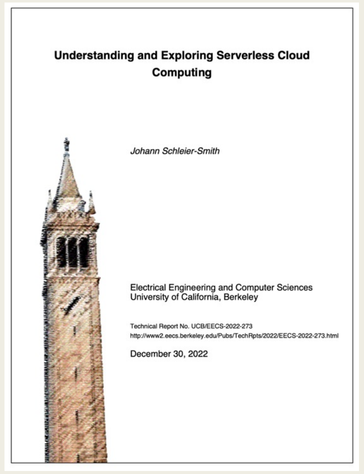
UNDERSTANDING SERVERLESS CLOUD COMPUTING

Chapter 2.1 ~ 2.6



Understanding and Exploring Serverless Cloud Computing

Johann Schleier-Smith

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2022-273
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2022/EECS-2022-273.html

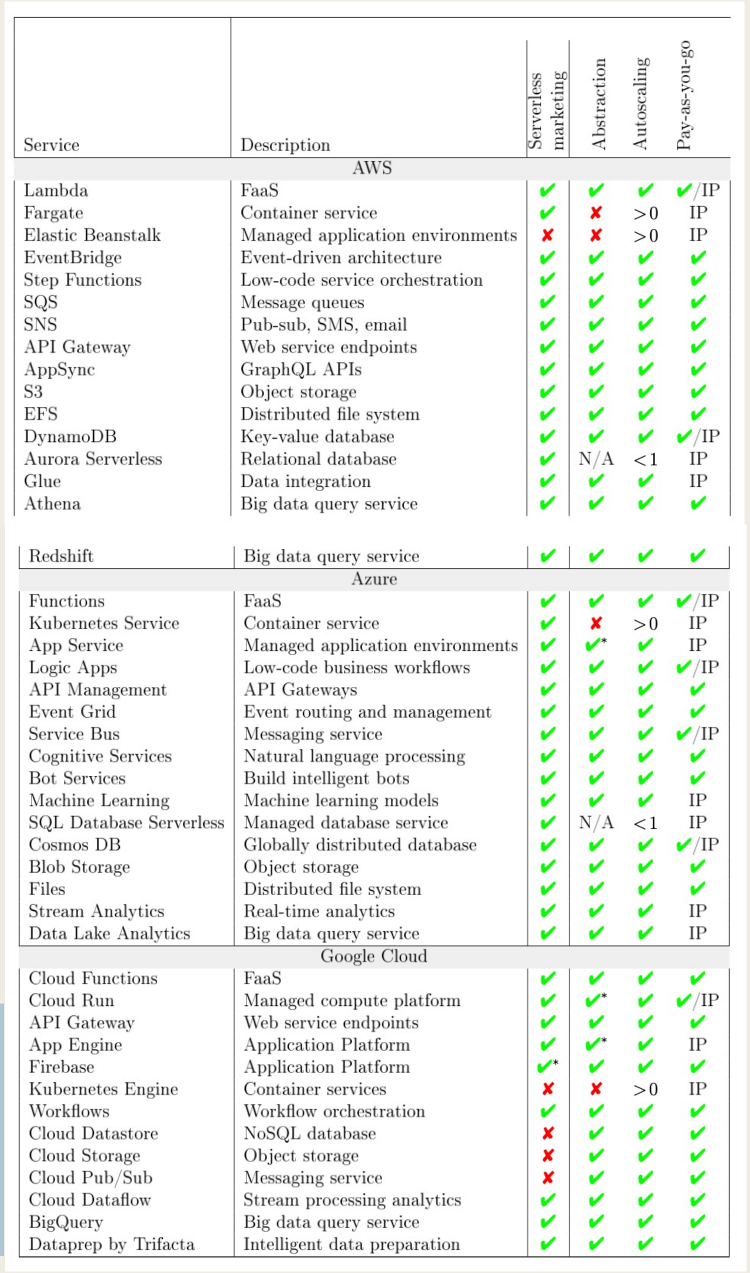
December 30, 2022

Pin Chun Lu

Outline

- Limitations
  - Function as a service (FaaS)
  - Storage, Databases, and File Systems
  - Big Data Analysis

Characteristics of Serverless computing:
1. Abstraction
2. autoscaling
3. Pay-as-you-go



Service	Description	Serverless marketing	Abstraction	Autoscaling	Pay-as-you-go
AWS					
Lambda	FaaS	✓	✓	✓	✓/IP
Fargate	Container service	✓	✗	>0	IP
Elastic Beanstalk	Managed application environments	✗	✗	>0	IP
EventBridge	Event-driven architecture	✓	✓	✓	✓
Step Functions	Low-code service orchestration	✓	✓	✓	✓
SQS	Message queues	✓	✓	✓	✓
SNS	Pub-sub, SMS, email	✓	✓	✓	✓
API Gateway	Web service endpoints	✓	✓	✓	✓
AppSync	GraphQL APIs	✓	✓	✓	✓
S3	Object storage	✓	✓	✓	✓
EFS	Distributed file system	✓	✓	✓	✓
DynamoDB	Key-value database	✓	✓	✓	✓/IP
Aurora Serverless	Relational database	✓	N/A	<1	IP
Glue	Data integration	✓	✓	✓	IP
Athena	Big data query service	✓	✓	✓	✓
Redshift	Big data query service	✓	✓	✓	✓
Azure					
Functions	FaaS	✓	✓	✓	✓/IP
Kubernetes Service	Container service	✓	✗	>0	IP
App Service	Managed application environments	✓	✓*	✓	IP
Logic Apps	Low-code business workflows	✓	✓	✓	✓/IP
API Management	API Gateways	✓	✓	✓	✓
Event Grid	Event routing and management	✓	✓	✓	✓
Service Bus	Messaging service	✓	✓	✓	✓/IP
Cognitive Services	Natural language processing	✓	✓	✓	✓
Bot Services	Build intelligent bots	✓	✓	✓	✓
Machine Learning	Machine learning models	✓	✓	✓	IP
SQL Database Serverless	Managed database service	✓	N/A	<1	IP
Cosmos DB	Globally distributed database	✓	✓	✓	✓/IP
Blob Storage	Object storage	✓	✓	✓	✓
Files	Distributed file system	✓	✓	✓	✓
Stream Analytics	Real-time analytics	✓	✓	✓	IP
Data Lake Analytics	Big data query service	✓	✓	✓	IP
Google Cloud					
Cloud Functions	FaaS	✓	✓	✓	✓
Cloud Run	Managed compute platform	✓	✓*	✓	✓/IP
API Gateway	Web service endpoints	✓	✓	✓	✓
App Engine	Application Platform	✓	✓*	✓	IP
Firebase	Application Platform	✓*	✓	✓	✓
Kubernetes Engine	Container services	✗	✗	>0	IP
Workflows	Workflow orchestration	✓	✓	✓	✓
Cloud Datastore	NoSQL database	✗	✓	✓	✓
Cloud Storage	Object storage	✗	✓	✓	✓
Cloud Pub/Sub	Messaging service	✗	✓	✓	✓
Cloud Dataflow	Stream processing analytics	✓	✓	✓	✓
BigQuery	Big data query service	✓	✓	✓	✓
Dataprep by Trifacta	Intelligent data preparation	✓	✓	✓	✓

Table 2.1: Serverless characteristics of cloud services

FaaS Limitations Overview

- Limited Execution Time
- Resources Restrictions
- Ephemeral State
- Networking Limitations
- Lack of Durable Storage
- Performance Overhead Concerns

FaaS Limitations Detailed (1)

- Execution Time :
  - AWS Lambda's 1-minute which extended to 15 minutes.
  - Google Cloud Run offers up to 1-hour
- Resources Restrictions:
  - Limitations on code size, memory, and CPU have been relaxed over time
    - Size: Evolved from 250 MB zip file to 10 GB Docker in AWS Lambda
    - Memory: 1.5 GB to 10 GB
    - CPU: 2 cores evolved to 6 cores
    - Architecture: x86 to ARM (but no GPU support)
- its capabilities are expanding, certain limitations persist

FaaS Limitations Detailed (2)

- Ephemeral State:
  - Functions are stateless
- Limited Networking
  - No direct Incoming network connections

- Current Solution:
  - Add networking capabilities to FaaS
  - Shift from unique function-based service to broader on-demand server

Resources allocated?

FaaS Limitations Detailed (3) – Challenges Stemming from Statelessness

- 1. no durable storage
  - Data lost once the function ends
  - Solutions:
    - External integration (databases, file systems) needed for long-term storage
    - Cost added due to data transfer
- 2. Ephemeral State => trap
  - Existing state is unaddressable and unnamed.
  - No guarantees on routing to the same function instance.
  - Potential solution: Session affinity

FaaS Limitations Detailed (4)

- Two scenarios relate to the overhead of invoking a function
- 1. Cold Start
  - New instance initiation
  - Time-consuming setup: environment, code loading, configurations
  - Efforts by cloud providers and researchers to minimize delays
- 2. Warm Start
  - AWS Lambda, for instance, may take ~25 ms compared to faster regular web services (~ 1 ms)
  - Possible reasons: Internal queuing mechanisms, other factors (multi-tenant)
- The Hidden Costs of FaaS
  - Charges can be based on invocation and execution time
  - Quick tasks may have disproportionate costs

meet Commercial deployment?

FaaS Limitations Review

- **Execution Time:** Increasing but still limited
- **Resources:** Code size, memory, CPU constraints
- **State:** Functions are inherently stateless
- **Networking:** No built-in support for incoming connections
- **Storage:** Requires integration with external solutions
- **Performance:** Overheads from cold/warm starts

Limitations for Object Storage, Key-Value Storage, OLTP Databases, and File Systems

Improvement might be in response to the needs of FaaS workloads

	Object Storage	Key-Value Storage and File Systems	OLTP Databases
Benefits	Economical for long-term storage	Supports richer data models	Rich capabilities
drawbacks	• Expensive data access • simplistic data model	High cost for data access and retention	Limited scalability Exceptions: CockroachDB offer distributed SQL

Gaps: Current offerings have missing functionality or high costs

Opportunities: Room for innovation

Limitations for Big Data Analytics Systems

- Strength
  - Competitive with server-based variants.
  - No significant limitations for general tasks.
- Architecture Concern - Separation of Compute & Storage
  - Service Solutions: Analysis close to storage
  - Integrated Storage: Big Query's built-in storage
  - Code Execution in Storage: ZeroVM with operator push-down
- Multi-Tenant Settings
  - Google Big Query: Ensures availability through capacity reservations
  - Trade-off: Paying for idle capacity vs. pay-as-you-go benefits

Questions?