A Cloud Guru – How we do Serverless

Dale Salter
SOFTWARE DEVELOPMENT LEAD

Platform Statistics

AWS Lambda

240 Million Monthly Lambda Invocations
(100 per second)

Amazon API Gateway

180M Monthly API Gateway Calls
(70 per second)

Amazon Cloudfront

90TB Monthly Data Transferred
(274 MB per second)

HOW A CLOUD GURU DOES

Serverless

Architecture Overview

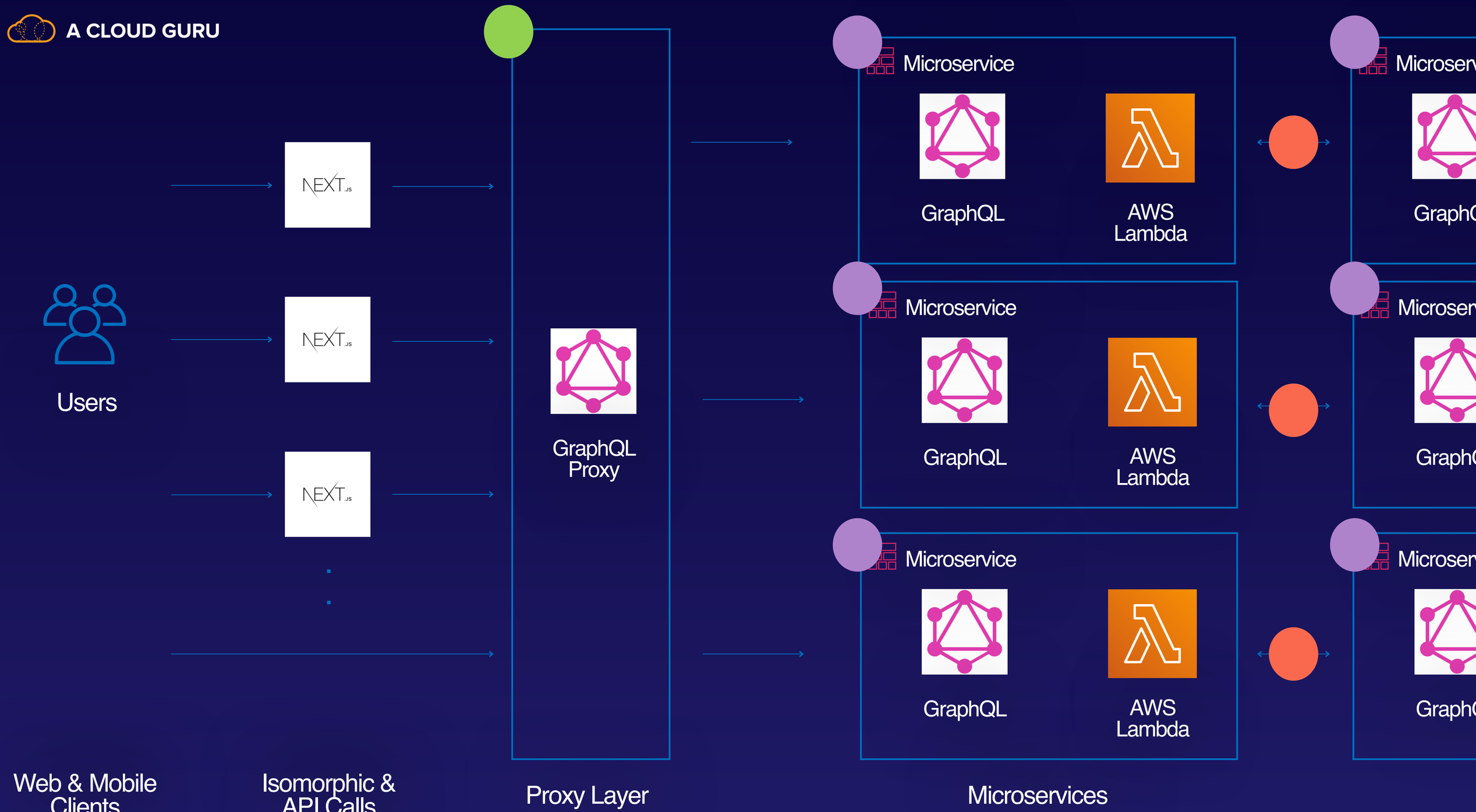
Serverless Challenges

Microservice Layout & Composition

Service To Service Communication

Frontend To Backend Communication

Summary

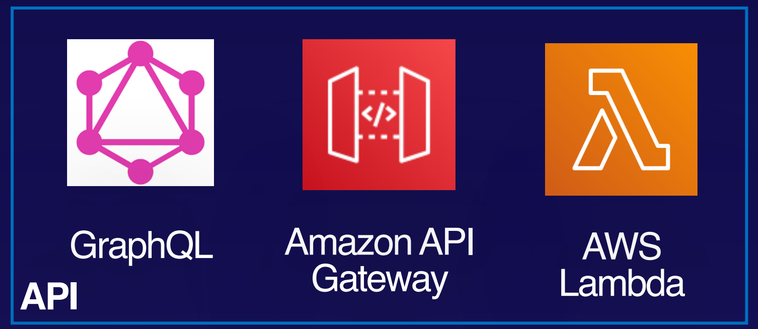

A CLOUD GURU
Users
NEXT.JS
NEXT.JS
NEXT.JS
GraphQL Proxy
Microservice
GraphQL
AWS Lambda
Microservice
GraphQL
AWS Lambda
Microservice
GraphQL
AWS Lambda
Web & Mobile Clients
Isomorphic & API Calls
Proxy Layer
Microservices

The two biggest challenges with completely Serverless Architectures

Latency & Limited Serverless Services

A CLOUD GURU

Microservice Layout & Composition


GraphQL
Amazon API Gateway
AWS Lambda
API

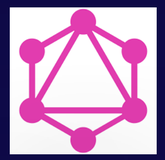
A CLOUD GURU

Stateless Resources

Cloudformation Stack

AWS Lambda

API



GraphQL

Amazon API Gateway

AWS Lambda

Stateful Resources

Cloudformation Stack

Amazon DynamoDB

Amazon S3

Amazon SQS

Amazon SNS

Algolia

Configuration and Core Dependencies

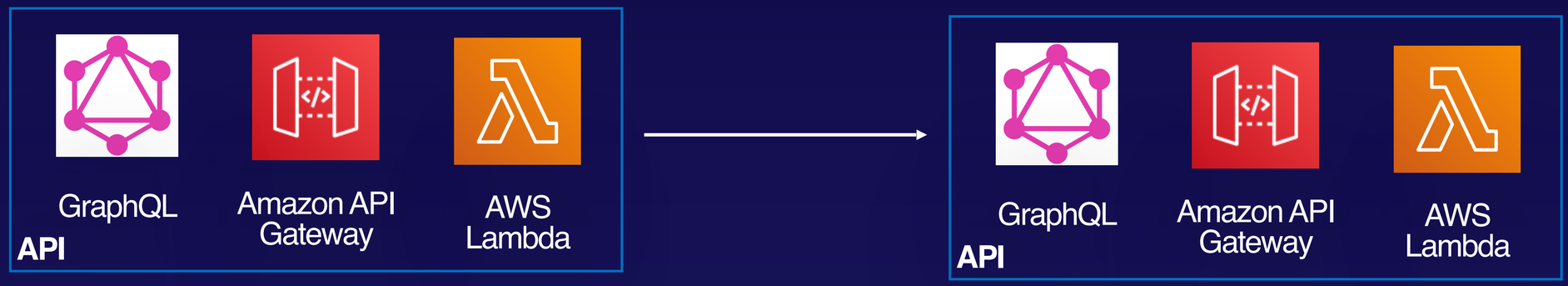
Cloudformation Stack

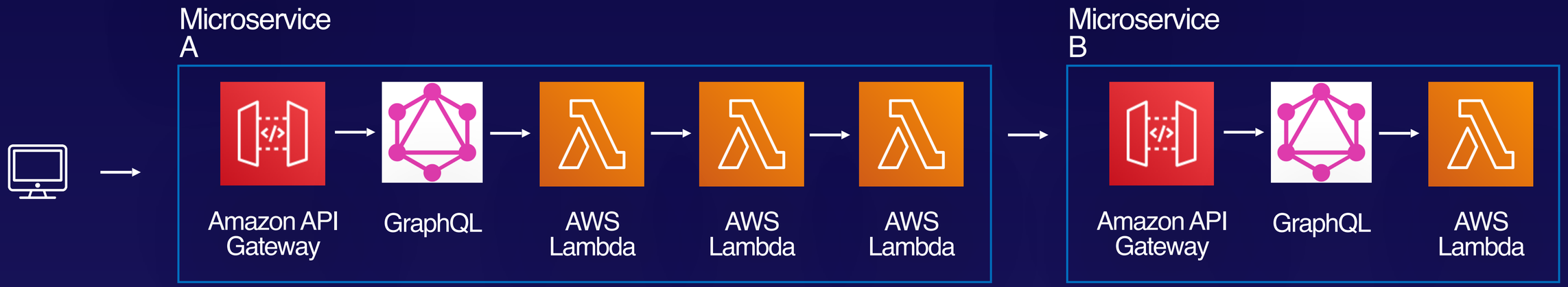
Parameter Store

Custom Resources

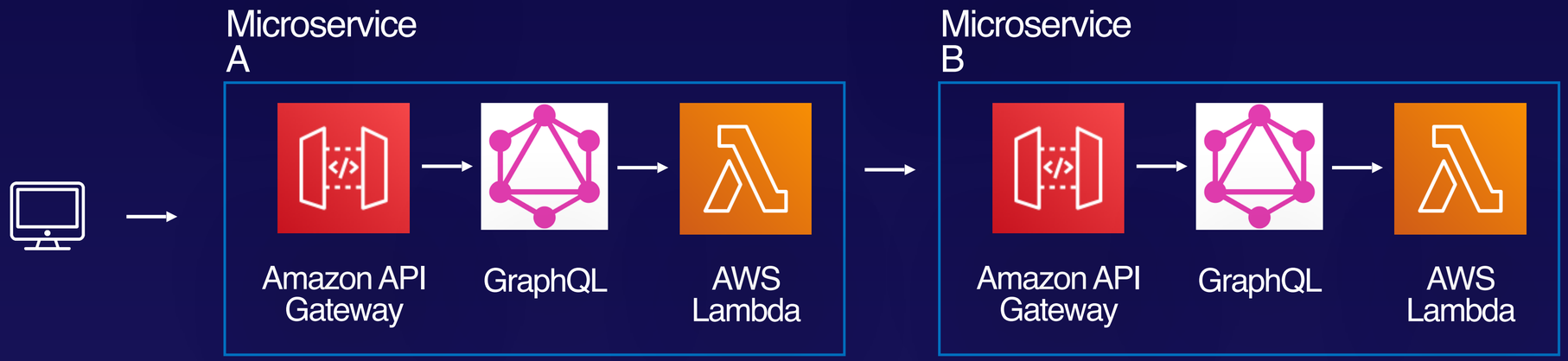
Service To Service Communication


Microservice
A
Microservice
B

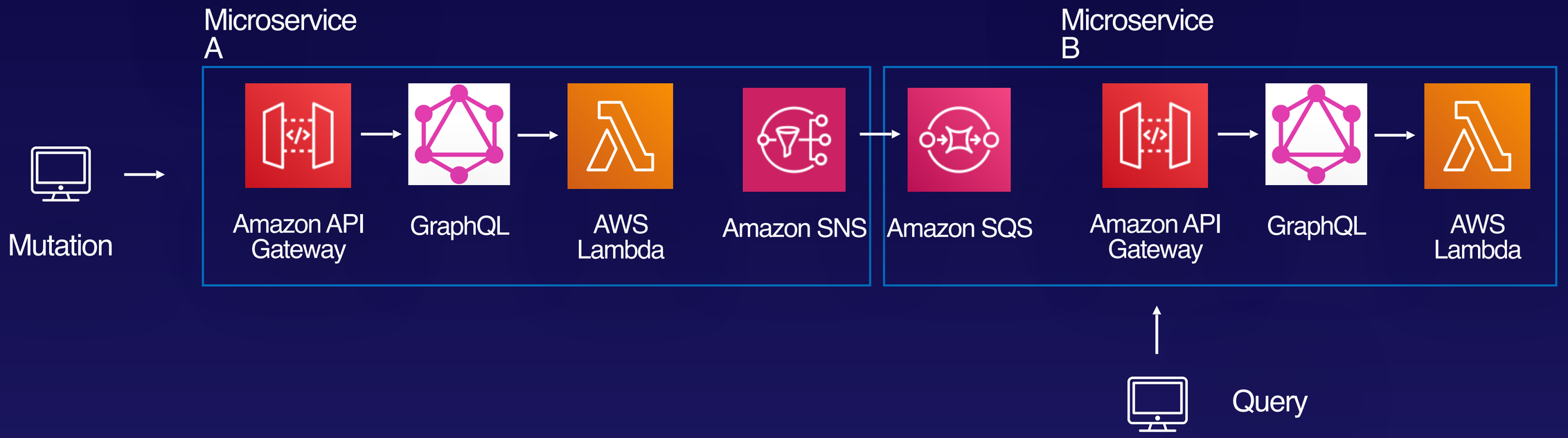

GraphQL
Amazon API Gateway
AWS Lambda
API
GraphQL
Amazon API Gateway
AWS Lambda
API


Microservice A
Amazon API Gateway
GraphQL
AWS Lambda
AWS Lambda
AWS Lambda
Microservice B
Amazon API Gateway
GraphQL
AWS Lambda

800ms – 2.5sec

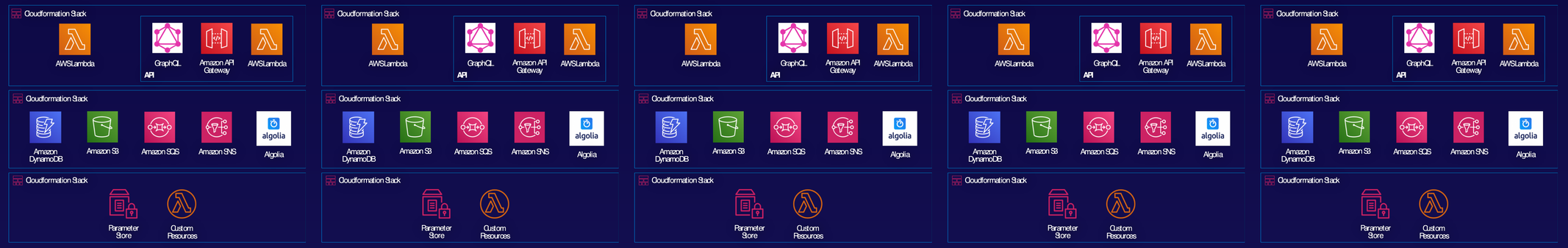

Microservice
A
Amazon API
Gateway
GraphQL
AWS
Lambda
Microservice
B
Amazon API
Gateway
GraphQL
AWS
Lambda

200ms – 800ms


Microservice A
Microservice B
Mutation
Amazon API Gateway
GraphQL
AWS Lambda
Amazon SNS
Amazon SQS
Amazon API Gateway
GraphQL
AWS Lambda
Query

100ms – 250ms

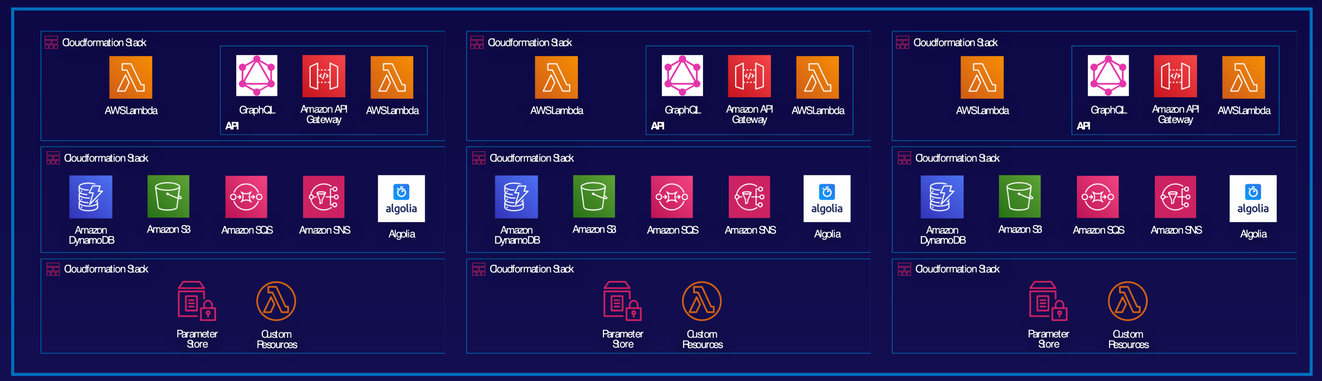
A CLOUD GURU

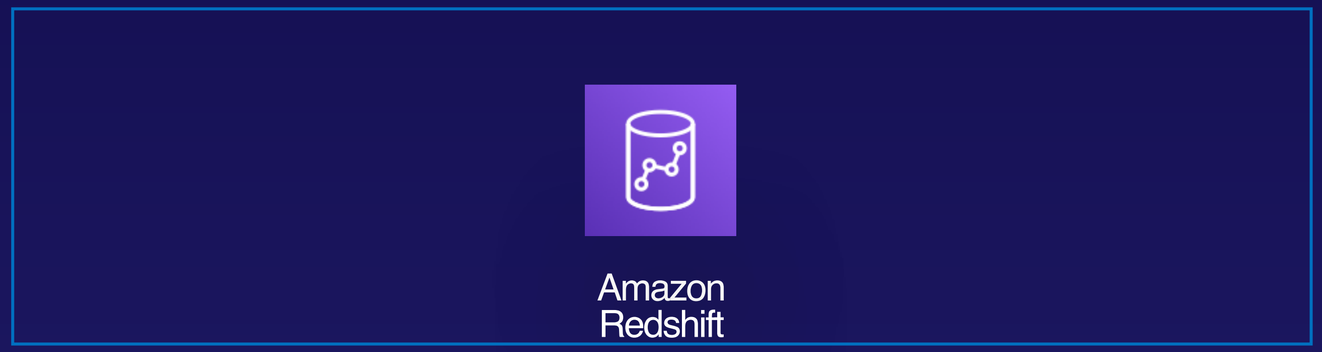


Amazon Redshift

AWS Glue

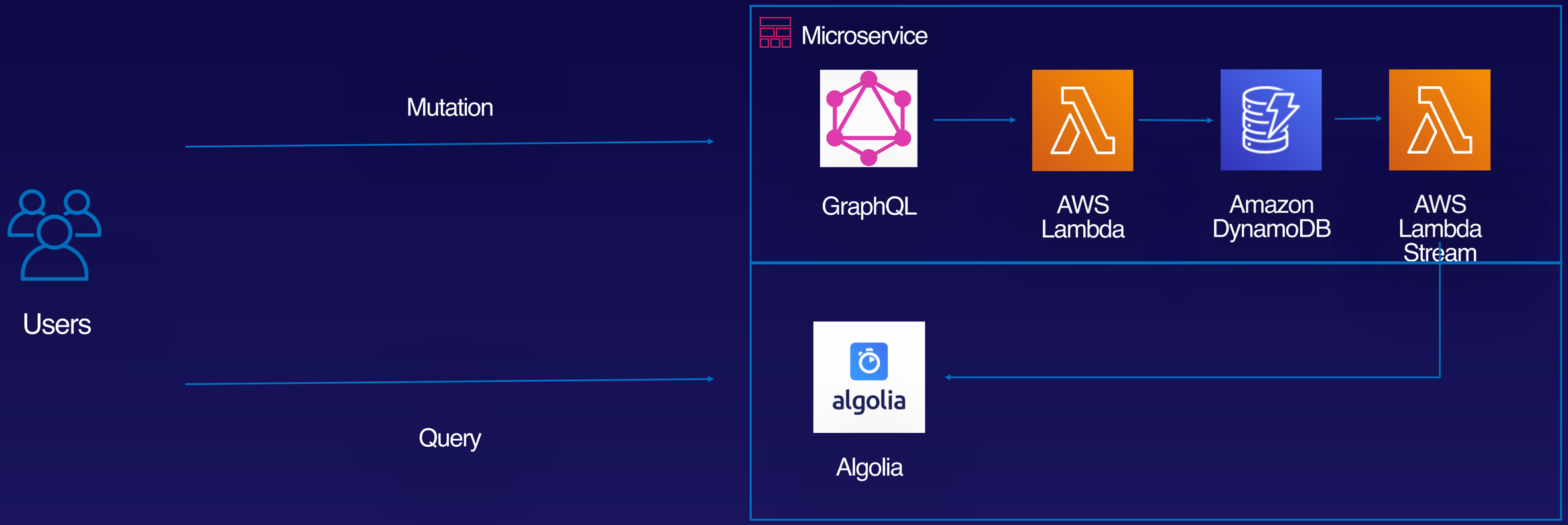
A CLOUD GURU


Cloudformation Stack
AWS Lambda
GraphQL API
Amazon API Gateway
AWS Lambda
Cloudformation Stack
Amazon DynamoDB
Amazon S3
Amazon SQS
Amazon SNS
algolia
Algolia
Cloudformation Stack
Parameter Store
Custom Resources
Cloudformation Stack
AWS Lambda
GraphQL API
Amazon API Gateway
AWS Lambda
Cloudformation Stack
Amazon DynamoDB
Amazon S3
Amazon SQS
Amazon SNS
algolia
Algolia
Cloudformation Stack
Parameter Store
Custom Resources
Cloudformation Stack
AWS Lambda
GraphQL API
Amazon API Gateway
AWS Lambda
Cloudformation Stack
Amazon DynamoDB
Amazon S3
Amazon SQS
Amazon SNS
algolia
Algolia
Cloudformation Stack
Parameter Store
Custom Resources


Amazon Redshift

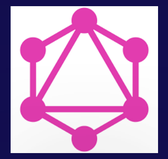

segment

Frontend To Backend Communication


Microservice
Mutation
Users
GraphQL
AWS Lambda
Amazon DynamoDB
AWS Lambda Stream
algolia
Algolia
Query

A CLOUD GURU

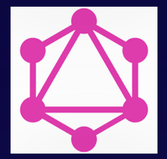
Users



GraphQL
Proxy

Proxy Layer

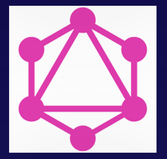
Microservice



GraphQL

AWS
Lambda

Microservice



GraphQL

AWS
Lambda

Microservice

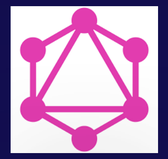
GraphQL

AWS
Lambda

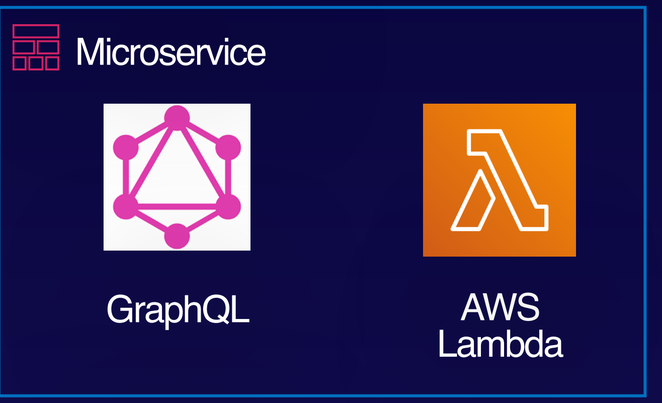
Microservices

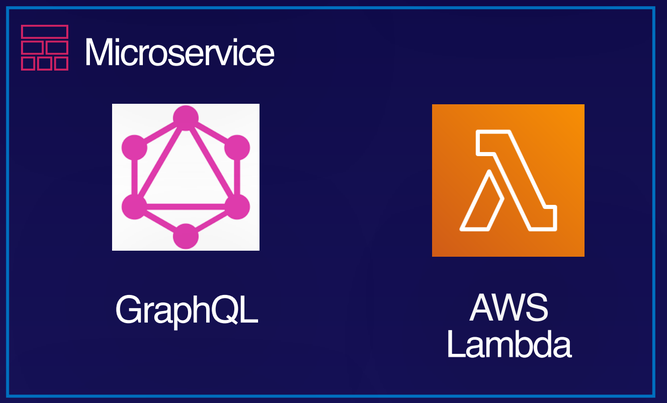
A CLOUD GURU

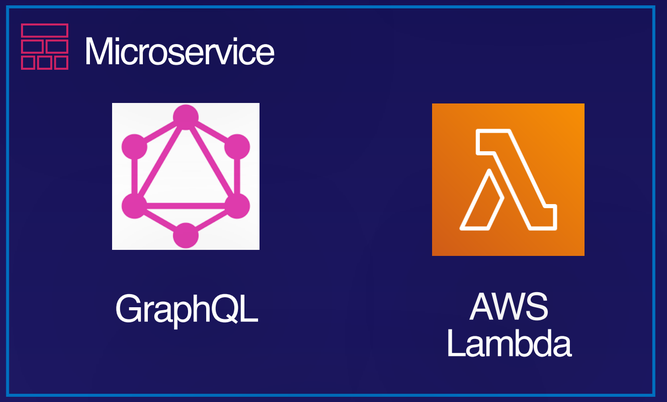
Users



GraphQL Proxy







Web & Mobile Clients

Isomorphic & API Calls

Proxy Layer

Microservices

The two biggest challenges with completely Serverless Architectures

Latency was solved through:

data replication

limiting the amount of lambda to lambda invokes

hitting 3rd party services directly when possible

avoiding VPC specific services

Limited Services was solved by:

using 3rd parties for functionality which would have been solved through an RDBMs

building application focusing on NoSQL stores

Thank You