UNIVERSITÉ Concordia UNIVERSITY

Appendix -
Server-less Computing (Function as a Service)

Roch Glitho, PhD

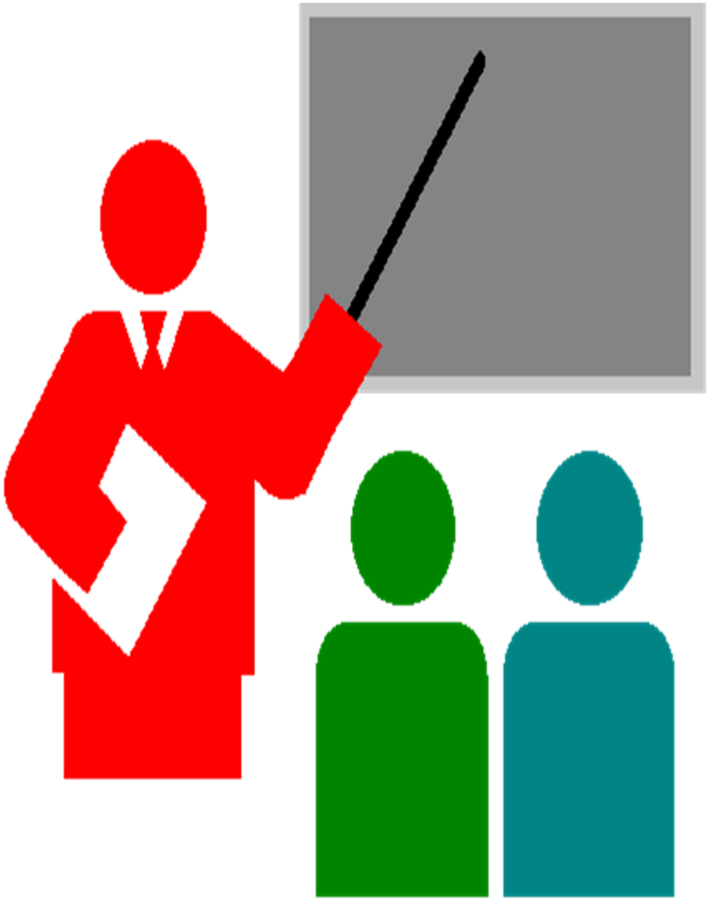
Professor and Canada Research Chair

My URL - http://users.encs.concordia.ca/~glitho/

Concordia University Engineering and Computer Science

Concordia Institute for Information Systems Engineering

Server-less Computing (Function as a Service)



- Introduction
- Architecture
- Pros / Cons

Introduction

Server-less does not mean there is no server !!!

- There are indeed servers !!!
    - However the servers are completely transparent to the cloud users, unlike (Virtual Machine (VM), Containers, Uni-kernel)
        - Server-less computing might actual rely on VMs or containers or uni-kernels
    - Cloud users deal with functions
        - thus Functions as a Service (FaaS)

Architecture

Principles

1) Applications built as a set of functions
2) When there is a request for a given function, a run time environment (e.g. VM, container, uni-kernel) is launched with the function code + libraries
3) The run time is terminated after the execution of the function

Architecture (Reference 1)

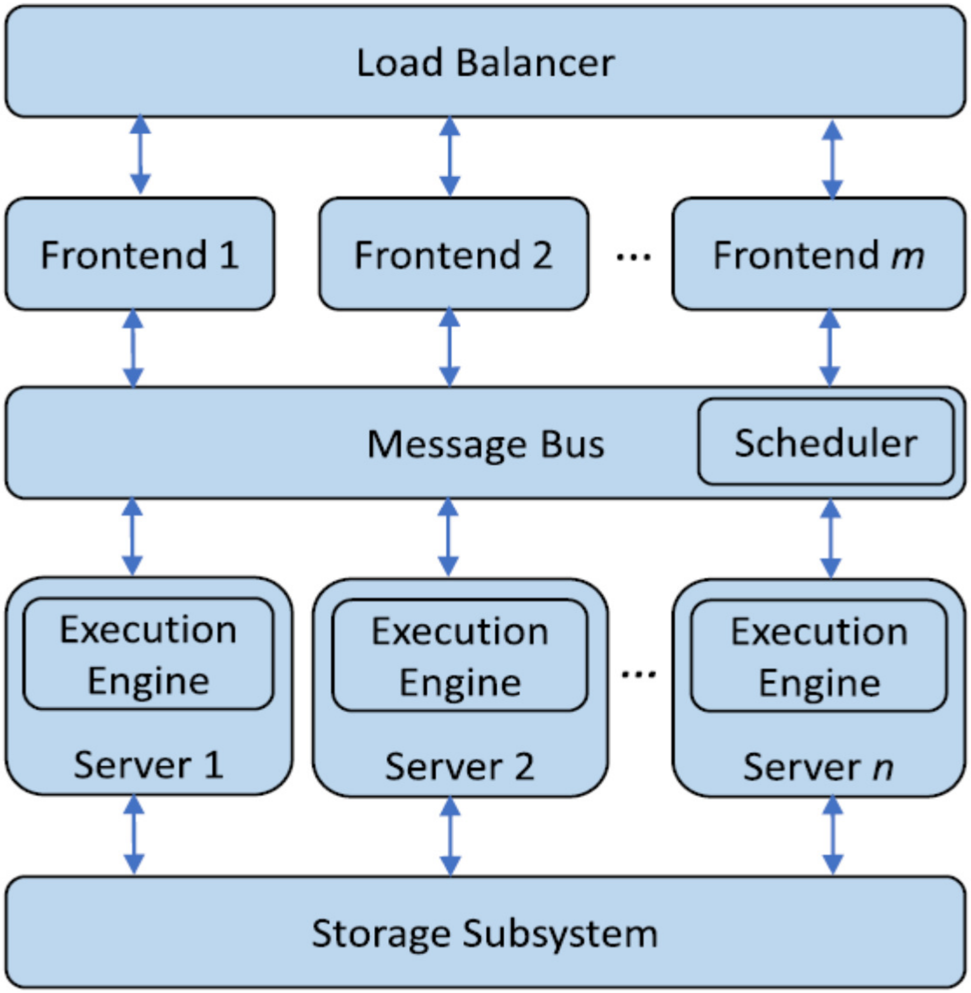


Fig. 1. *Serverless platform architecture.*

Architecture

Load balancer:

- Self explanatory

Front end:

- End user interface

Message bus and scheduler:

- Mediation between front ends and execution engines

Architecture

Load balancer:

- Self explanatory

Front end:

- End user interface

Message bus and scheduler:

- Mediation between front ends and execution engines
  - Relies on a publication / subscription principles

Architecture

Execution engine:

- Self explanatory
  - Might rely on VM, containers and uni-kernels

Storage sub-system:

- States
- Persistent data

Pros (Examples)

- No real / virtual server management by cloud users

- Resource Efficiency and low cost

- Built-in scalability

Cons (Examples)

- **Most cited:**
  - Start up latency
- **Others:**
  - Learning curve of the new programming model (e.g. stateless functions + events)

Pros vs Cons

- Decision to be made on case by case basis (Ref. 1)

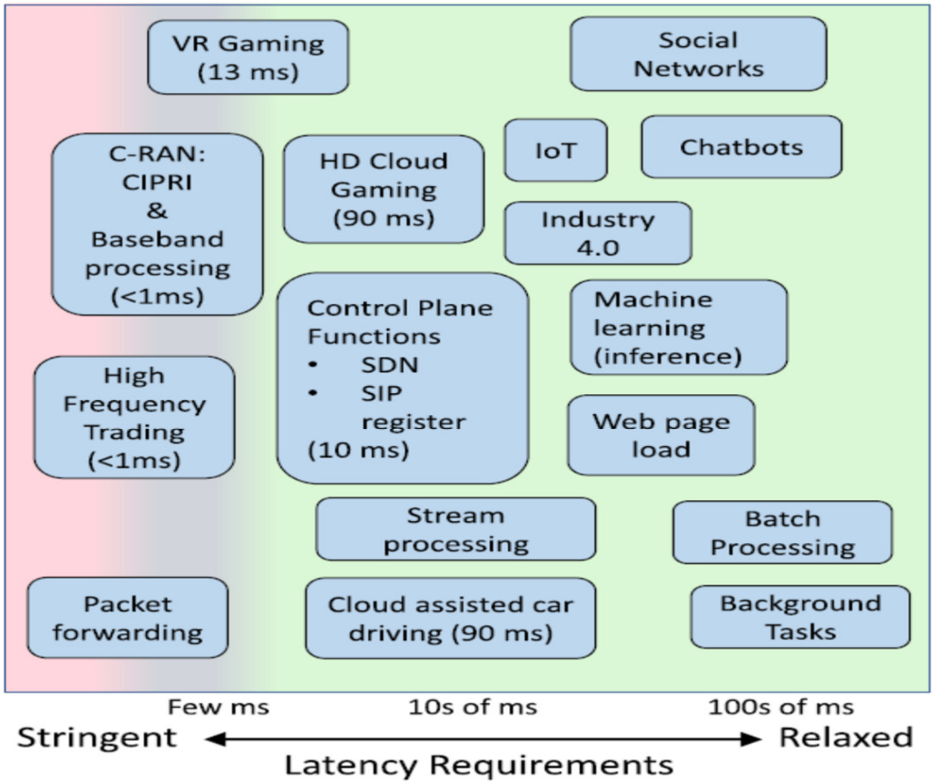


Fig. 3. *Latency requirement ranges for various applications.*

Pros vs Cons

- **Decision to be made on case by case basis (Ref. 1)**

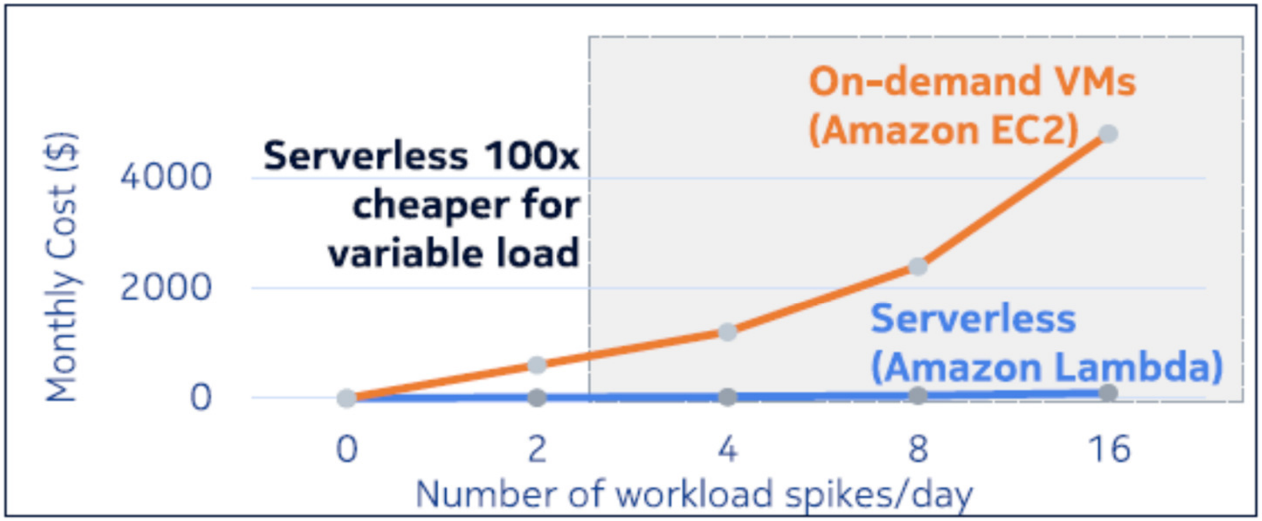


Fig. 4. *Cost comparison between Amazon Lambda (serverless) and Amazon EC2 (VMs) for spiky workload. In the gray region, serverless is 100x cheaper.*

References

1. P. Aditya et al, Will Servless Computing Revolutionize NFV, Proceedings of the IEEE, April 2019

The End