Time (UTC +00:00)	Message
2019-11-26	
	No older events found at the moment. Retry
17:32:13	{"id":"1","type":"ExecutionStarted","details":{"roleArn":"arn:aws:iam:[redacted]role/service-role/Temp211262019","input":"{\n \

```
{
  "id": "1",
  "type": "ExecutionStarted",
  "details": {
     "roleArn": "arn:aws:iam:[redacted]role/service-role/Temp211262019",
     "input": "{\n  \"jobName\": \"my-job\",\n  \"jobDefinition\": \"arn:aws:batch:ca-central-1:[redacted]job-definition/SampleJobDefinition
  },
  "previous_event_id": "0",
  "event_timestamp": "1574789533425",
  "execution_arn": "arn:aws:states:ca-central-1:[redacted]express:Temp212262019:9f0fa2e2-ada7-74b9-40be-0b468bf057b9:c2073d15-496f-42d2-acd7-
```

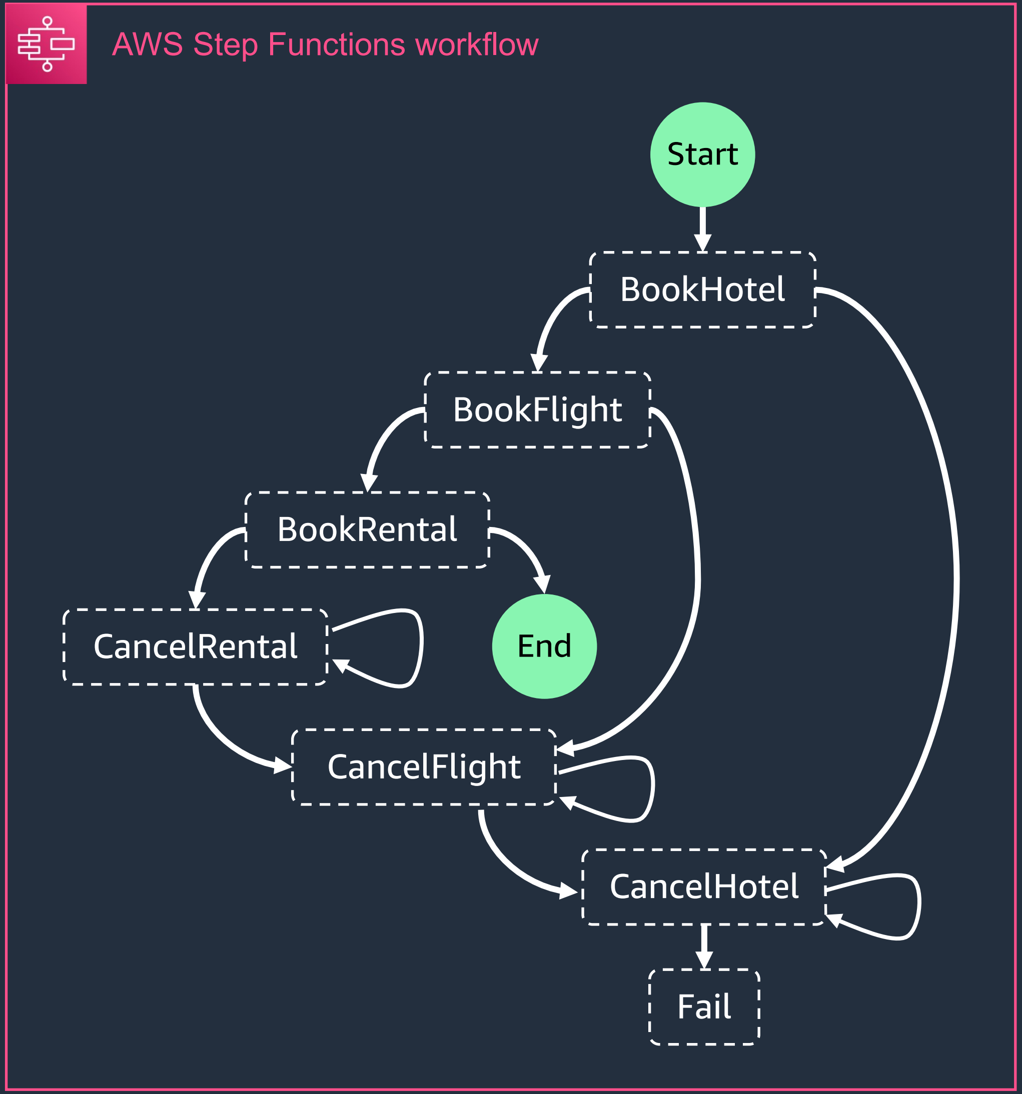
Example Use Case

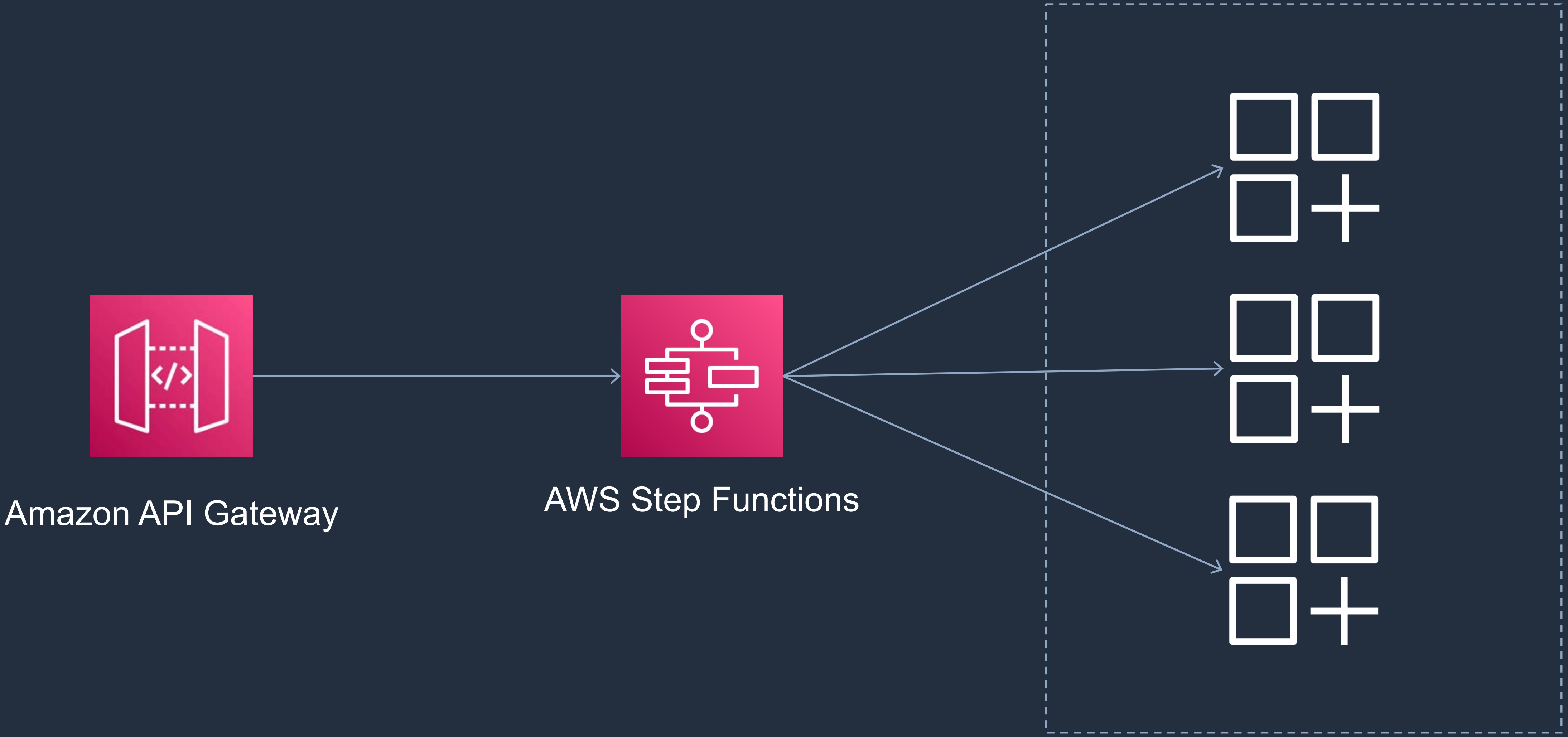
Suppose I want to book my vacation to the **Grand Canyon**…

…but unwind my reservations if I can't rent a car.

Saga Pattern

Roll-back work done by sequence of distributed transactions.




Amazon API Gateway
AWS Step Functions

Useful resources

https://serverlessland.com/patterns

https://www.jeremydaly.com/serverless-microservice-patterns-for-aws/

https://aws.amazon.com/architecture/well-architected/?wa-lens-whitepapers.sort-by=item.additionalFields.sortDate&wa-lens-whitepapers.sort-order=desc

https://docs.aws.amazon.com/wellarchitected/latest/serverless-applications-lens/welcome.html

https://docs.aws.amazon.com/lambda/latest/dg/lambda-typescript.html

https://github.com/aws-samples/serverless-typescript-demo

Thank you!

Roman Boiko

@romannboiko