AWS re:Invent

DOP340-R

Debugging serverless applications

Pongsan Sayampol
Solutions Architect
Amazon Web Services

AWS re:Invent

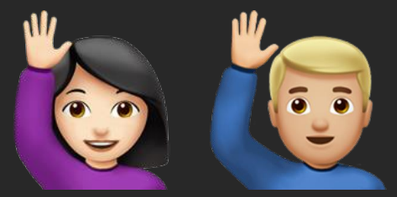
aws

Agenda

1. How should I develop serverless applications?
2. How do I improve developer productivity?
3. How do I trace and find root causes of errors?
4. How do I identify performance issues?

1. How should I develop serverless applications?

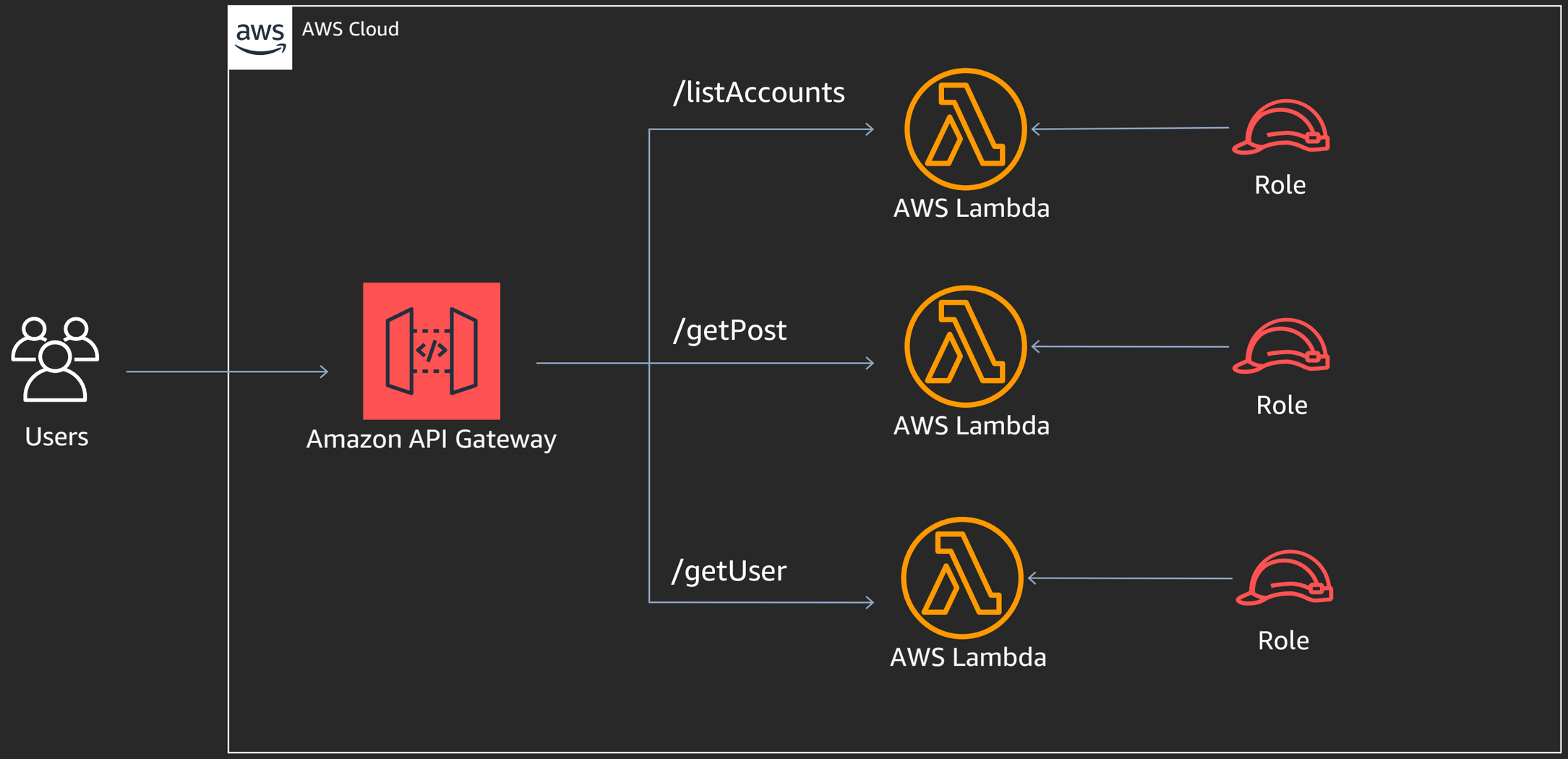
How are *you* developing serverless applications currently?



Challenges of serverless development

- AWS Lambda console restrictions
- Resources usually exist only during execution
- Can’t install agents for monitoring
- Latency in collecting, aggregating, and collating logs in real time
- Events need to be correlated across services
- Difficult to manage versions, deployment, and rollbacks

Application vs. functions



Serverless development stages

Write code directly in Lambda console

Zip packages and upload

Use serverless frameworks

Serverless frameworks

AWS

AWS Amplify

AWS SAM

Third-party

APEX

Claudia.js

Serverless Framework

Sparta

Zappa

AWS Serverless Application Model (SAM)

https://github.com/awslabs/serverless-application-model

- AWS CloudFormation extension, optimized for serverless
- Serverless resource types: Functions, APIs, tables, layers, and applications
- AWS CloudFormation supports (e.g., Amazon Kinesis)
- Supports parameters, mappings, outputs, global variables, intrinsic functions, and some import values
- YAML or JSON

SAM template

```
AWSTemplateFormatVersion: '2010-09-09’
Transform: AWS::Serverless-2016-10-31
Resources:
  CalculatorAdd:
    Type: AWS::Serverless::Function
    Properties:
      Handler: calculator
      Runtime: go1.x
      Events:
        Add:
          Type: Api
          Properties:
            Path: /
            Method: get
```

SAM template

```
AWSTemplateFormatVersion: '2010-09-09’
Transform: AWS::Serverless-2016-10-31
Resources:
  CalculatorAdd:
    Type: AWS::Serverless::Function
    Properties:
      Handler: calculator
      Runtime: go1.x
      Events:
        Add:
          Type: Api
          Properties:
            Path: /
            Method: get
```

Transformed template


```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Resources": {
    "AwesomeCalculatorRole": {
      "Type": "AWS::IAM::Role",
      "Properties": {
        "ManagedPolicyArns": [
          "arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecut
        ],
        "AssumeRolePolicyDocument": {
          "Version": "2012-10-17",
          "Statement": [
            {
              "Action": [
                "sts:AssumeRole"
              ],
              "Effect": "Allow",
              "Principal": {
                "Service": [
                  "lambda.amazonaws.com"
                ]
              }
            }
          ]
        }
      }
    },
    "AwesomeCalculatorAddPermissionTest": {
      "Type": "AWS::Lambda::Permission",
      "Properties": {
        "Action": "lambda:invokeFunction",
        "Principal": "apigateway.amazonaws.com",
        "FunctionName": {
          "Ref": "AwesomeCalculator"
        },
        "SourceArn": {
          "Fn::Sub": [
            "arn:aws:execute-api:${AWS::Region}:${AWS::AccountId}:${
            {
              "__Stage__": "*",
              "__ApiId__": {
                "Ref": "ServerlessRestApi"
              }
            }
          ]
        }
      }
    },
    "ServerlessRestApiProdStage": {
      "Type": "AWS::ApiGateway::Stage",
      "Properties": {
        "DeploymentId": {
          "Ref": "ServerlessRestApiDeployment9bcc4decb0"
        },
        "RestApiId": {
          "Ref": "ServerlessRestApi"
        },
        "StageName": "Prod"
      }
    },
    "AwesomeCalculator": {
      "Type": "AWS::Lambda::Function",
      "Properties": {
        "Code": {
          "S3Bucket": "sydney-summit-sam",
          "S3Key": "001f1d29dc238636e2b94fe7a71aa00c"
        },
        "Handler": "index.handler",
        "Role": {
          "Fn::GetAtt": [
            "AwesomeCalculatorRole",
            "Arn"
          ]
        },
        "Runtime": "nodejs6.10",
        "Tags": [
          {
            "Value": "SAM",
            "Key": "lambda:createdBy"
          }
        ]
      }
    },
    "ServerlessRestApi": {
      "Type": "AWS::ApiGateway::RestApi",
      "Properties": {
        "Body": {
          "info": {
            "version": "1.0",
            "title": {
              "Ref": "AWS::StackName"
            }
          },
          "paths": {
            "/": {
              "get": {
                "x-amazon-apigateway-integration": {
                  "httpMethod": "POST",
                  "type": "aws_proxy",
                  "uri": {
                    "Fn::Sub": "arn:aws:apigateway:${AWS::Region}:l
                  }
                },
                "responses": {}
              }
            }
          },
          "swagger": "2.0"
        }
      }
    },
    "AwesomeCalculatorAddPermissionProd": {
      "Type": "AWS::Lambda::Permission",
      "Properties": {
        "Action": "lambda:invokeFunction",
        "Principal": "apigateway.amazonaws.com",
        "FunctionName": {
          "Ref": "AwesomeCalculator"
        },
        "SourceArn": {
          "Fn::Sub": [
            "arn:aws:execute-api:${AWS::Region}:${AWS::AccountId}:$
            {
              "__Stage__": "Prod",
              "__ApiId__": {
                "Ref": "ServerlessRestApi"
              }
            }
          ]
        }
      }
    },
    "ServerlessRestApiDeployment9bcc4decb0": {
      "Type": "AWS::ApiGateway::Deployment",
      "Properties": {
        "RestApiId": {
          "Ref": "ServerlessRestApi"
        },
        "Description": "RestApi deployment id: 9bcc4decb055b4dac489
        "StageName": "Stage"
      }
    }
  }
}
```

SAM template

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  CalculatorAdd:
    Type: AWS::Serverless::Function
    Properties:
      Handler: calculator
      Runtime: go1.x
      Events:
        Add:
          Type: Api
          Properties:
            Path: /
            Method: get
```

Lambda function

SAM template

```
AWSTemplateFormatVersion: '2010-09-09’
Transform: AWS::Serverless-2016-10-31
Resources:
  CalculatorAdd:
    Type: AWS::Serverless::Function
    Properties:
      Handler: calculator
      Runtime: go1.x
      Events:
        Add:
          Type: Api
          Properties:
            Path: /
            Method: get
```

API Gateway

SAM commands

Package

- Creates a deployment package (.zip file)
- Uploads deployment package to an Amazon S3 bucket
- Adds a CodeUri property with S3 URI

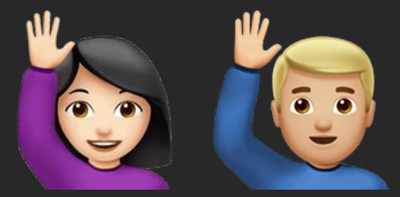
Deploy

- Calls AWS CloudFormation "*CreateChangeSet*" API
- Calls AWS CloudFormation "*ExecuteChangeSet*" API

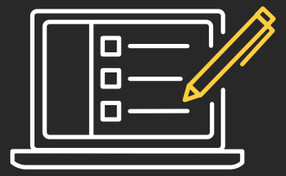
Demo: AWS SAM templates

2. How do I improve developer productivity?

What do developers want in their IDE?

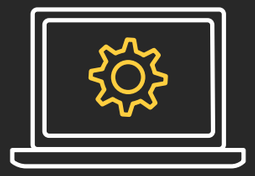


Typical developers' needs



Lightning fast, no-frills code editor

Dark background



Easy debugging and testing

Beautiful code linting

CI/CD integration

Full-featured, easily consumable SDK

AWS developer tools at a glance

IDE integration

AWS Toolkit

AWS Cloud9

PC IJ

Microsoft Visual Studio

Programmable SDK

AWS SDK

Microsoft Visual Studio

AWS CDK

Command line tools

AWS CLI

AWS SAM

AWS Amplify

DevOps & automation

AWS code*

AWS App Mesh

Monitoring & tracing

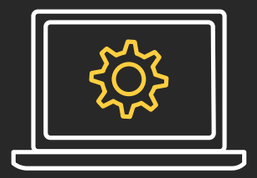
Amazon CloudWatch

AWS X-Ray

AWS Toolkit for Visual Studio Code

Integrated experience targeting development of serverless applications in Node.js, Python, or .NET

“Getting started” project template



Step-through debugging

Deployment from the IDE

Demo: AWS Toolkit for Visual Studio Code

AWS re:Invent

aws

AWS SAM Command Line Interface (CLI)

https://aws.amazon.com/serverless/sam

- CLI tool for local development, debugging, testing, deploying, and monitoring of serverless applications
- Supports API Gateway “proxy-style” and Lambda service API testing
- Response object and function logs available on your local machine
- Uses open-source docker-lambda images to mimic Lambda’s execution environment
- Can tail production logs from CloudWatch logs
- Can help you build in native dependencies

AWS SAM CLI

- `sam build` – install dependencies
- `sam deploy` – deploy AWS SAM application with AWS CloudFormation
- `sam init` - initialize a serverless application
- `sam local`:
  - `generate-event` – generate sample payloads from event sources
  - `invoke` – invokes a Lambda function once
  - `start-api` – set up local endpoint to test your API
  - `start-lambda` – start a local endpoint to test your local Lambda
- `sam logs` – fetch logs for a function
- `sam package` – package an AWS SAM application

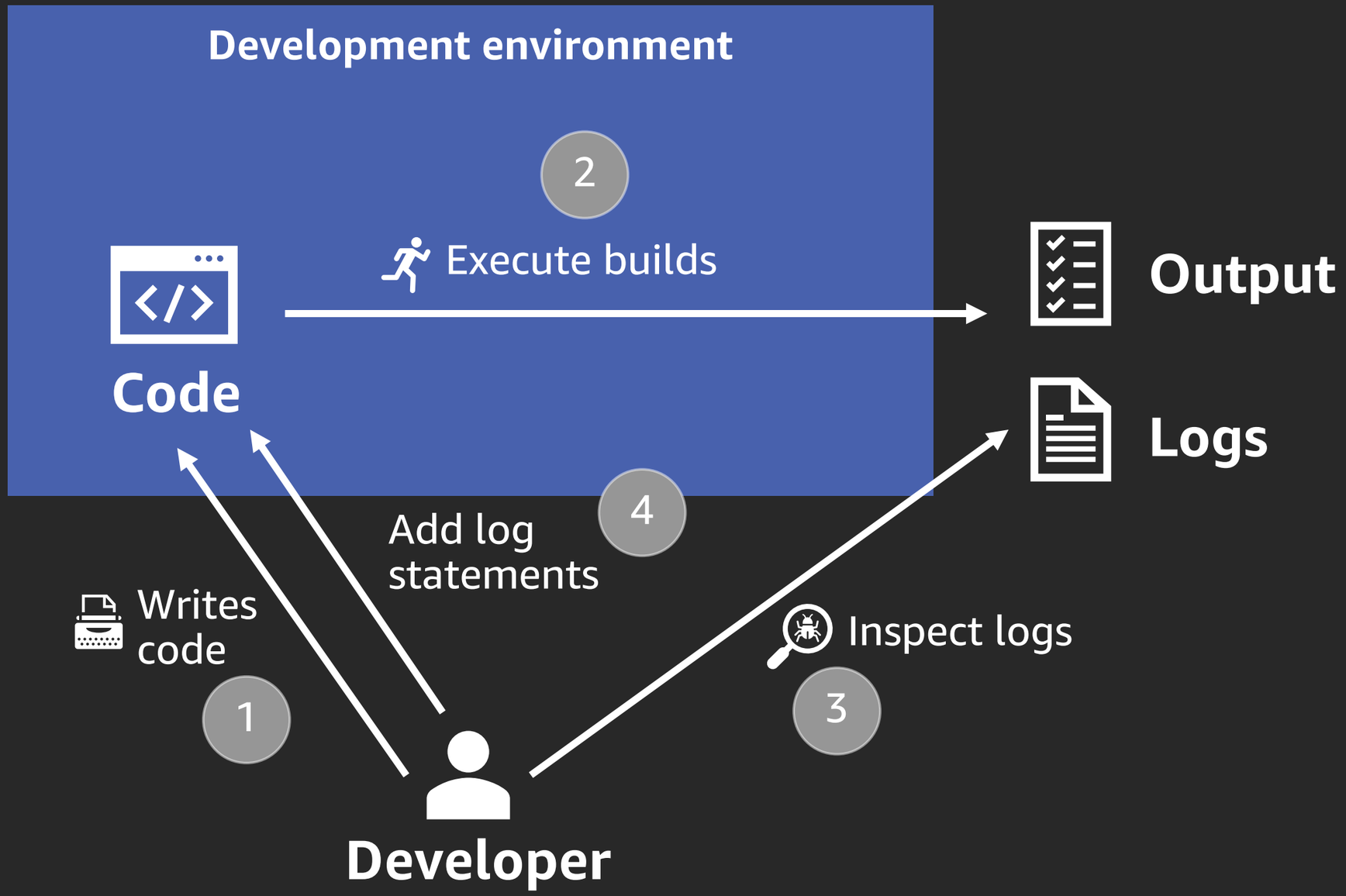
Demo: AWS SAM CLI

3. How do I trace and find root causes of errors?

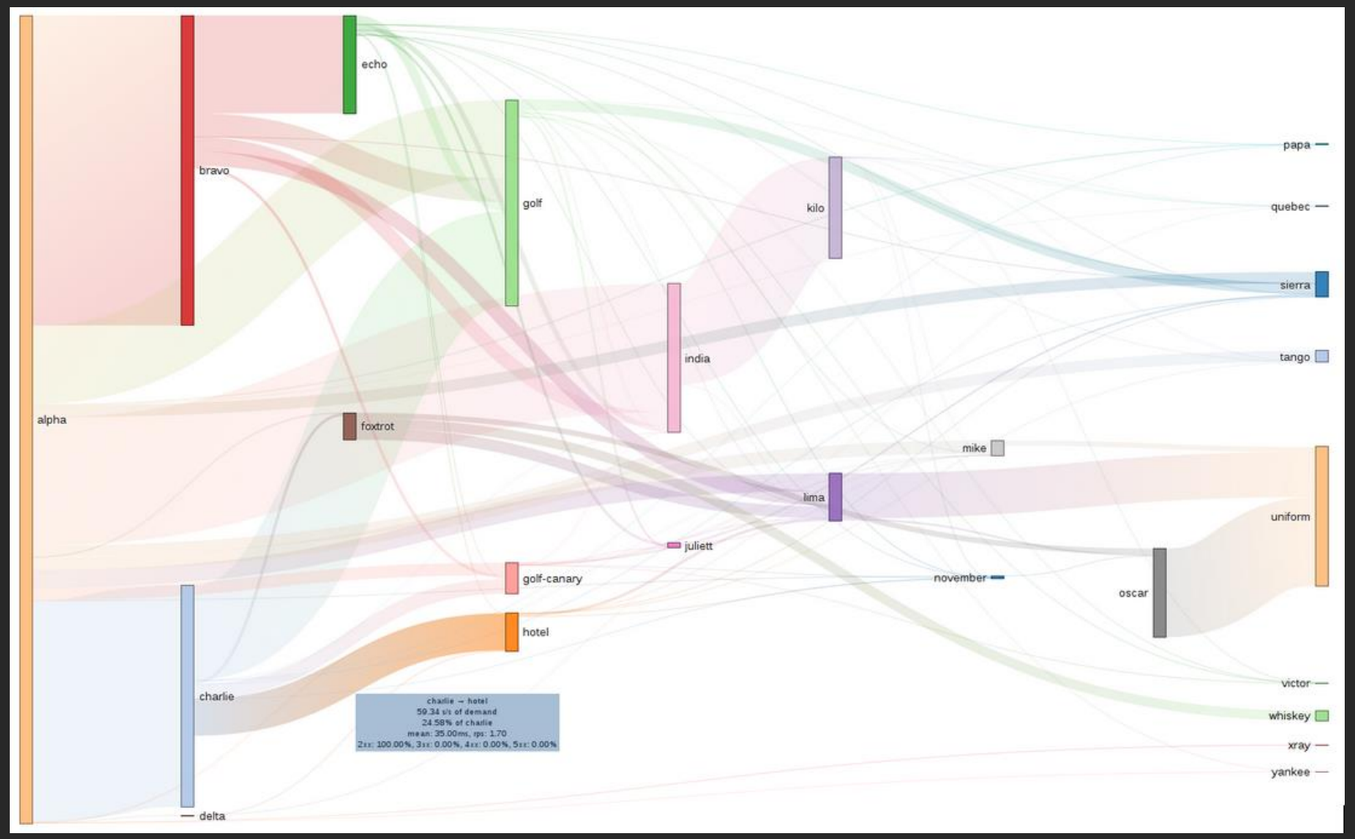
"When the going gets tough, the tough gets debugging."

Any developer

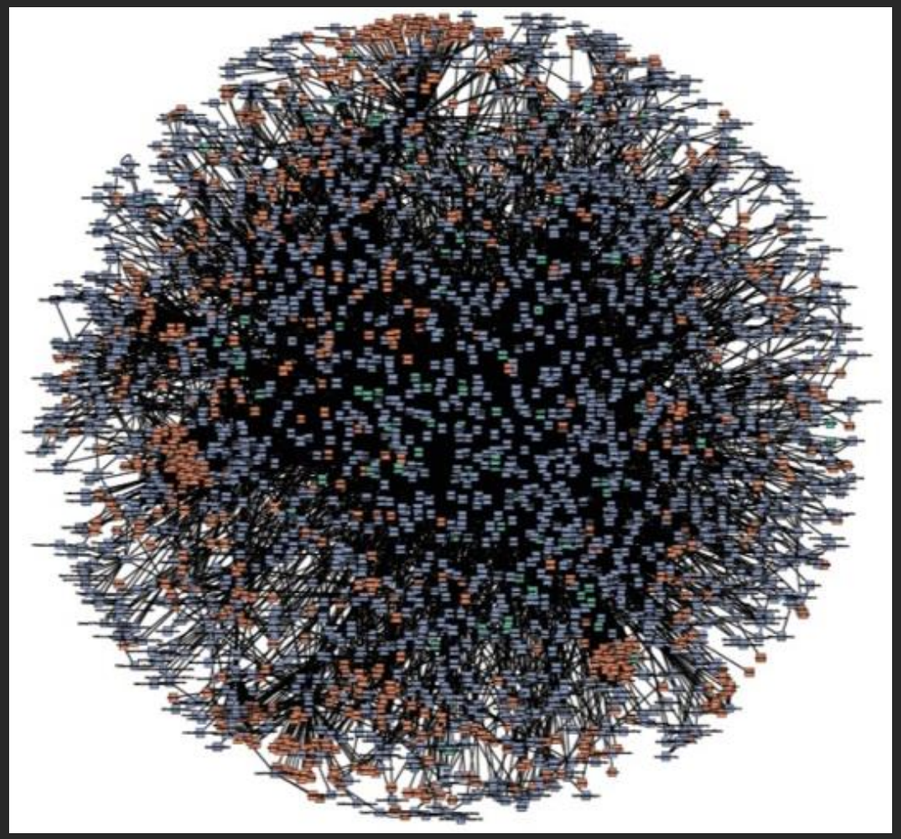
Traditional debugging



Reality of microservices environment



Request flow



Dependency visualization

Source: https://medium.com/netflix-techblog/a-microscope-on-microservices-923b906103f4

Source: https://medium.com/@LachlanEvenson/practicing-microservices-ac43c8b3b712

AWS X-Ray

Analyze and debug issues quickly

End-to-end view of individual services

Visualize service calls

Identify performance bottlenecks

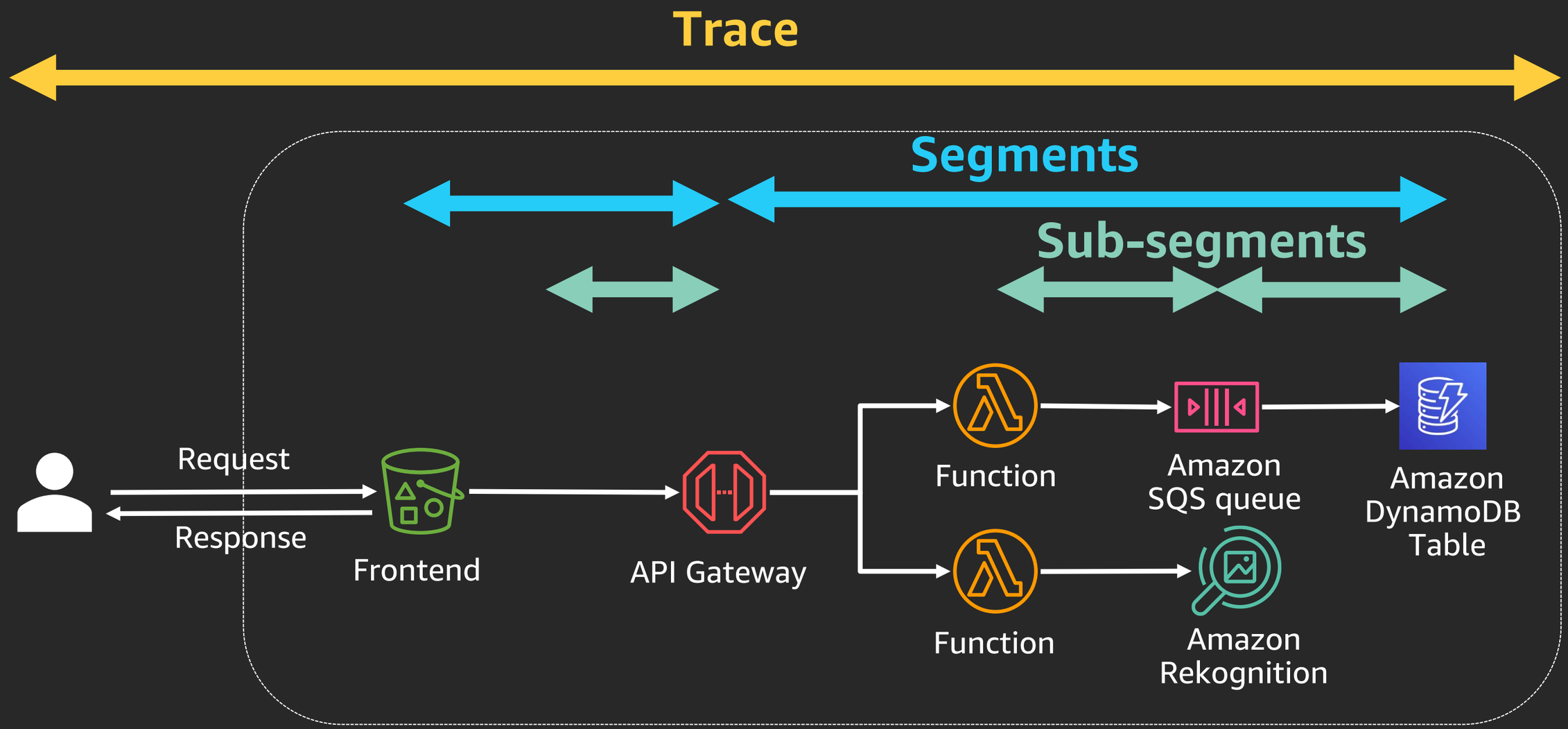
Pinpoint issues

Identify errors

AWS X-Ray + Lambda

- X-Ray agent is natively built into Lambda
- Identify initialization and cold starts in Lambda
- Pinpoint issues in downstream services called from your Lambda function
- Happens with low latency in real time; can see traces in seconds

AWS X-Ray concepts



X-Ray SDK

Available for Java, .NET, .NET Core, Ruby, Python, Go, and Node.js

Adds filters to automatically capture metadata for calls to

- AWS services using the AWS SDK
- Non-AWS services over HTTP and HTTPS (third-party APIs)
- Databases (MySQL, PostgreSQL, and Amazon DynamoDB)
- Queues (Amazon SQS)

Enables you to get started quickly without having to manually instrument your application code to log metadata about requests

App instrumentation

```
//Add aws-xray-sdk package to package.json

var XRay = require('aws-xray-sdk');
var AWS = captureAWS(require('aws-sdk'));
…
XRay.config([XRay.plugins.EC2]);
XRay.captureHTTPs(http);
XRay.setDefaultName('myfrontend-dev');
…
app.use(XRay.express.openSegment());

app.get('/', function(req, res)
…
app.get('/blog', function(req, res)
…

app.use(XRay.express.closeSegment());
```

Add business data

```
//Example showing how to add business data to traces
app.post('/signup', function(req, res) {
        var item = {
            'email': {'S': req.body.email},
            'name': {'S': req.body.name},
            'preview': {'S': req.body.previewAccess},
            'theme': {'S': req.body.theme}
        };

        var seg = XRay.getSegment();
        seg.addAnnotation('email', req.body.email);
        seg.addAnnotation('theme', req.body.theme);
        seg.addAnnotation('previewAccess', req.body.previewAccess);

        //Store item to DB
        //Send sign-up notification to user
}
```

X-Ray pricing

Free tier

- The first 100,000 traces recorded are free
- The first 1,000,000 traces retrieved or scanned are free

Additional charges

- Beyond the free tier, traces recorded cost $5.00 per million traces
- Beyond the free tier, traces retrieved or scanned cost $0.50 per million traces

Demo: Using AWS X-Ray for tracing

4. How do I Identify performance issues?

Demo: AWS X-Ray – Visualize for Performance

AWS re:Invent

aws

Amazon CloudWatch ServiceLens

- Fully managed observability solution
- Consolidates logs, metrics, and traces in a single place
- Visualise and analyse the health, performance, and availability of applications
- Easily isolate endpoints and resources that are experiencing issues
- Correlate metrics, logs, and application traces
- Understand the relationships and dependencies within your applications to quickly isolate faults

Demo – CloudWatch ServiceLens

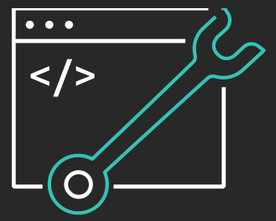
Hands-on

AWS re:Invent

aws

Learn DevOps with AWS Training and Certification

Resources created by the experts at AWS to propel your organization and career forward



Take free digital training to learn best practices for developing, deploying, and maintaining applications

Classroom offerings, like DevOps Engineering on AWS, feature AWS expert instructors and hands-on activities

Validate expertise with the **AWS Certified DevOps Engineer - Professional** or **AWS Certified Developer - Associate** exams

Visit aws.amazon.com/training/path-developing/

aws training and certification

Thank you!

Loh Yiang Meng

ymloh@amazon.com

AWS re:Invent

aws

Please complete the session survey in the mobile app.