



AWS Theory

- Serverless: Introduction

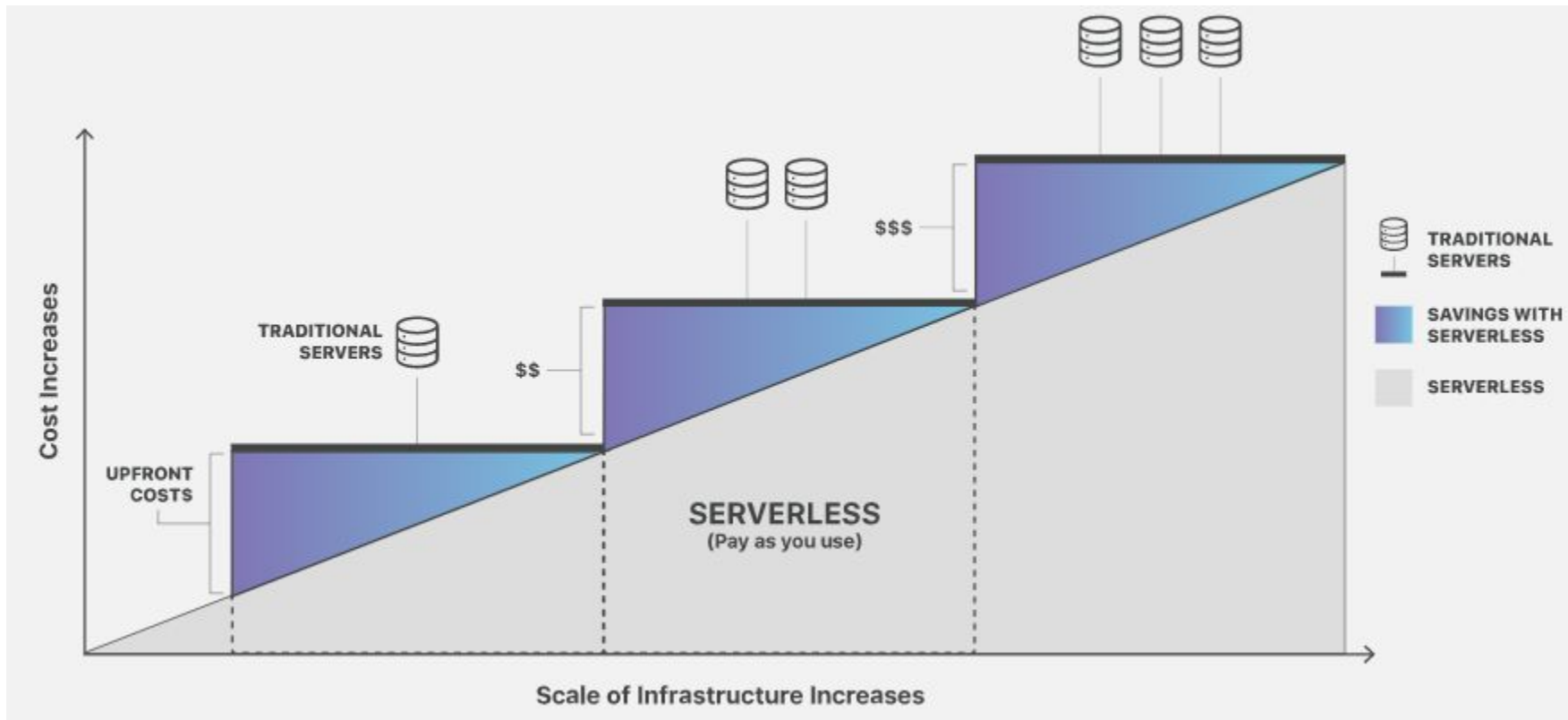
Sung-Dong Kim,
School of Computer Engineering,
Hansung University

Contents

- What - Why
- AWS Serverless Platform
- 이점
- 사례
- Application 구축 사례

What (1)

- 사용량에 따라 백엔드 서비스를 제공하는 방법
- 물리적 서버는 사용되며 개발자는 서버를 알 필요 없음
- 웹 애플리케이션 구축 역사
 - 서버 보유
 - 클라우드 시대 - 고정 개수 서버나 서버 공간 대여
 - 서버리스 - 고정 대역폭이나 서버 개수를 유지하지 않아도 됨 -> 사용량에 따라 백엔드 서비스 비용 지불



CLOUDFLARE: Serverless

What (2)

- 운영상의 책임을 클라우드로 전환 -> 민첩성과 혁신을 높이는 클라우드의 네이티브 아키텍처
- 서버를 고려하지 않고 application과 server를 구축하고 실행
- 거의 모든 application, 백엔드 서비스 구축 가능
- 고가용성 실행/확장에 필요한 사항이 자동 처리 됨

Why?

- 민첩성, 낮은 비용으로 최신 application 개발
- 개발자는 개발에 집중

AWS Serverless Platform (1)

- serverless application 구축/실행을 위한 완전 관리형 서비스 제공
- 백엔드 구성 요소 프로비저닝, 애플리케이션 내결함성/가용성에 대해 고려하지 않아도 됨

AWS Serverless Platform (2)

- Computing: Lambda, Lambda@Edge, Fargate
- Storage: S3, EFS
- Data Store: Dynamo DB, Aurora Serverless
- API Proxy: API Gateway
- Application 통합: SNS, SQS, AppSync, EventBridge
- Ochestration: Step Functions

AWS Serverless Platform (3)

- 분석: Kinesis, Athena
- 개발자 도구

이점

- 낮은 비용 - 종량제 요금
- 서버 관리 불필요
- 유연한 규모 조정: 간편한 확장성
- 자동화된 고가용성
- 빠른 개발: 복잡한 배포 과정, 버그 수정 대신 필요에 따라 코드 추가/수정

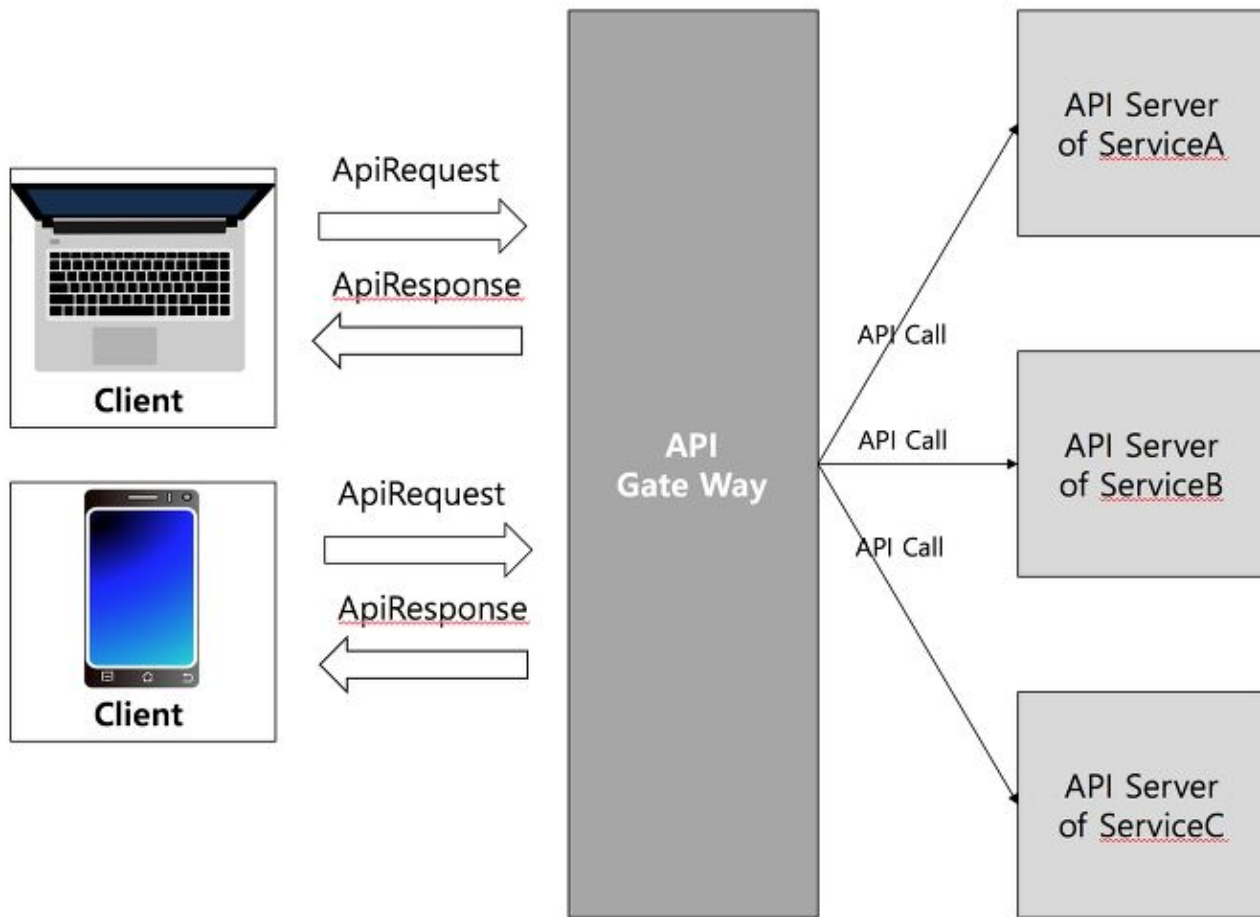
사례

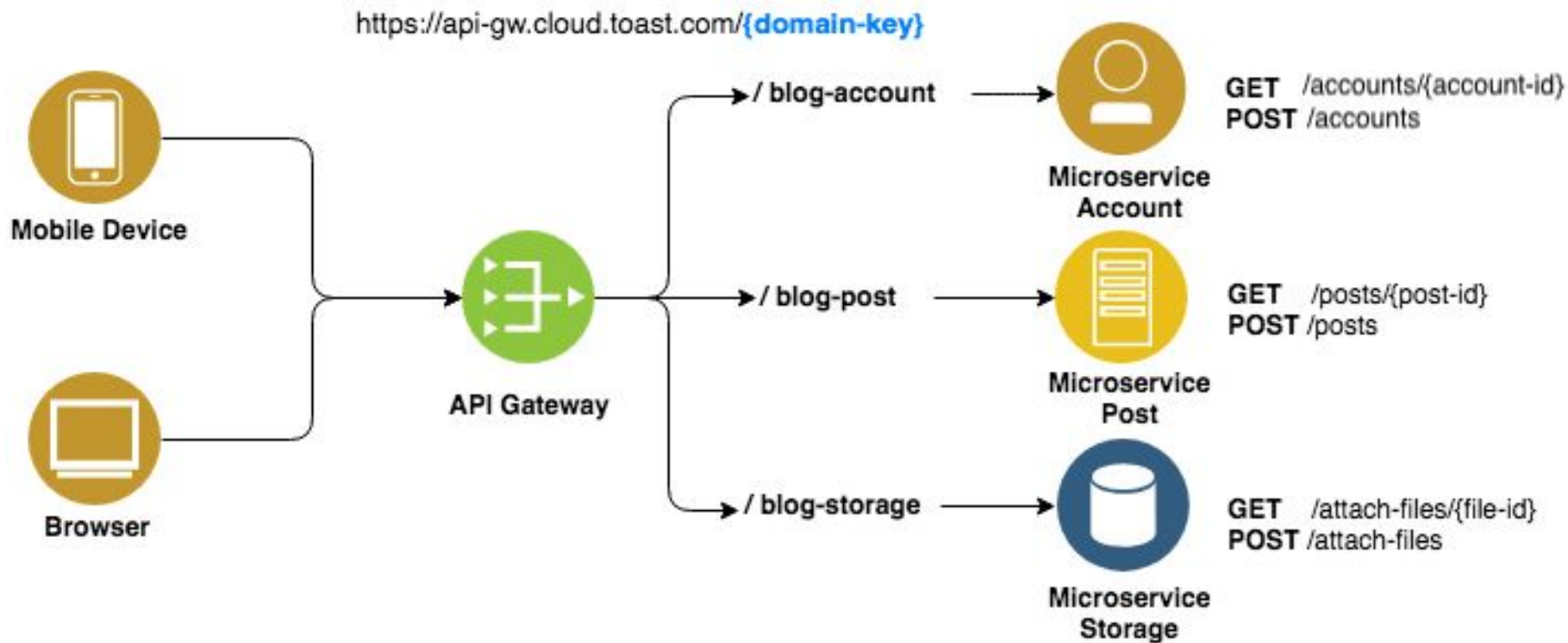
- 코카콜라: Lambda, Step Functions
- FINRA
- iRobot: Lambda, IoT -> Roomba 진공청소기에 연결하는 web application 실행
- Autodesk: Lambda -> IT 운영 자동화

Application 사용 사례 (1)

- 웹 애플리케이션과 백엔드 구축 (1)
 - Lambda, API Gateway, S3, Dynamo DB
 - 예: 날씨 애플리케이션



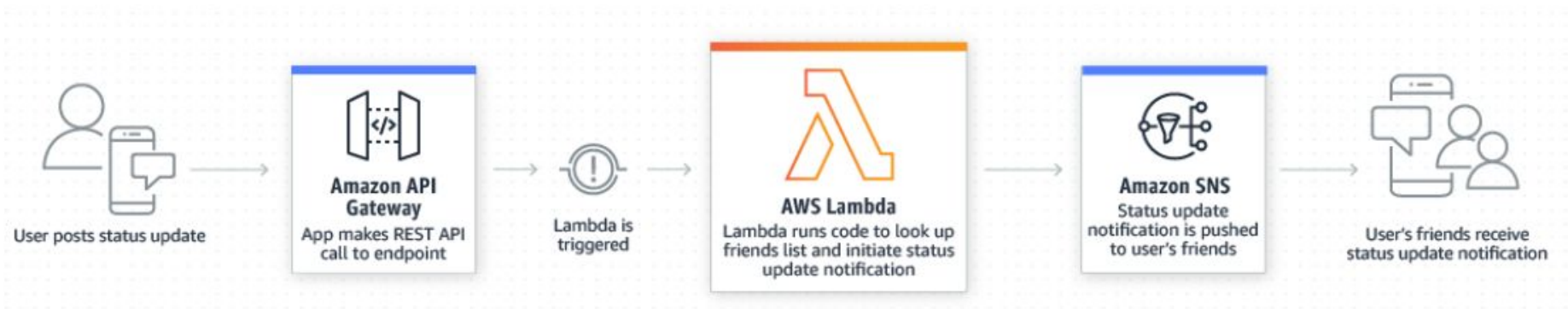




API Gateway 소개 (TOAST Service 들여다보기)

Application 사용 사례 (2)

- 웹 애플리케이션과 백엔드 구축 (2)
 - 예: 소셜 미디어 앱에 대한 모바일 백엔드



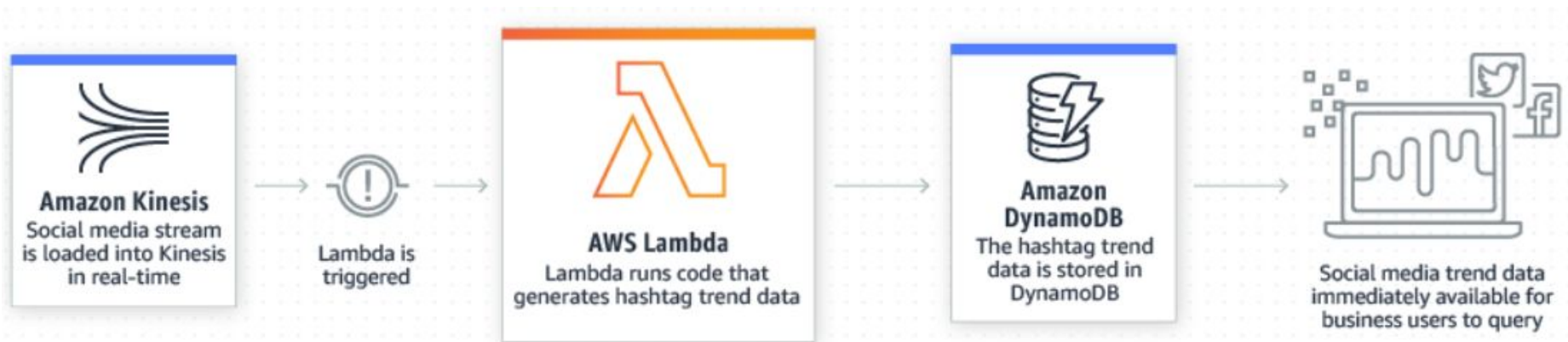
Application 사용 사례 (3)

- 실시간 데이터 처리 시스템 구축 (1)
 - Lambda, Kinesis, S3, Dynamo DB
 - 예: 이미지 썸네일 생성



Application 사용 사례 (4)

- 실시간 데이터 처리 시스템 구축 (2)
 - 예: 스트리밍 소셜 미디어 데이터 분석



Source



- <https://aws.amazon.com/ko/serverless/>