AWS SERVERLESS COMUTING

What is Serverless Computing?

Serverless Means

AWS Serverless Services – Compute

AWS Lambda

AWS Lambda is an event-driven, pay-as-you-go compute service that lets you run code without provisioning or managing servers.

AWS Fargate

AWS Fargate is a serverless compute engine that works with Amazon Elastic Container Service (ECS) and Amazon Elastic Kubernetes Service (EKS).

AWS Serverless Services – App Integration

Amazon EventBridge

Amazon EventBridge is a serverless event bus that lets you build event-driven applications at scale across AWS and existing systems.

AWS Step Functions

AWS Step Functions is a visual workflow orchestrator that makes it easy to sequence multiple AWS services into business-critical applications.

Amazon SQS

Amazon Simple Query Service (SQS) is a message queuing service enabling you to decouple and scale microservices, distributed systems, and serverless applications.

Amazon SNS

Amazon Simple Notification Service (SNS) is a fully managed messaging service for both application-to-application (A2A) and application-to-person (A2P) communication.

Amazon API Gateway

Amazon API Gateway is a fully managed service that makes it easy to create and publish APIs at any scale.

AWS AppSync

AWS AppSync is a fully managed service that accelerates application development with scalable GraphQL APIs.

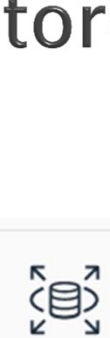
AWS Serverless Services – Data Store

Amazon S3

Amazon Simple Storage Service (Amazon S3) is an object storage service designed to store and protect any amount of data.

Amazon DynamoDB

Amazon DynamoDB is a key-value and document database service, delivering single-digit millisecond performance at any scale.



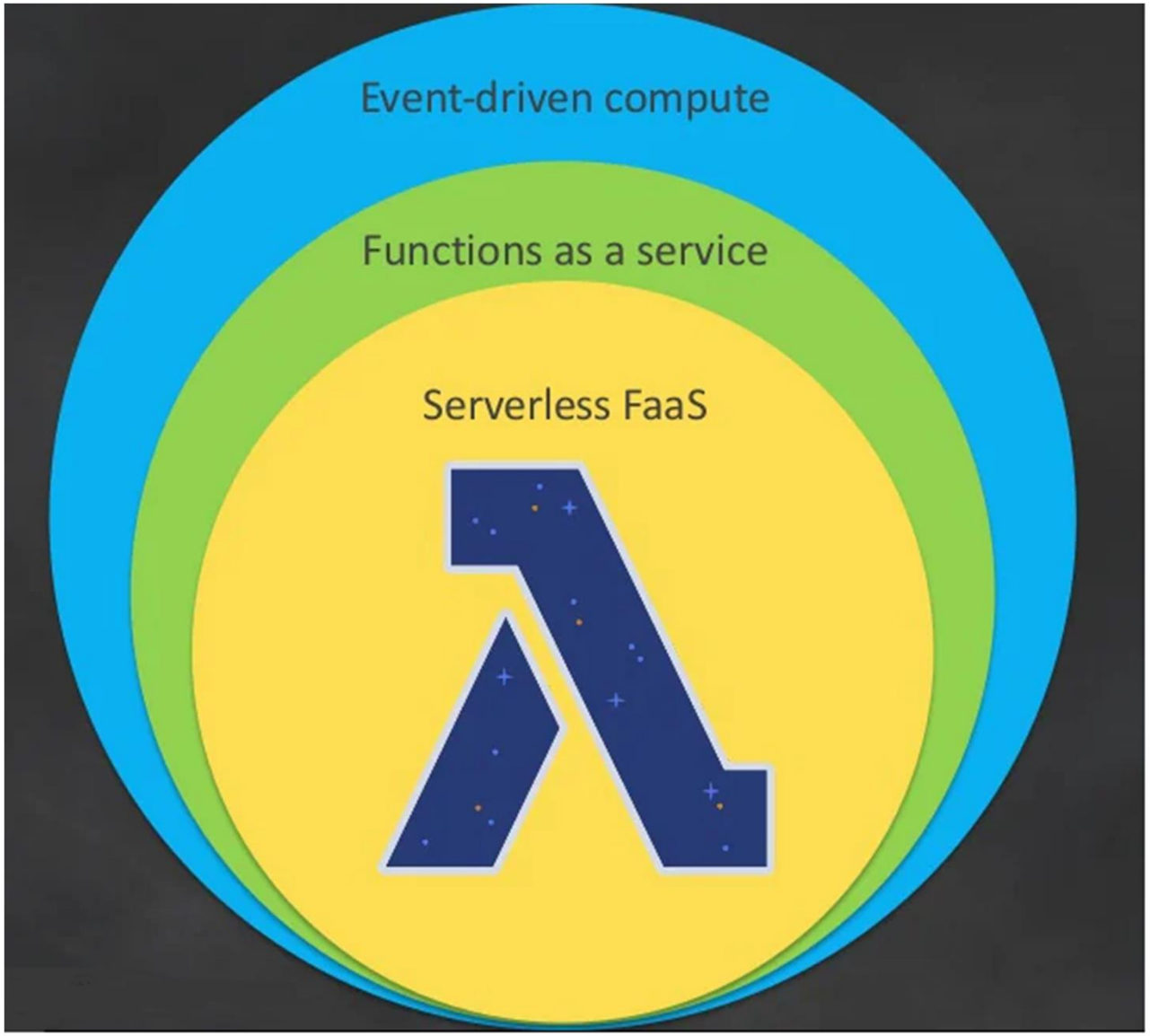
Amazon RDS Proxy

Amazon RDS Proxy is a managed database proxy for Amazon Relational Database Service (RDS) that makes applications more scalable and secure.

Amazon Aurora Serverless

Amazon Aurora Serverless is a MySQL and PostgreSQL-compatible relational database that automatically scales capacity based on your application's needs.

Lambda


Event-driven compute
Functions as a service
Serverless FaaS

AWS Lambda Functions

Stateless, trigger-based code execution

1

No Infrastructure to manage

Focus on business logic, not infrastructure. You upload code; AWS Lambda handles everything else.

2

High performance at any scale; Cost-effective and efficient

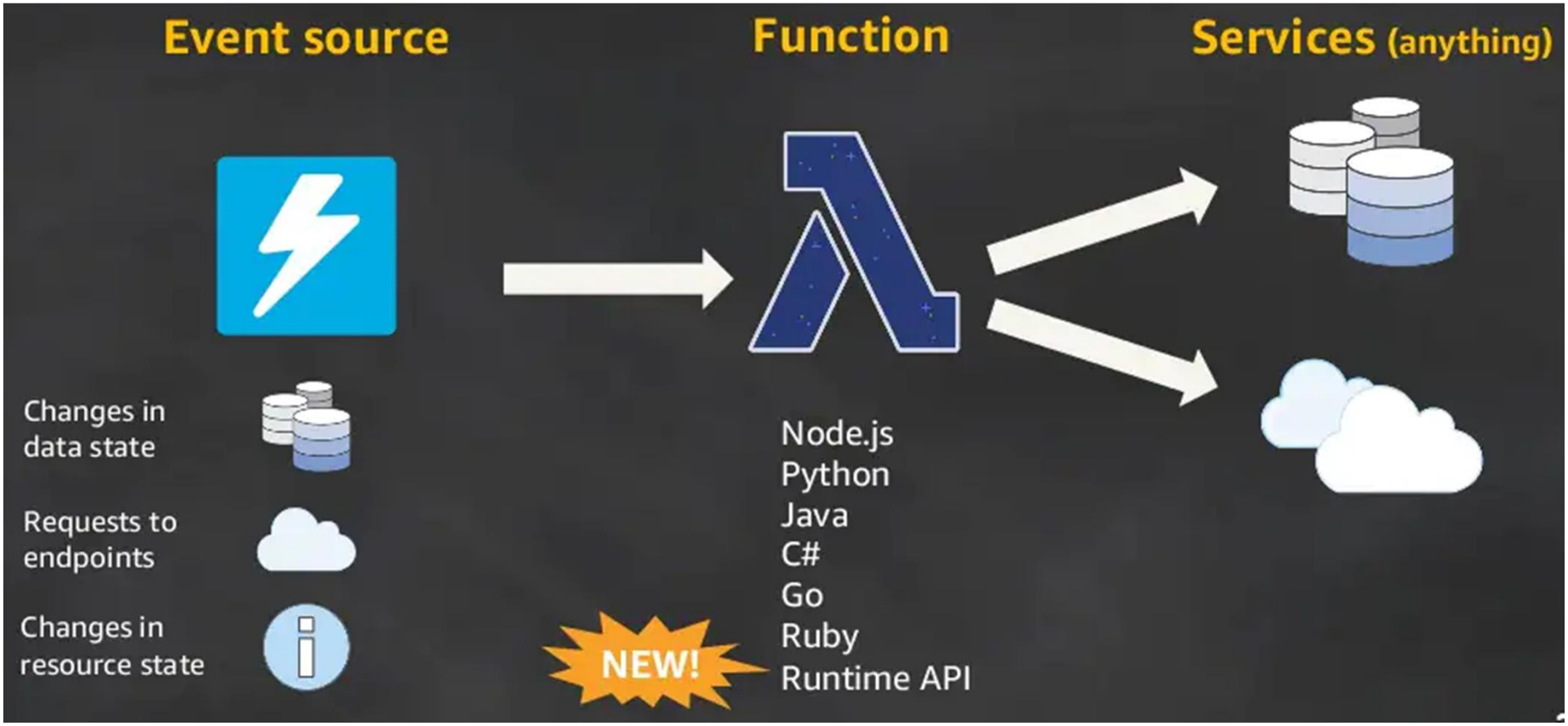
Pay only for what you use: Lambda automatically matches capacity to your request rate. Purchase compute in 100ms increments.

3

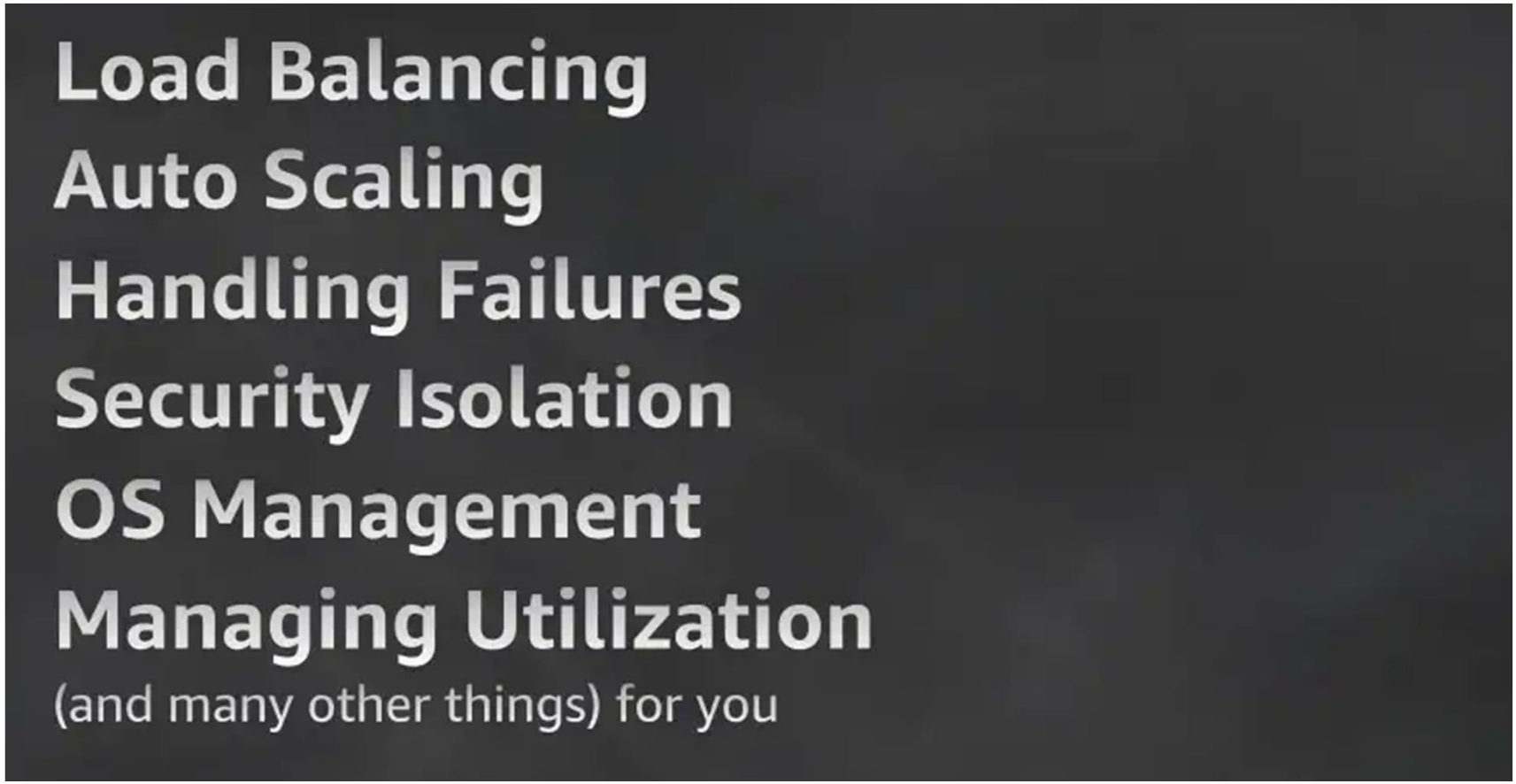
Bring Your Own Code

Run code in a choice of standard languages. Use threads, processes, files, and shell scripts normally.

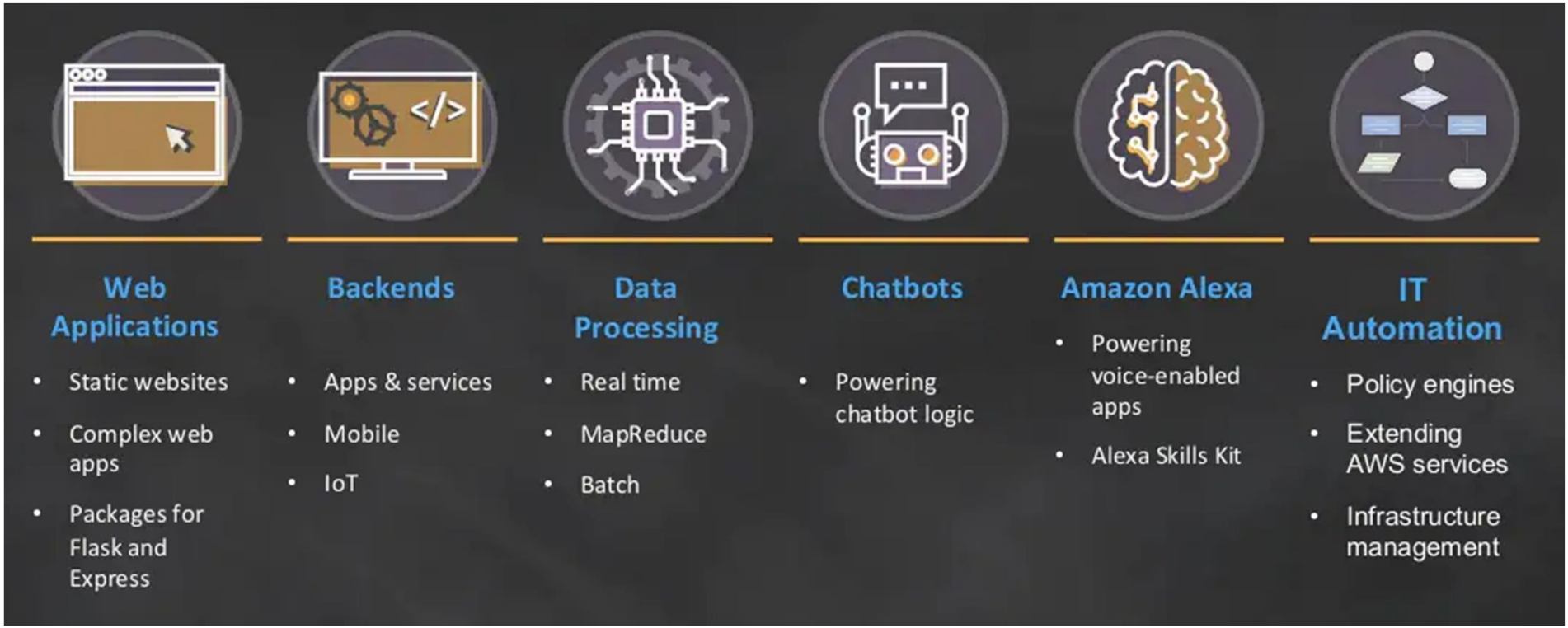
AWS Lambda Functions



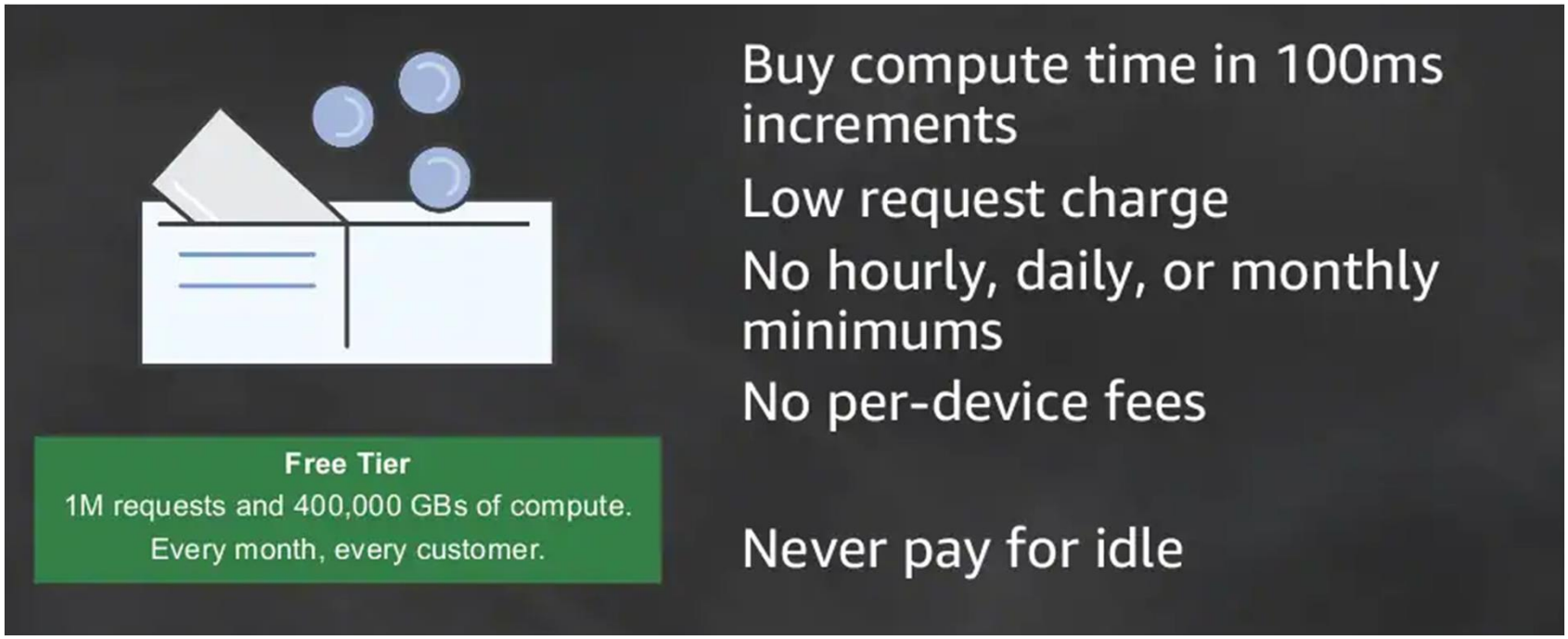
AWS Lambda Handles



Lambda Usecases



Lambda Pricing



Anatomy of a Lambda Function

Handler() function
Function to be executed upon invocation

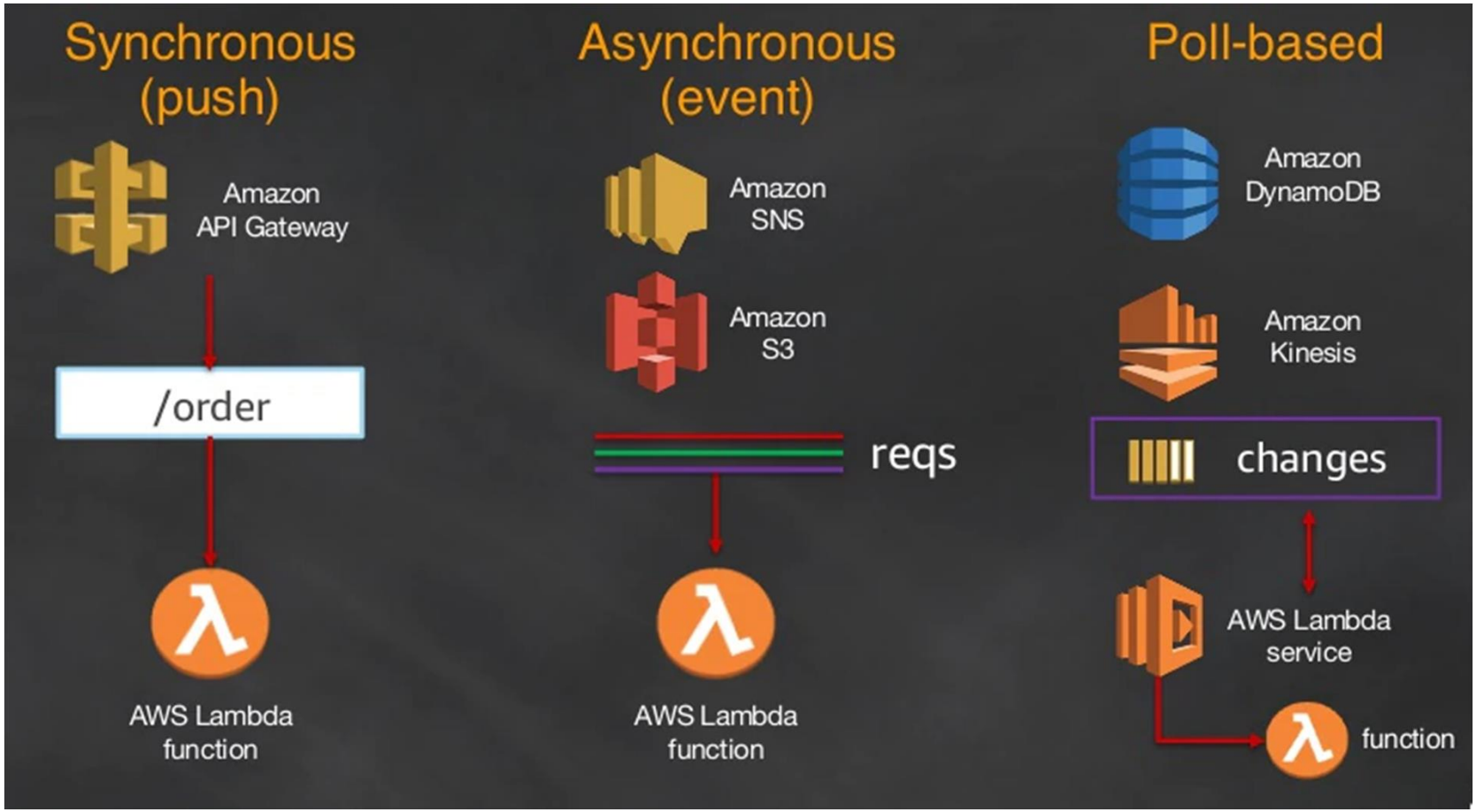
Event object
Data sent during Lambda function Invocation

Context object
Methods available to interact with runtime information (request ID, log group, more)

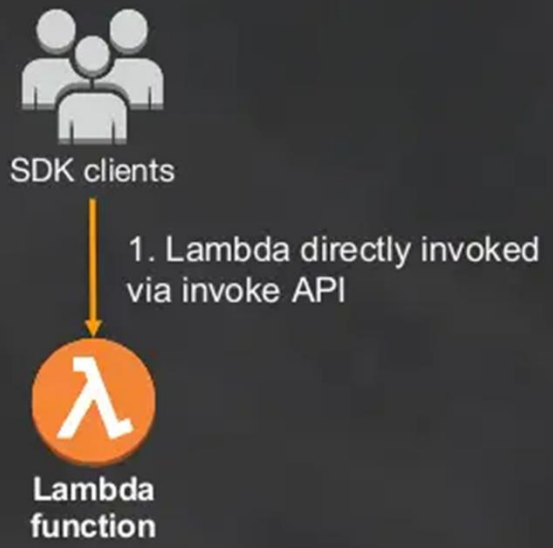
```
public String handleRequest(Book book, Context context) {
    saveBook(book);

    return book.getName() + " saved!";
}
```

Lambda Execution Model



Lambda API



API provided by the Lambda service

Used by all other services that invoke Lambda across all models

Supports sync and async

Can pass any event payload structure you want

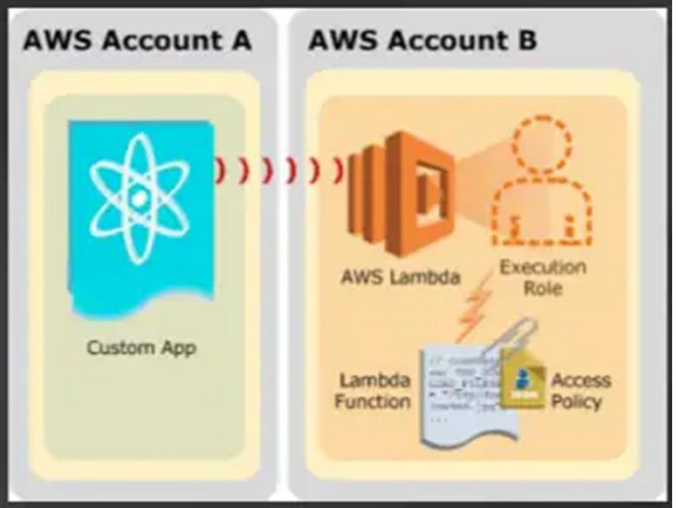
Client included in every SDK

Lambda Permission Model

Fine grained security controls for both execution and invocation:

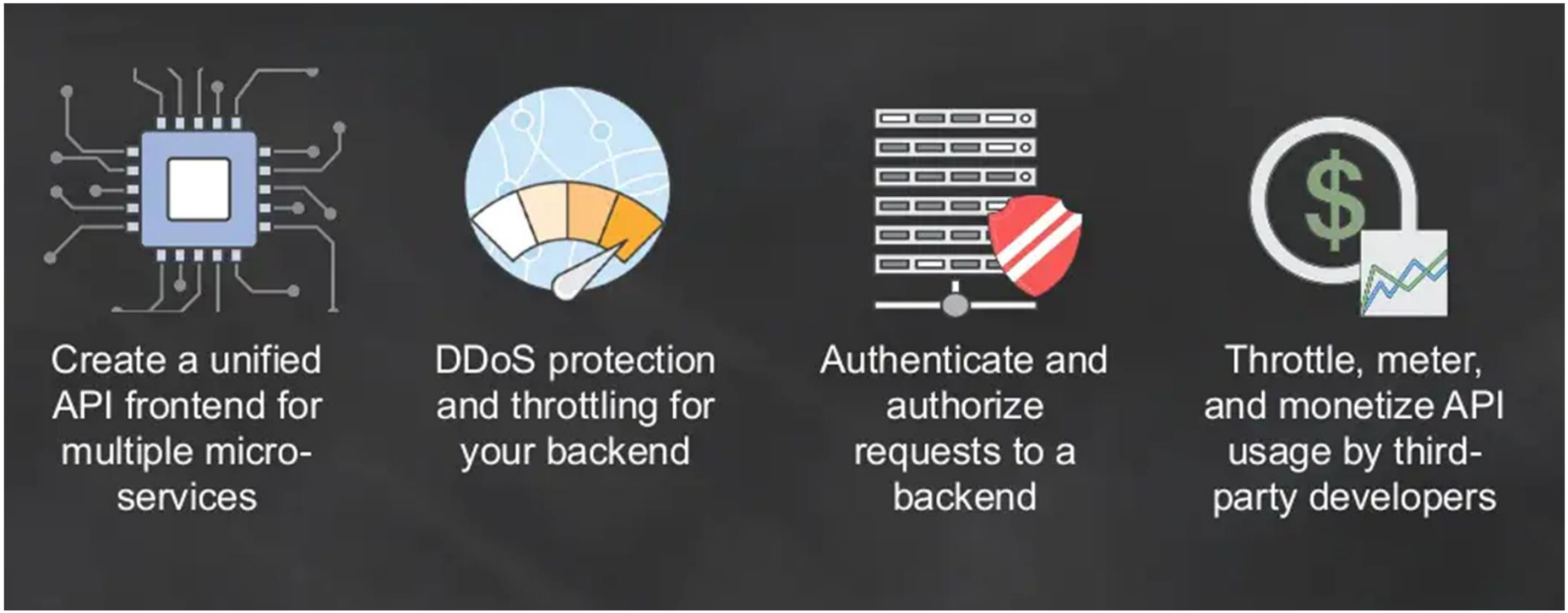
- **Execution policies:**
- Define what AWS resources/API calls can this function access via IAM
- Used in streaming invocations
- E.g. “Lambda function A can read from DynamoDB table users”
- **Function policies:**
- Used for sync and async invocations
- E.g. “Actions on bucket X can invoke Lambda function Z"
- Resource policies allow for cross account access

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "*"
        }
    ]
}
```

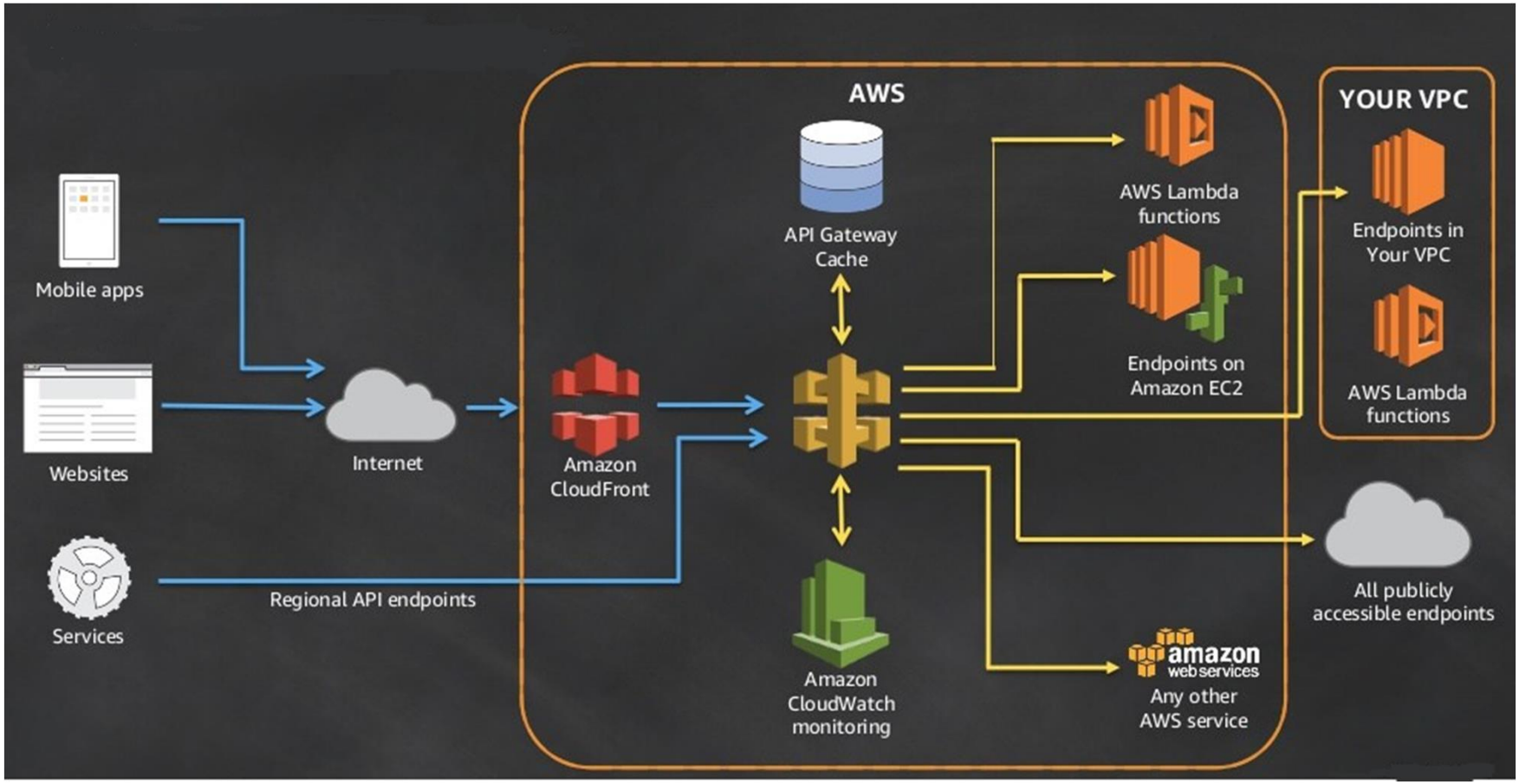


API Gateway

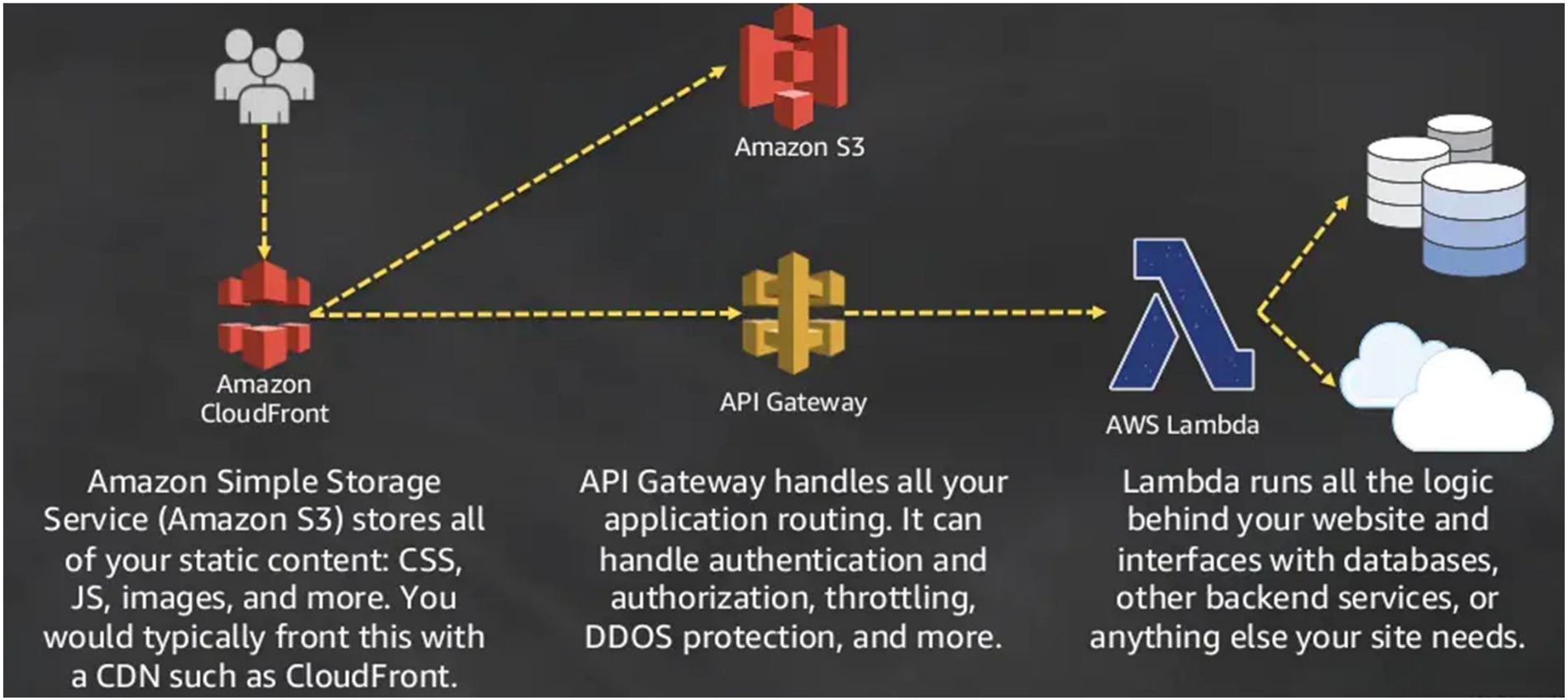
AWS API Gateway



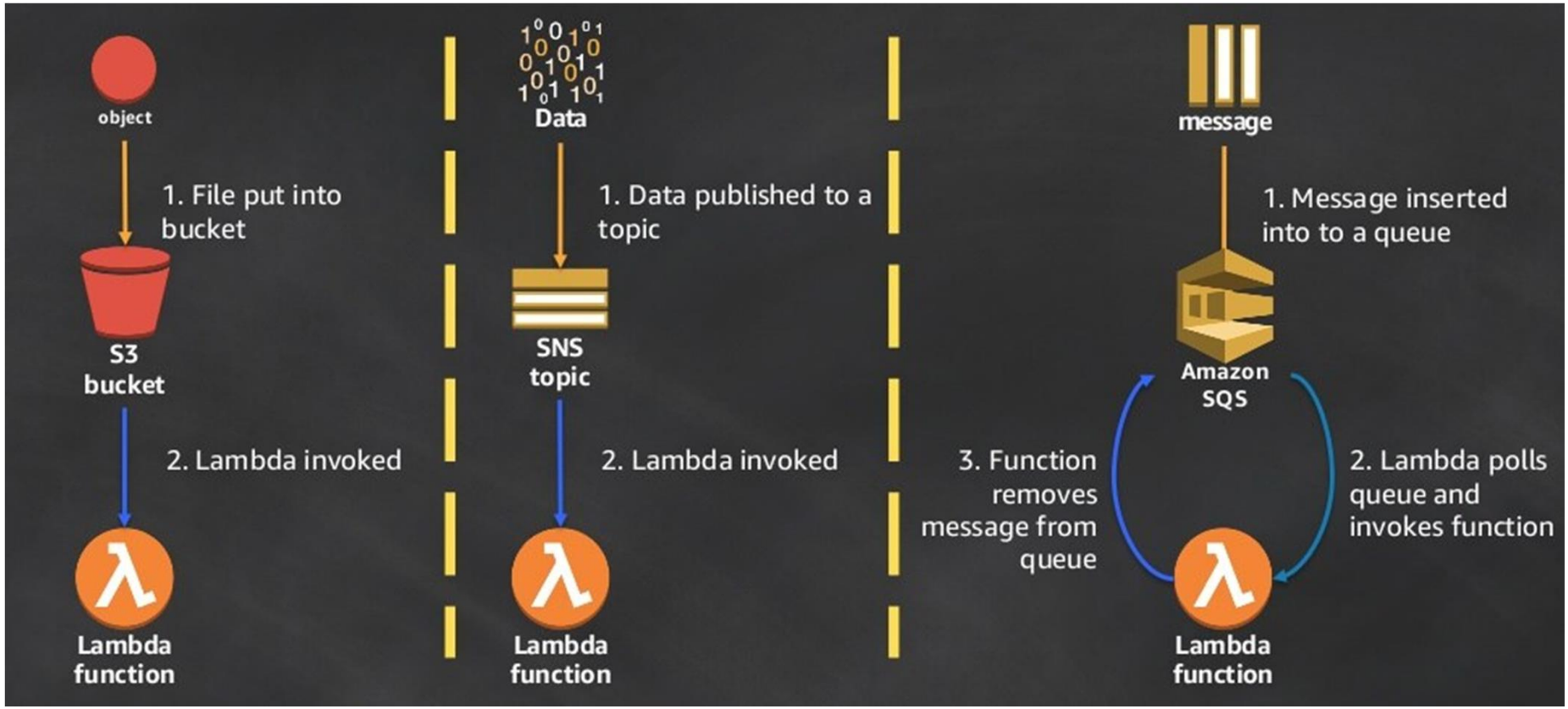
AWS API Gateway



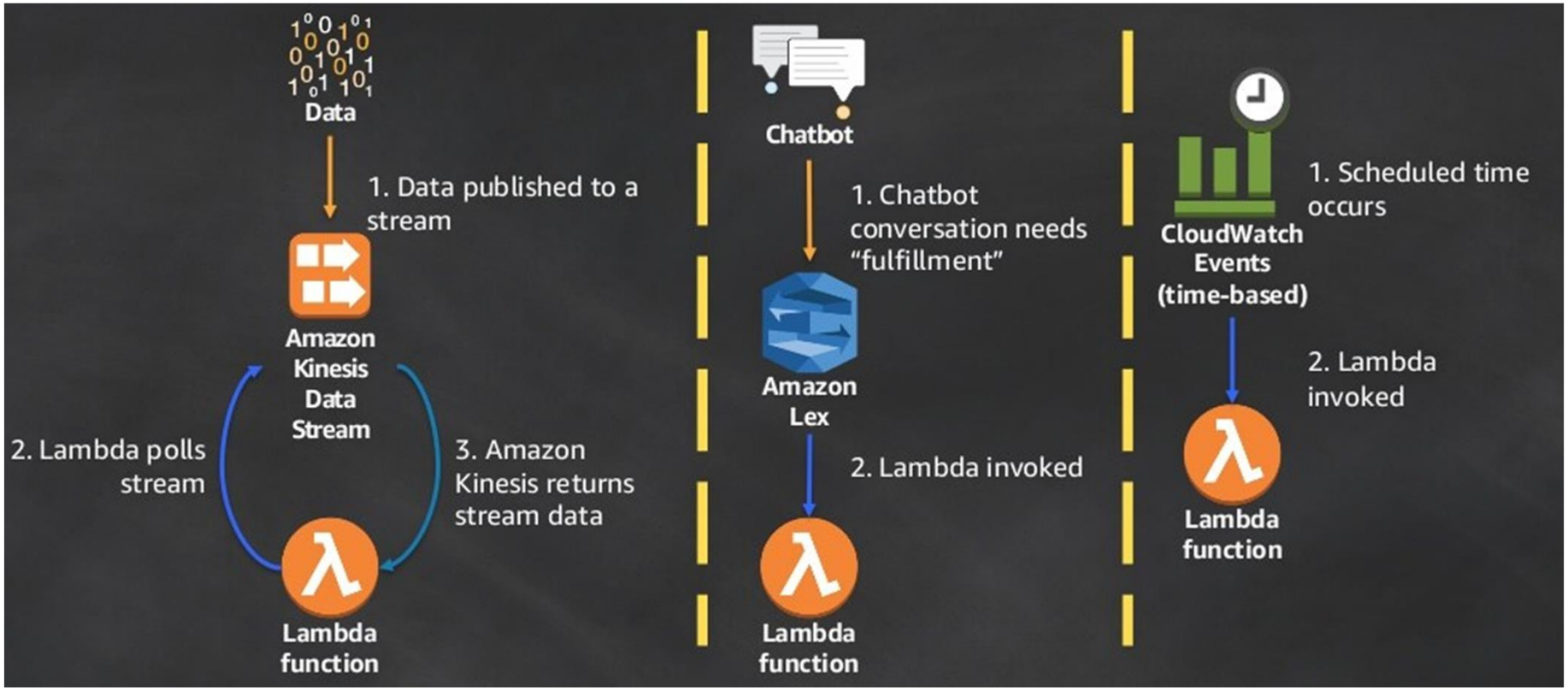
Serverless Web Application with API Gateway



Serverless Architectures

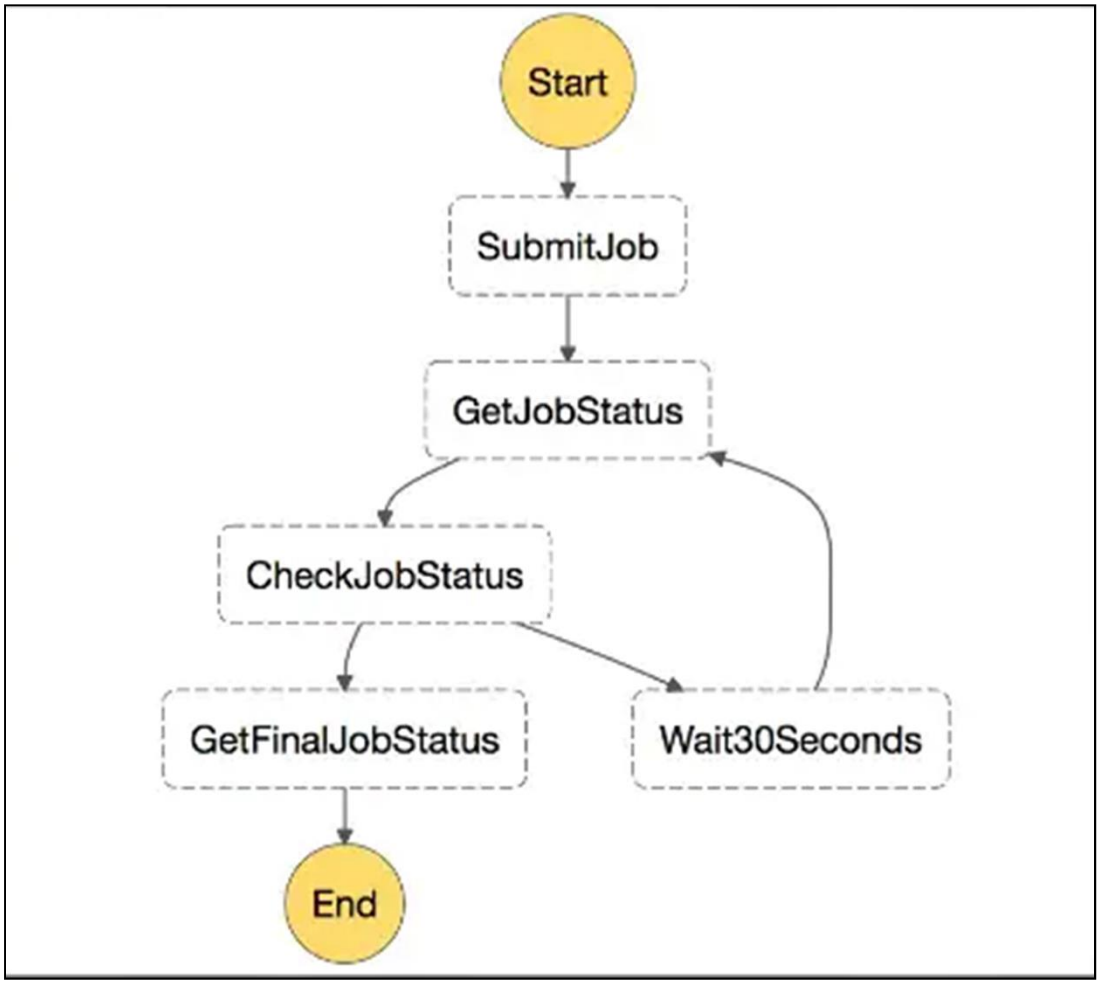


Serverless Architectures



Step Functions

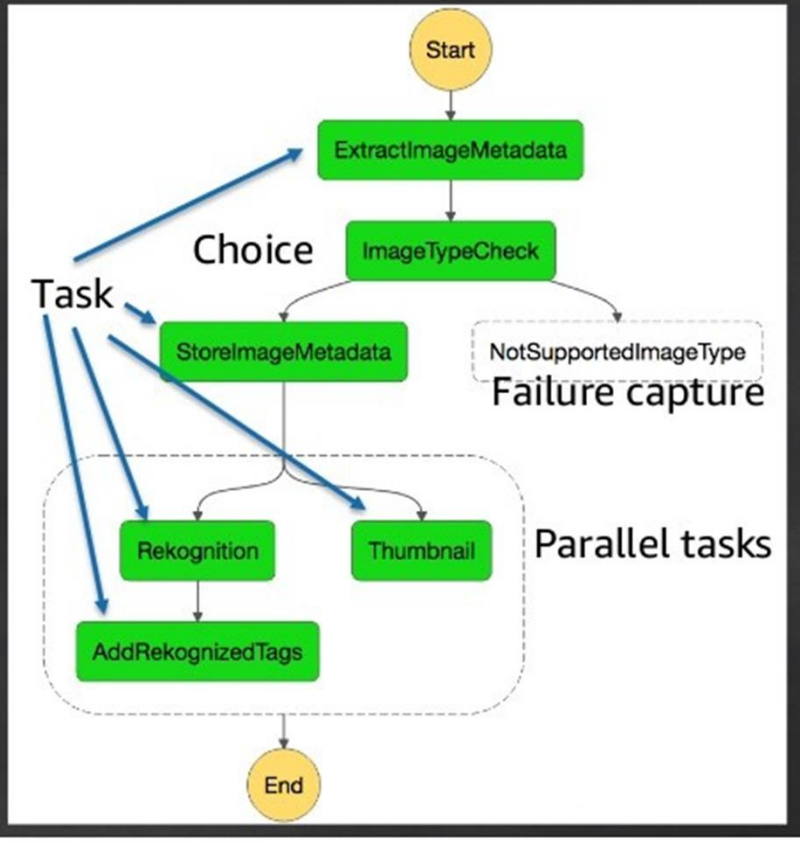
Keep Orchestration Out of Code



AWS Step Functions

"Serverless" workflow management with zero administration

- Makes it easy to coordinate the components of distributed applications and microservices using visual workflows
- Automatically triggers and tracks each step and retries when there are errors, so your application executes in order and as expected
- Logs the state of each step, so when things do go wrong, you can diagnose and debug problems quickly



Thank You!