@Large Research
Massivizing Computer Systems

http://atlarge.science

The SPEC-RG Reference Architecture For the Compute Continuum

Matthijs Jansen, Auday Al-Dulaimy, Alessandro Papadopoulos, Animesh Trivedi, Alexandru Iosup

m.s.jansen@vu.nl
atlarge.science/offsense

VU VRIJE UNIVERSITEIT AMSTERDAM

spec RG Cloud

spec Research

Use Case: Video Processing

Requirement: Process live video

Problem: Little resources for native processing

Solution: Offload data

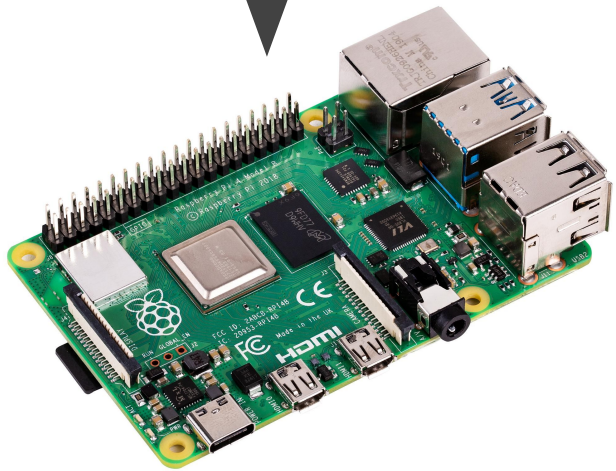
Task Offloading

Offloading targets?

Available resources?

Requirements?

→ Computing models!



Mobile Cloud Computing

\+ High compute capacity

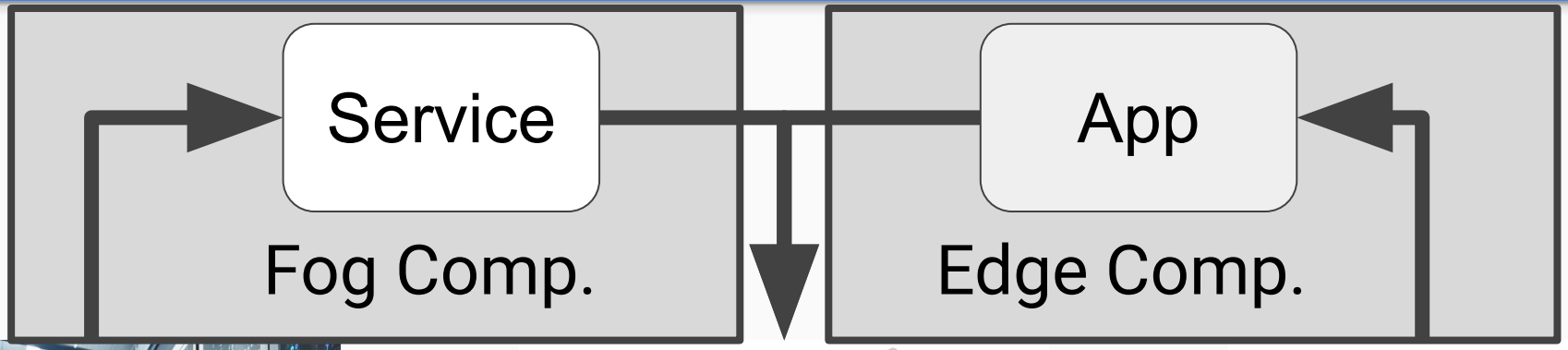
\-- But high latency

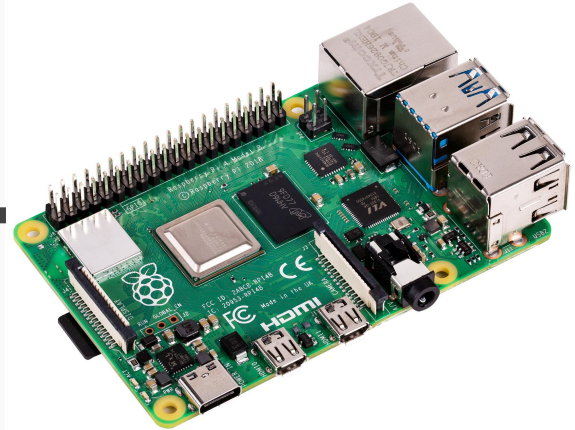
→ Resource-heavy tasks with no latency requirement

Cloud

Endpoint

Fog vs Edge Computing



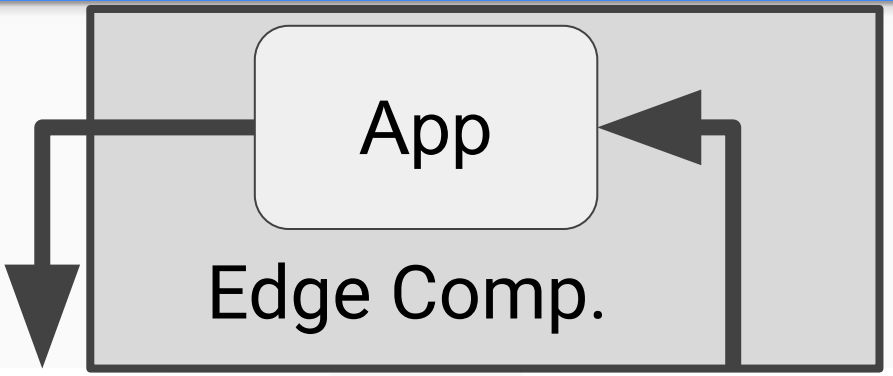


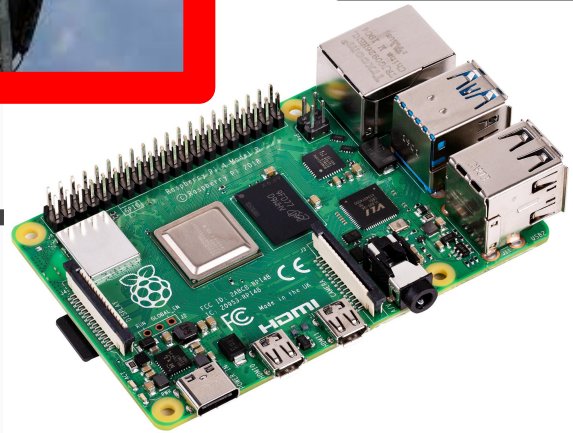
Cloud

Edge

Endpoint

Multi-access Edge Computing





Cloud

Edge

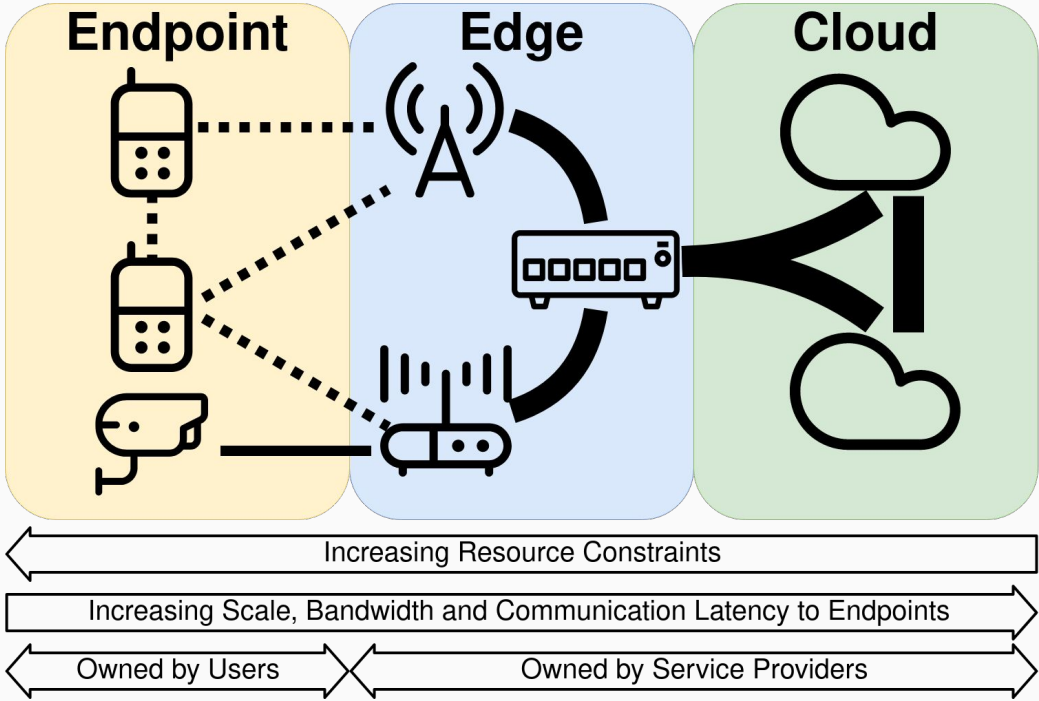
Endpoint

Compute Continuum

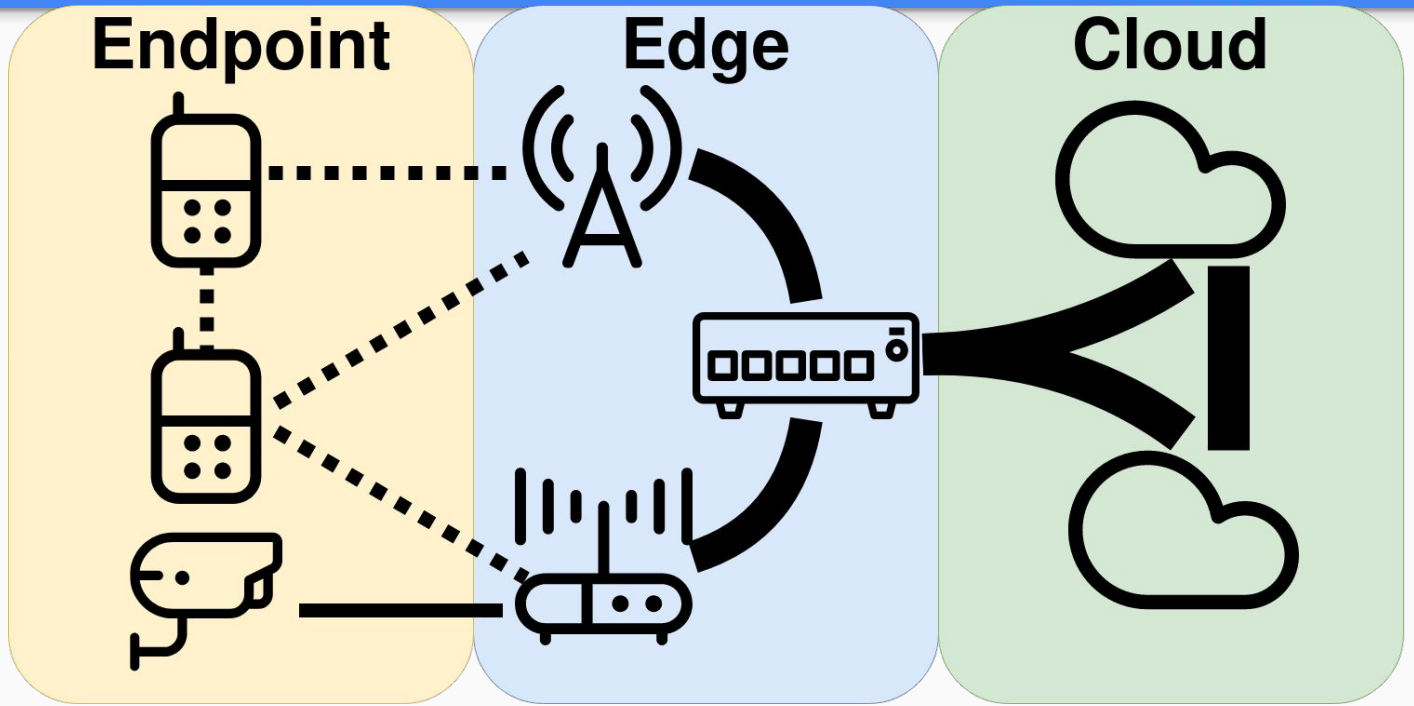
Many computing models, part of 1 continuum

Unification:

1. Break isolated models
2. Shared responsibilities
3. End-to-end concerns

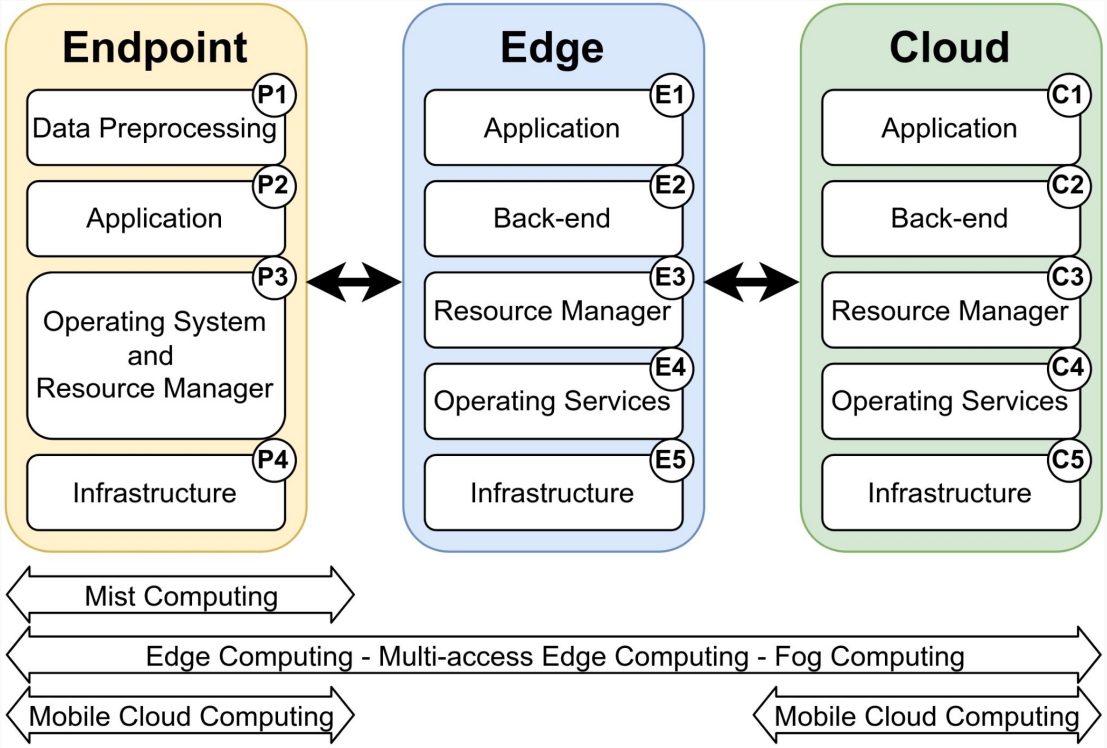


Compute Continuum



SPEC-RG Reference Architecture for Cont.

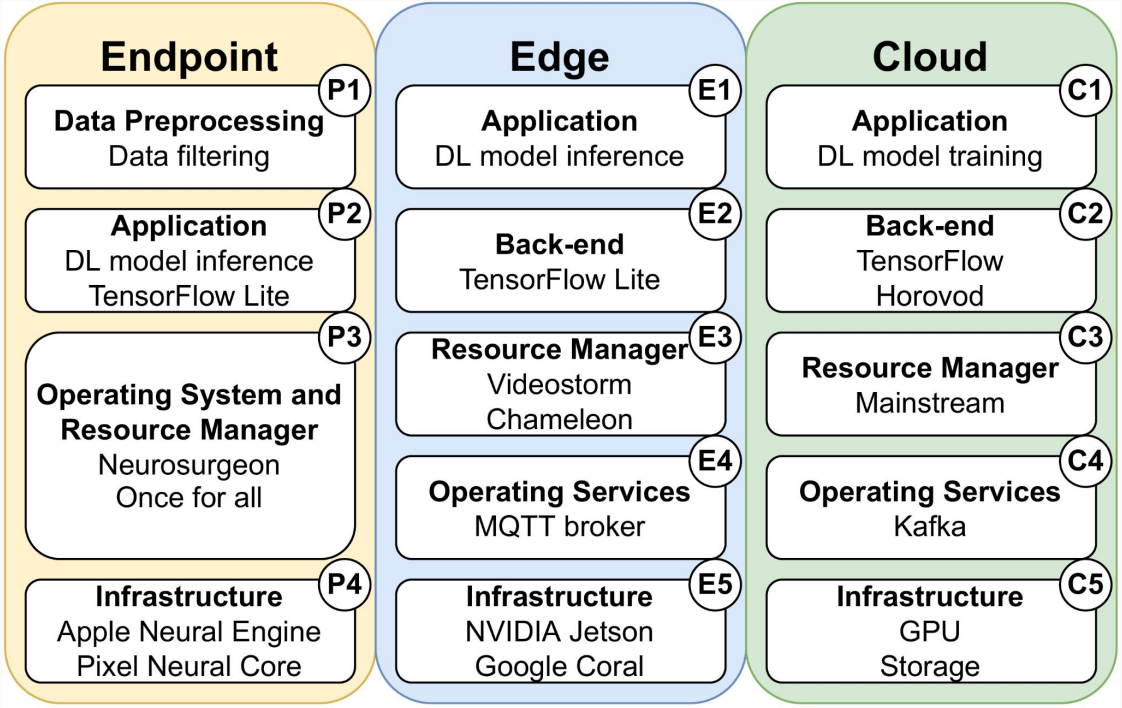
Common components in continuum



Implementation in specific components

Domain-specific Architecture: Deep Learn.

Overview of all deployment options



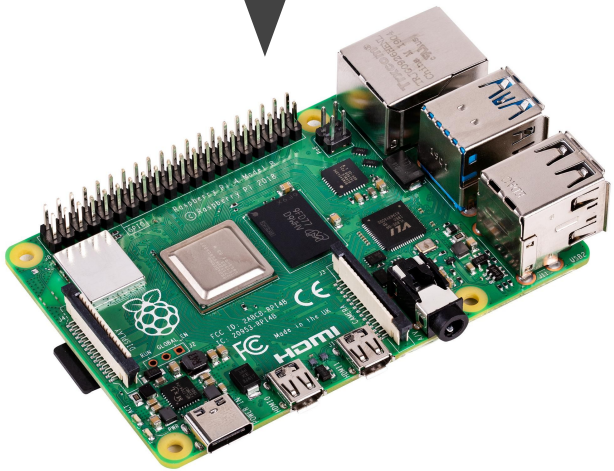
Start of design space exploration → How?

Many Ways to Deploy in the Continuum

Big performance differences

Iterate, but can't test all:

1. Analyze
2. Prune



Continuum

Automate cloud-edge infrastructure deployment and benchmarking in the compute continuum

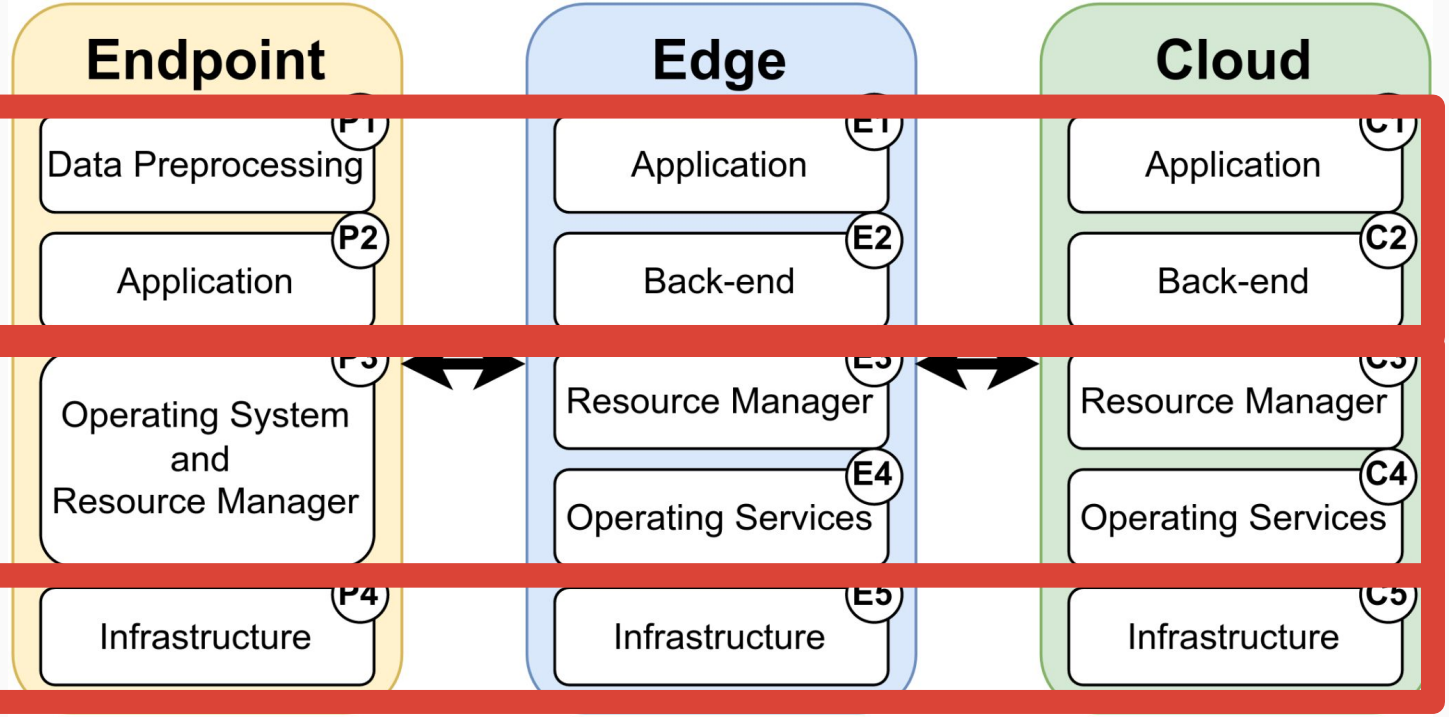
https://github.com/atlarge-research/continuum

SPEC-RG Reference Architecture

Application Benchmark

Software

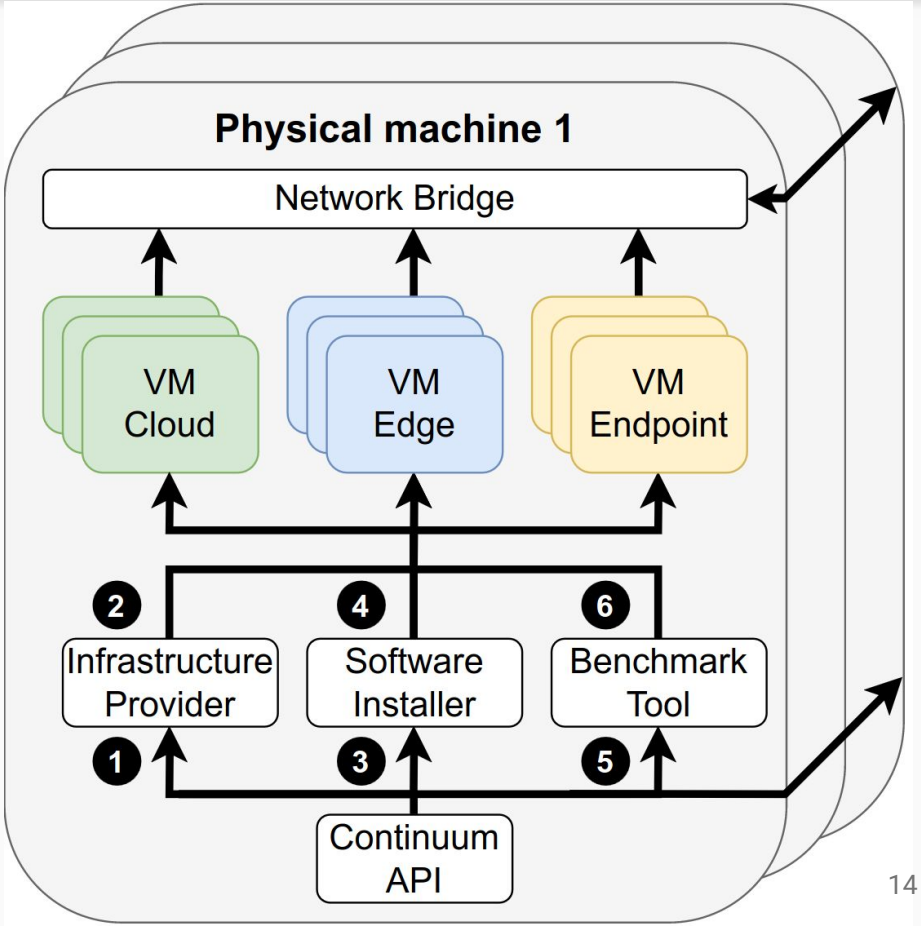
Infrastruct.



The Continuum Framework

Design principles

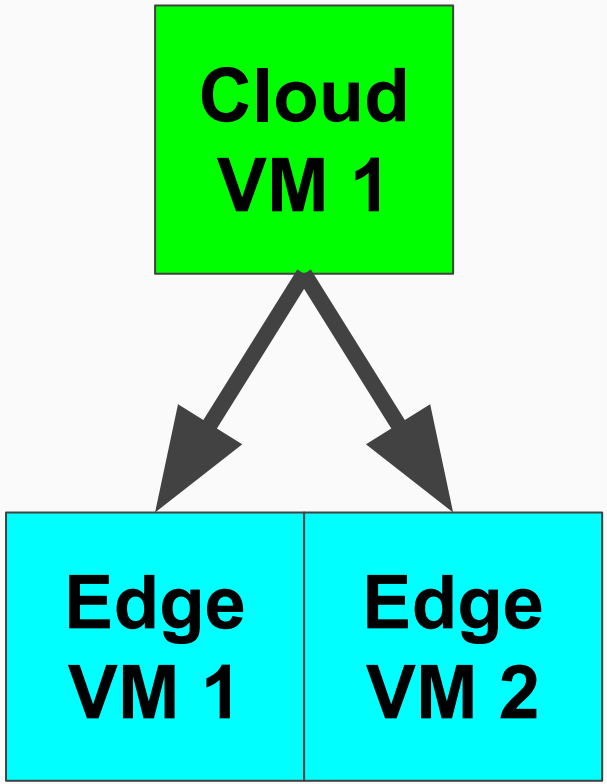
1. Accuracy
   → Hardware deployment
2. Flexibility
   → Emulation
3. Automation
   → Scripting
4. Extendibility
   → Modular design



Step 1: Infrastructure Provisioning

config-example.cfg

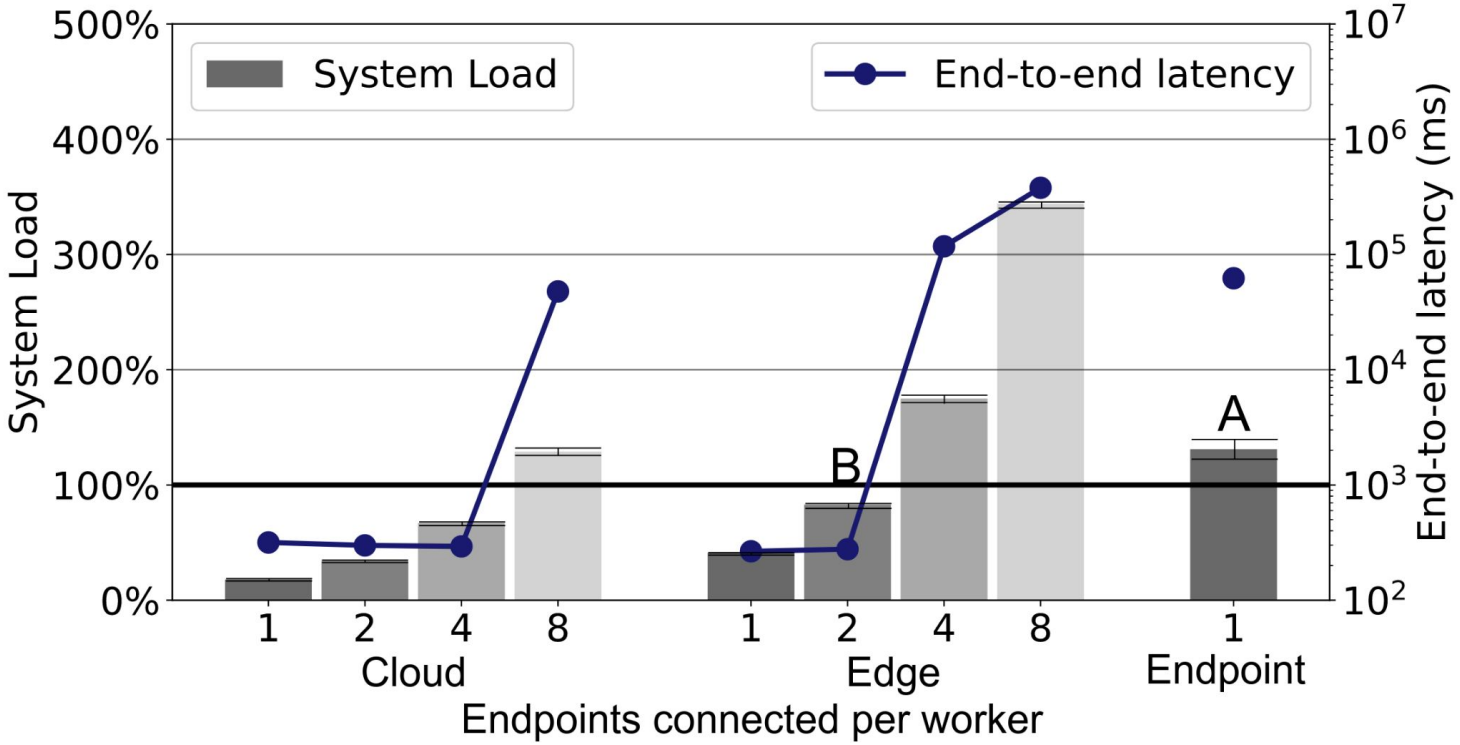
Provider	= GCP
Cloud-VMs	= 1
Cloud-cores	= 8
Cloud-memory	= 16 GB
Edge-VMs	= 2
Edge-cores	= 4
Edge-memory	= 4 GB
Cloud-Edge-Latency	= 15 ms



Multiple Endpoints per Offload Target

System Load < 100%: Real-time processing

> 100%: Queue starts to form



Take-away message

Compute continuum is complex, difficult to navigate

We offer:

1. SPEC-RG Reference Architecture for the Comp. Cont.
2. Continuum: Automate Infrastructure Deployment and Benchmarking in the Compute Continuum

Open Research Objects (ORO)

Research Objects Reviewed (ROR)

https://github.com/atlarge-research/continuum
https://atlarge-research.com/offsense/

This presentation was based of work from

1. **Matthijs Jansen**, Auday Al-Dulaimy, Alessandro V. Papadopoulos, Animesh Trivedi, and Alexandru Iosup (2023). The SPEC-RG Reference Architecture for the Compute Continuum. 2023 23th IEEE/ACM International Symposium on Cluster, Cloud, and Internet Computing. https://atlarge-research.com/pdfs/2023-ccgrid-refarch.pdf
2. **Matthijs Jansen**, Linus Wagner, Animesh Trivedi, and Alexandru Iosup. Continuum: Automate Infrastructure Deployment and Benchmarking in the Compute Continuum (2023). Companion of the 2023 ACM/SPEC International Conference on Performance Engineering (ICPE'23). https://atlarge-research.com/pdfs/2023-fastcontinuum-continuum.pdf

Further reading

1. Alexandru Iosup, Alexandru Uta, Laurens Versluis, Georgios Andreadis, Erwin van Eyk, Tim Hegeman, Sacheendra Talluri, Vincent van Beek, and Lucian Toader (2018). Massivizing Computer Systems: a Vision to Understand, Design, and Engineer Computer Ecosystems through and beyond Modern Distributed Systems. CoRR. http://arxiv.org/abs/1802.05465