AWS Serverless

An Introduction to AWS Serverless

Lovro Gašperin, 24.10.2024

Agenda

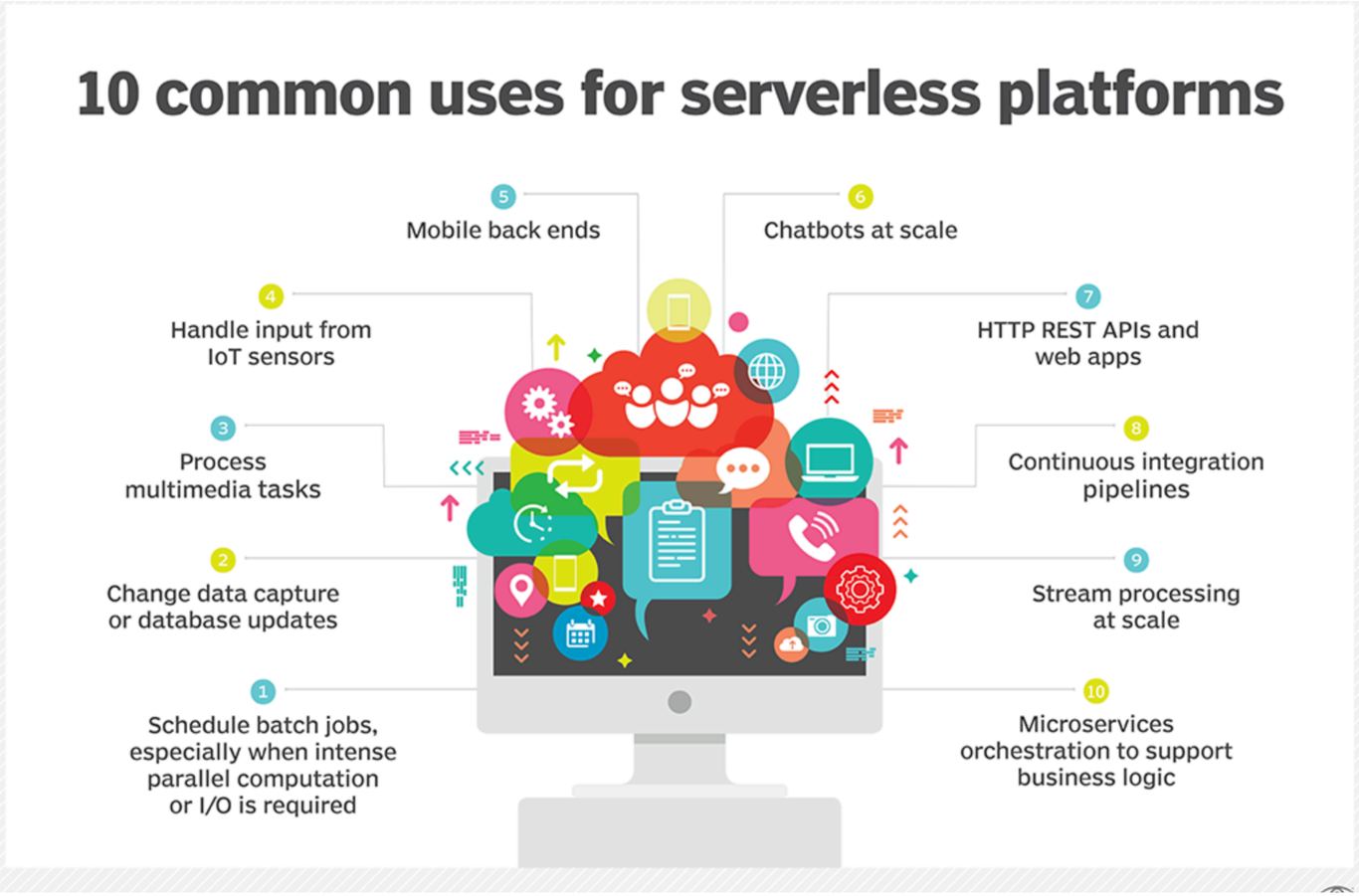
- What is serverless?
- AWS Lambda
- A brief overview of other services mostly used in serverless solutions.
- Our journey from Kubernetes to serverless
- Q&A

What is serverless?

Serverless computing is a cloud computing execution model in which the cloud provider allocates machine resources on demand, taking care of the servers on behalf of their customers.

Key Characteristics

- No server management
- Event-driven execution
- Automatic scaling



Advantages of serverless computing

- Cost efficiency
- Operational efficiency
- Scalability
- Simplified back-end code
- Ecosystem and community

Disadvantages of serverless computing

- Performance issues
- Vendor lock-in
- Limited control and flexibility
- Security
- Monitoring and debugging challenges

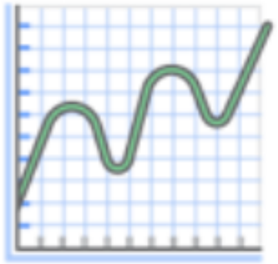
Key AWS Serverless Services

- AWS Lambda: Event-driven compute service that runs code in response to events.
- Amazon API Gateway: Managed service to create, publish, maintain, and secure APIs.
- AWS Step Functions: Orchestrates serverless workflows.
- Amazon S3: Object storage for data and files.
- Amazon DynamoDB: Fully managed NoSQL database service.
- "AWS Fargate" ECS, EKS

Aws Lambda

Serverless compute platform for stateless code execution in response to events

No servers to manage



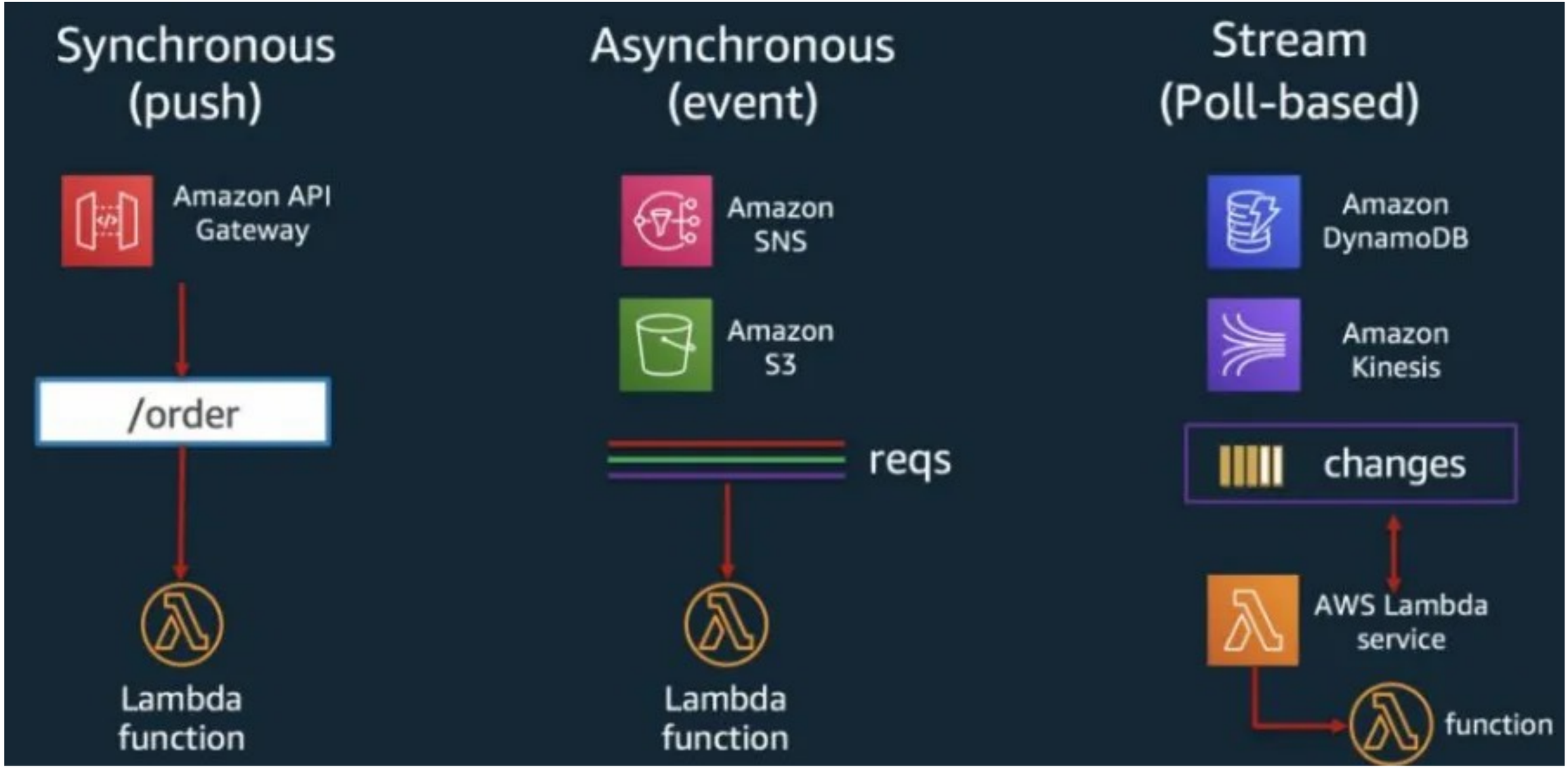
Continuous scaling (not indefinetly)

No idle/cold servers

Key Features of AWS Lambda

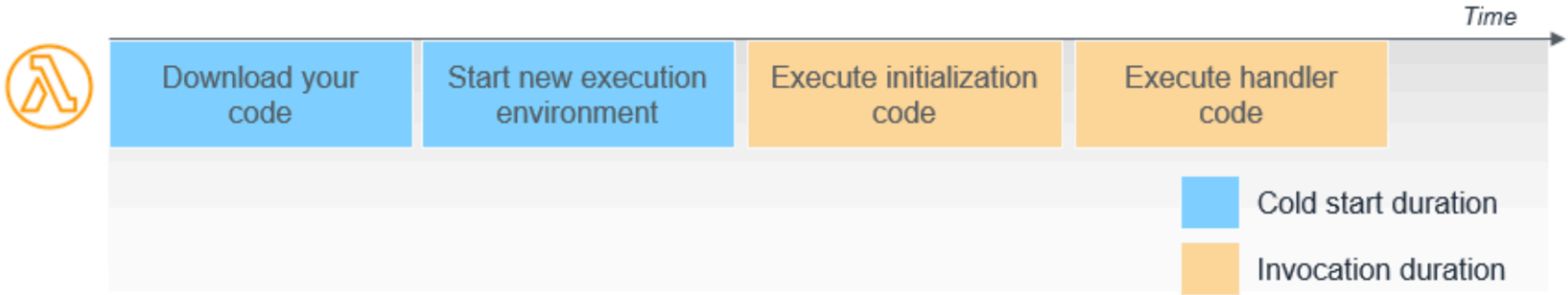
- **Event-Driven:** Trigger functions from various AWS services (S3, API Gateway, DynamoDB, etc.).
- **Automatic Scaling:** Scales your application automatically in response to incoming requests.
- **Pay-As-You-Go:** Only pay for the compute time consumed; no charge when your code isn’t running.
- **Multiple Language Support:** Supports Node.js, Python, Java, Go, C#, and more.

AWS lambda pricing model


Synchronous
(push)
Amazon API
Gateway
/order
Lambda
function
Asynchronous
(event)
Amazon
SNS
Amazon
S3
reqs
Lambda
function
Stream
(Poll-based)
Amazon
DynamoDB
Amazon
Kinesis
changes
AWS Lambda
service
function

Lambda Cold Starts

Cold starts in AWS Lambda occur when an AWS Lambda function is invoked after not being used for an extended period, or when AWS is scaling out function instances in response to increased load. - [AJ Stuyvenberg, serverless hero]



Mitigating the cold start problems

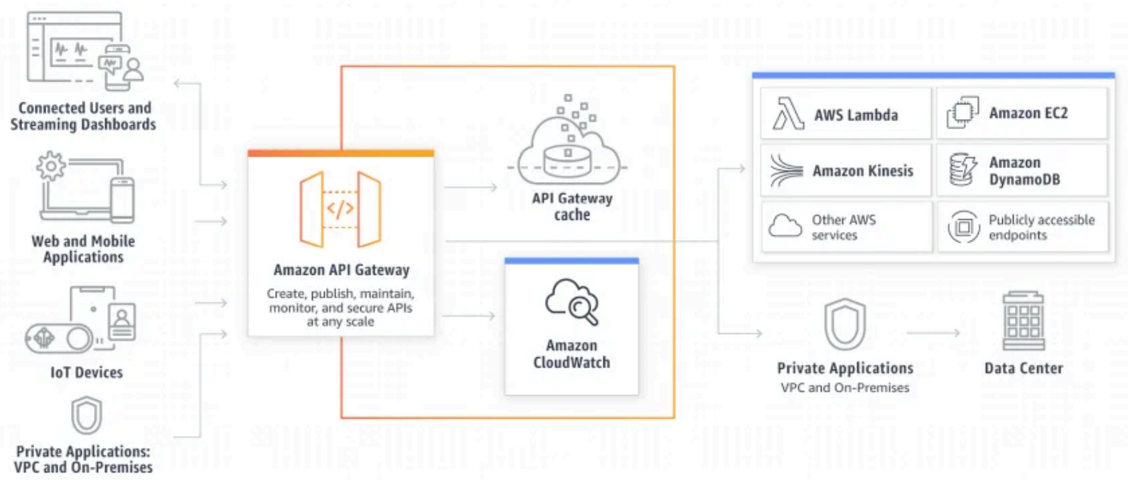
Strategy	Complexity	Cost	Effectiveness
Provisioned Concurrency	High	High	High
Event-Driven Warm-Ups	High	Medium	High
Optimize Function Code and Dependencies	Medium	Low	Medium
Choose Language and Runtime	Low	Low	Medium
Increase Memory Allocation	Low	Medium	Low

API GATEWAY

A fully managed service that makes it easy to create, publish, maintain, and secure APIs at any scale.

Key Characteristics:

- RESTful and WebSocket APIs
- Authentication and authorization
- AWS services integration
- Caching and throttling
- Monitoring and analytics
- CORS support



DynamoDB

A fully managed NoSQL database service that provides fast and predictable performance with seamless scalability.

Key Characteristics:

- NoSQL Database
- Fully managed: Automatic backups, scaling, and patching
- Scalability and performance
- Flexible Data model
- High Availability and durability
- Fine-grained access control
- AWS services integration

Q&A