Università degli Studi di Napoli Federico II

itee PhD
information technology
electrical engineering

DIETI
UNINA

Marco Barletta

Mixed-criticality Orchestration of Real-time Containerized Systems

Tutor: Marcello Cinque

Cycle: XXXVII

Year: 2024

itee PhD
information technology
electrical engineering

Candidate’s information

- MSc degree in Computer Engineering at UniNa (October 2021)
- Research group: DEpendable and Secure Software Engineering
- and Real Time (DESSERT) research group
- PhD start date: 01/11/2021
- Scholarship type: UNINA
- Periods abroad:
  -Germany at Nokia Bell Labs (01/06/22 to 30/09/22)
  -USA at University of Illinois at Urbana-Champaign (12/04/23 to 23/12/23)

Summary of study activities

- Courses attended:
  - Real-time Industrial Systems
  - Quantum information
  - Imprenditorialità accademica
  - Virtualization technologies and their applications
  - IoT data anlysis
  - Statistical data analysis for science and engineering research
  - How to boost your PhD
  - Strategic orientation for STEM and research writing
- Relevant seminars:
  - Industry 4.0 Fundamentals in Bosch Applications
  - Potential and challenges of next generation railway signaling systems

Research area

Real-time cloud for mixed-criticality environments

Changeable conditions

Increasingly complex

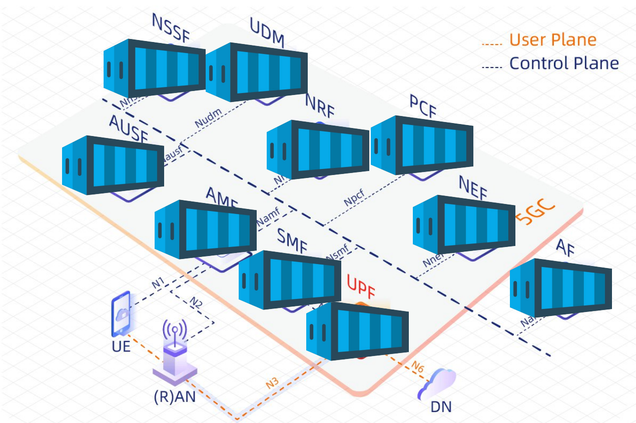
Strict non-functional requirements

Need for maintainability, flexibility, interoperability, scalability

How? Cloud-native technologies!
(e.g., containers and orchestrators)

❯

consolidation,
offloading,
migration,
integration



Virtual Network Functions

Reconfigurable manufacturing

Interconnected vehicles

Research problem

Since they were designed for different environments,

can we actually **depend** on them?

Research results

- Current container orchestration systems can undermine application SLOs in non-nominal conditions
  - One error can lead to the failure of an entire cluster
  - It may take tens of seconds to scale a service or handle a failure
- Real-time containers based on Linux and general purpose hardware cannot host real-time critical tasks
- Mixed-criticality orchestration improves the availability, determinism, and response times of critical services
- Ad-hoc software (e.g., RTOS and hypervisors) and asymmetric hardware can be integrated into orchestration systems to improve containers' non functional properties
- Stateless cloud components improve resiliency but are not a panacea

Research products

[P1]	M. Barletta, M. Cinque, and R. Della Corte “Hierarchical Scheduling for Real-Time Containers in Mixed-Criticality Systems” 32nd IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW) Wuhan, China, Oct. 2021, pp. 286-287
[P2]	M. Barletta, M. Cinque, L. De Simone, and R. Della Corte “Achieving isolation in mixed-criticality industrial edge systems with real-time containers” 34th Euromicro Conference on Real-Time Systems (ECRTS 2022)Modena, Italy, July 2022, pp. 15:1-15:23, Schloss-Dagstuhl-Leibniz Zentrum für Informatik.
[P3]	M. Barletta, M. Cinque, L. De Simone, and R. Della Corte, “Introducing k4. 0s: a Model for Mixed-Criticality Container Orchestration in Industry 4.0” 2022 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech) Falerna, Italy, September 2022, pp. 1-6, IEEE.
[P4]	M. Barletta, M. Cinque, L. De Simone, R. Della Corte, G. Farina, D. Ottaviano “RunPHI: Enabling Mixed-criticality Containers via Partitioning Hypervisors in Industry 4.0” 2022 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)Charlotte, NC, USA, October 2022, pp. 134-135, IEEE.
[P5]	M. Barletta, M. Cinque, L. De Simone, R. Della Corte, G. Farina, and D. Ottaviano “Partitioned Containers: Towards Safe Clouds for Industrial Applications” 2023 53rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks - Supplemental Volume (DSN-S), Porto, Portugal, June 2023, pp. 84-88, IEEE
[P6]	M. Barletta, L. De Simone, R. D. Corte and C. Di Martino, "Failover Timing Analysis in Orchestrating Container-based Critical Applications” 2024 19th European Dependable Computing Conference (EDCC) Leuven, Belgium, April 2024, pp. 81-84, IEEE
[P7]	M. Barletta, M. Cinque, D. De Vita, "Orchestrating Mixed-Criticality Cloud Workloads in Reconfigurable Manufacturing Systems” 2024 19th European Dependable Computing Conference – Fast Abstracts and Student Forum Proceedings Leuven, Belgium, April 2024, pp. 81-84, IEEE
[P8]	M. Barletta, M. Cinque, C. Di Martino, Z. Kalbarczyk and R. Iyer, "Mutiny! How Does Kubernetes Fail, and What Can We Do About It?" 2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Brisbane, Australia, June 2024, pp. 1-14., IEEE

Research products

[J1]	M. Barletta, M. Cinque, L. De Simone, and R. Della Corte "Criticality-aware monitoring and orchestration for containerized industry 4.0 environments" ACM Transactions on Embedded Computing Systems, Vol. 23 (1), pp. 1-28, 2024
[M1]	M. Barletta, M. Cinque, and C. Di Martino "SLA-Driven Software Orchestration in Industry 4.0" IEEE Internet of Things MagazineVol. 5 (4), pp. 136-141, 2022,
[A1]	M. Barletta, M. Cinque, L. De Simone, and R. Della Corte "Achieving isolation in mixed-criticality industrial edge systems with real-time containers" (Artifact) 34th Euromicro Conference on Real-Time Systems (ECRTS 2022)Modena, Italy, July 2022, pp. 15:1-15:23, Schloss-Dagstuhl-Leibniz Zentrum für Informatik.
[A2]	M. Barletta, M. Cinque, C. Di Martino, Z. Kalbarczyk and R. Iyer, "Mutiny! How Does Kubernetes Fail, and What Can We Do About It?" (Artifact) 2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Brisbane, Australia, June 2024, pp. 1-14., IEEE
[Pat1]	Patent: WO2024175183A1 - "SLA-Driven Orchestration of Software Containers". Inventors: Catello Di Martino, Marco Barletta, September 2024.

PhD thesis overview

- Problem statement
  How can orchestration systems meet mixed-criticality requirements (focus on resilience and timeliness)?
- Objective
  Mixed-criticality container orchestration
- Methodology
  - *Model the system of a mixed-criticality cluster*
  - *Gather functional (placement) and non-functional (resilience and timeliness) requirements from literature*
  - *Qualitative evaluations for functional requirements and mixed evaluations for non-functional requirements*
  - *Analyze data (log analysis, regression tests, t-test)*
  - *Understand problems and design solutions to comply with the model*
  - *Perform quantitative measurements on the proposed system*

PhD thesis

Recent research:

- uses Linux real-time containers
- extends Kuberentes to support:

  real-time Linux containers, TSN networking, millisecond-time migration

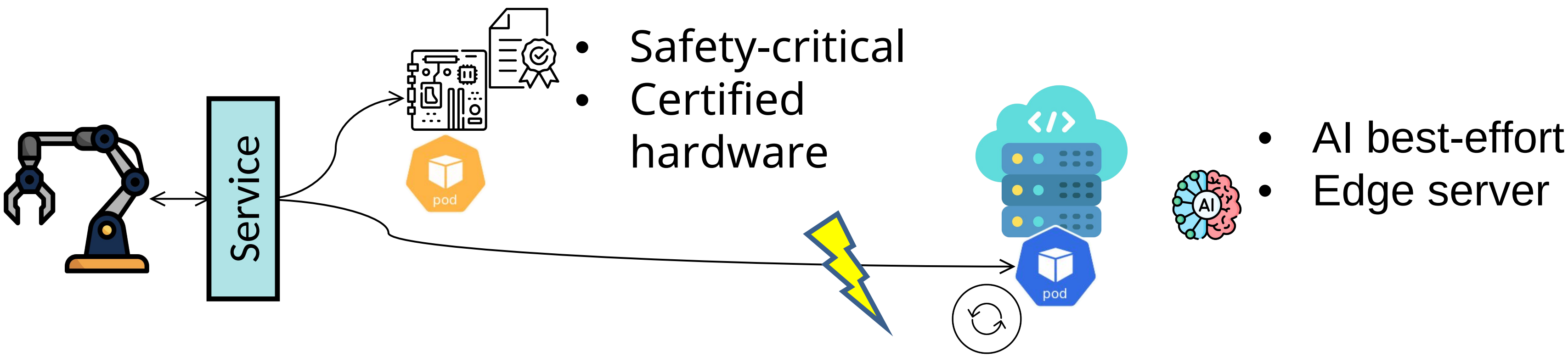
Patches to one-size-fits-all solutions…

Simple to use, but hard to rely on when:

- strict non-functional requirements
- heterogeneous hardware and software

Mixed-criticality clusters

Not all nodes provide the same guarantees
Not all pods are equally important



Need to account for failure probability:

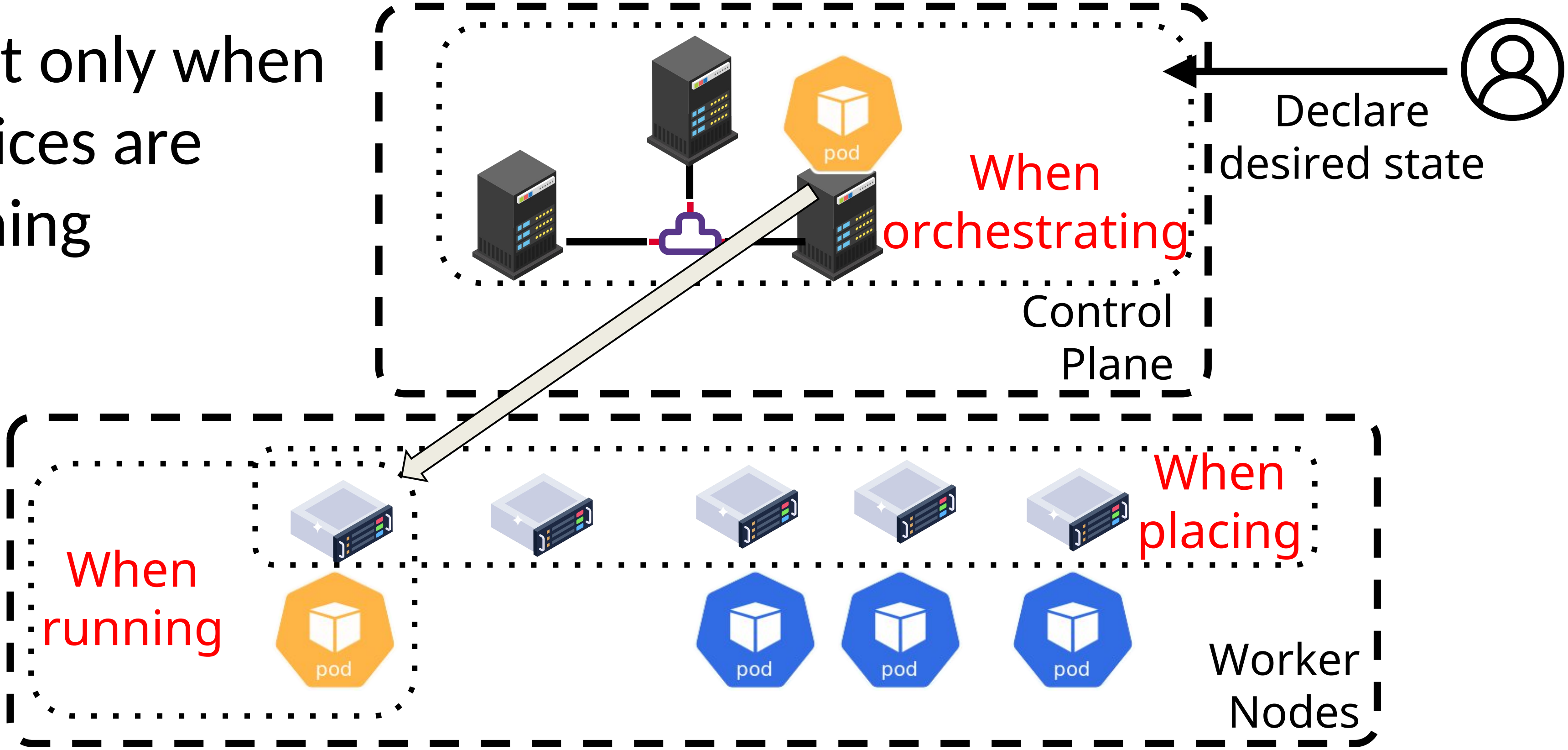
Node assurance *Service & Pod criticality*

- *Diverse replication*
- *Seamless migration*
- *Diversified rolling update*

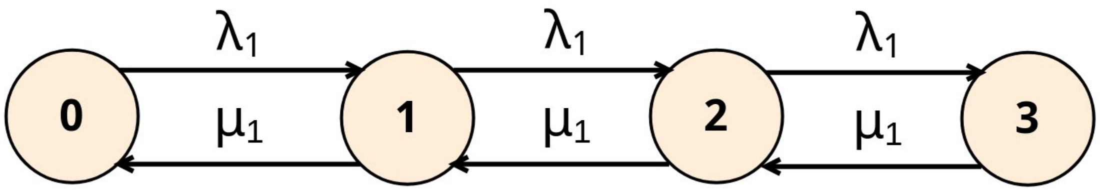
Mixed-criticality orchestration

Need **always** to account for failure probability…

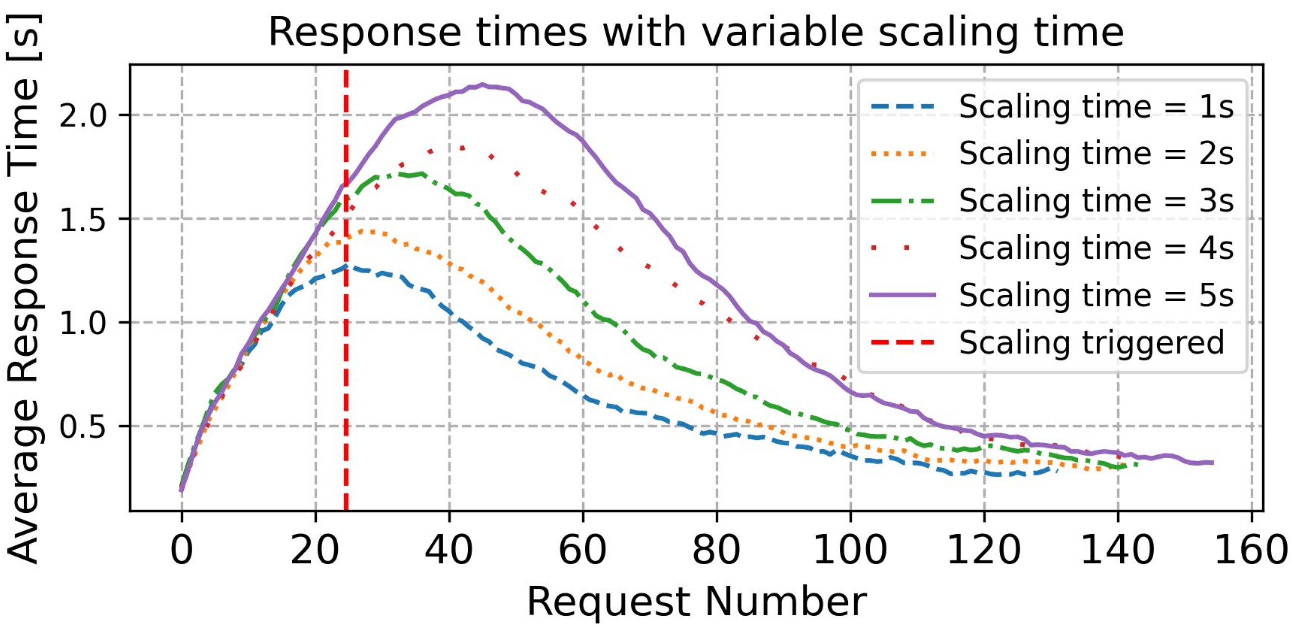
…not only when services are running



Impact of orchestration on services



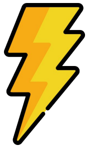
The orchestrator determines the scaling times
The scaling times affect the response times



Failure analysis

Analysis of 81 online Kubernetes-related failures

Design Mutiny!
≈9000 fault injected targeting the cluster state



Evaluate results of fault injection to improve testing

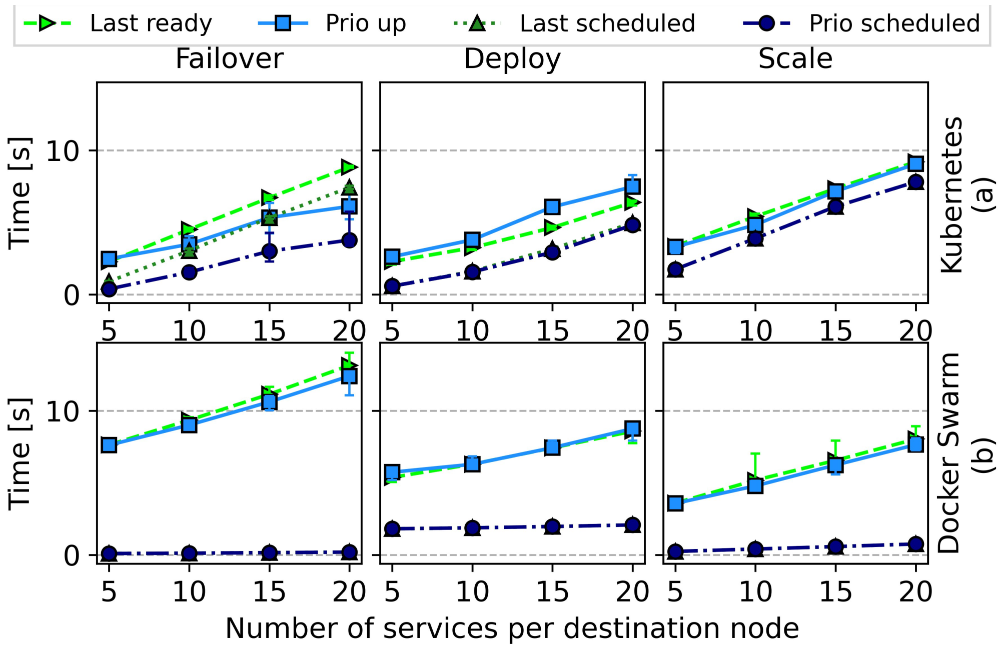
18% of real-world failures were cluster outages*

25% of real-world failures were related to overloads

3% of the injections one error led to system-wide failures

*A significant number or all the running services don't respond

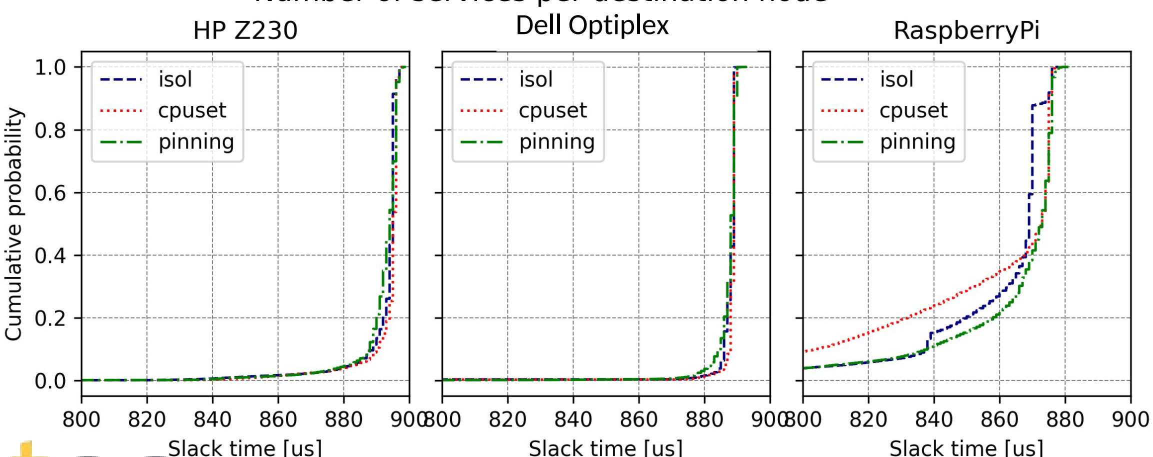
Timing analysis



Orchestrators are not able to prioritize service management

It may take tens of seconds to scale a service or handle a failure

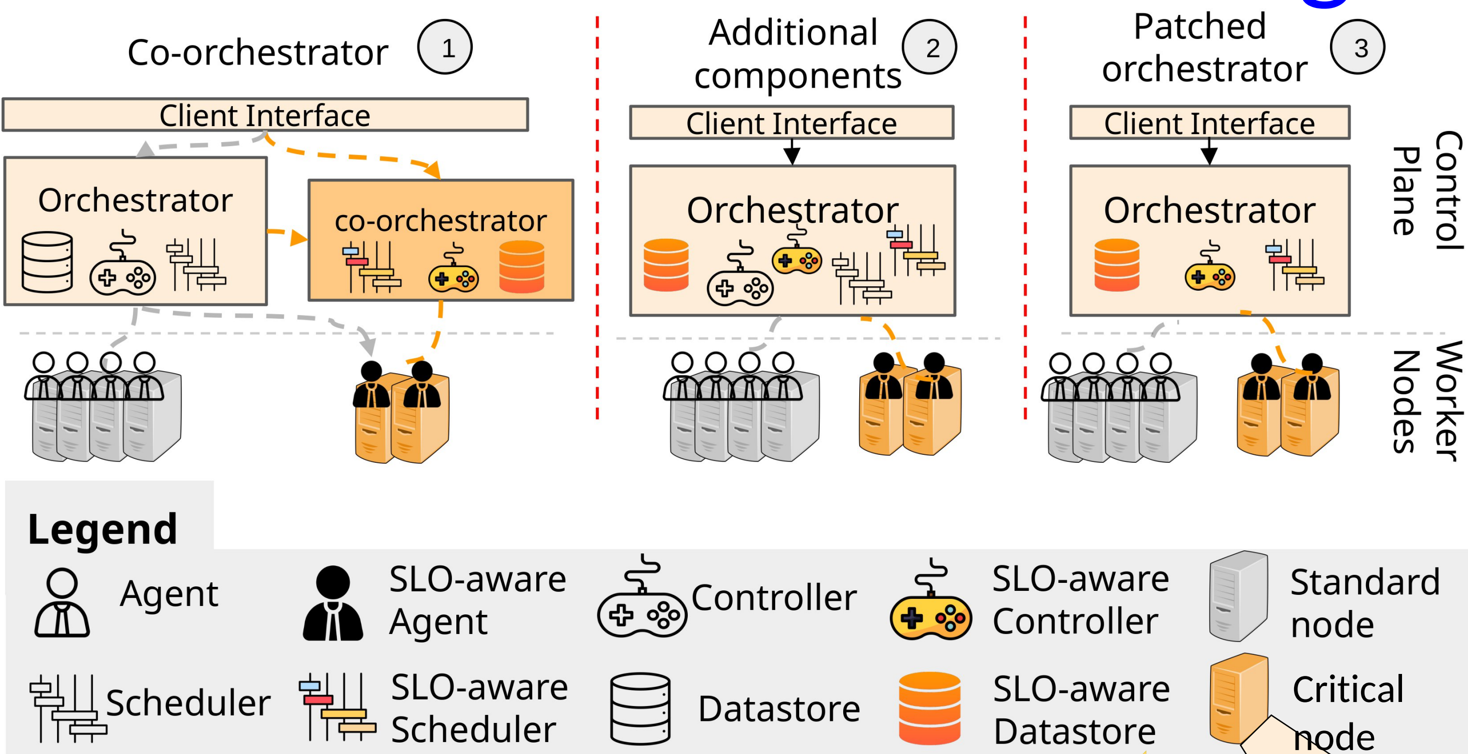
Affects services' response times and availabiltiy



Linux real-time containers provide limited temporal and failure isolation

The behavior hardly depends on the node

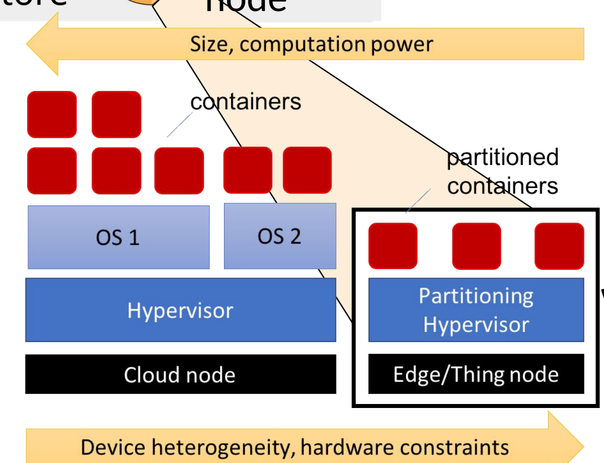
Architectural designs



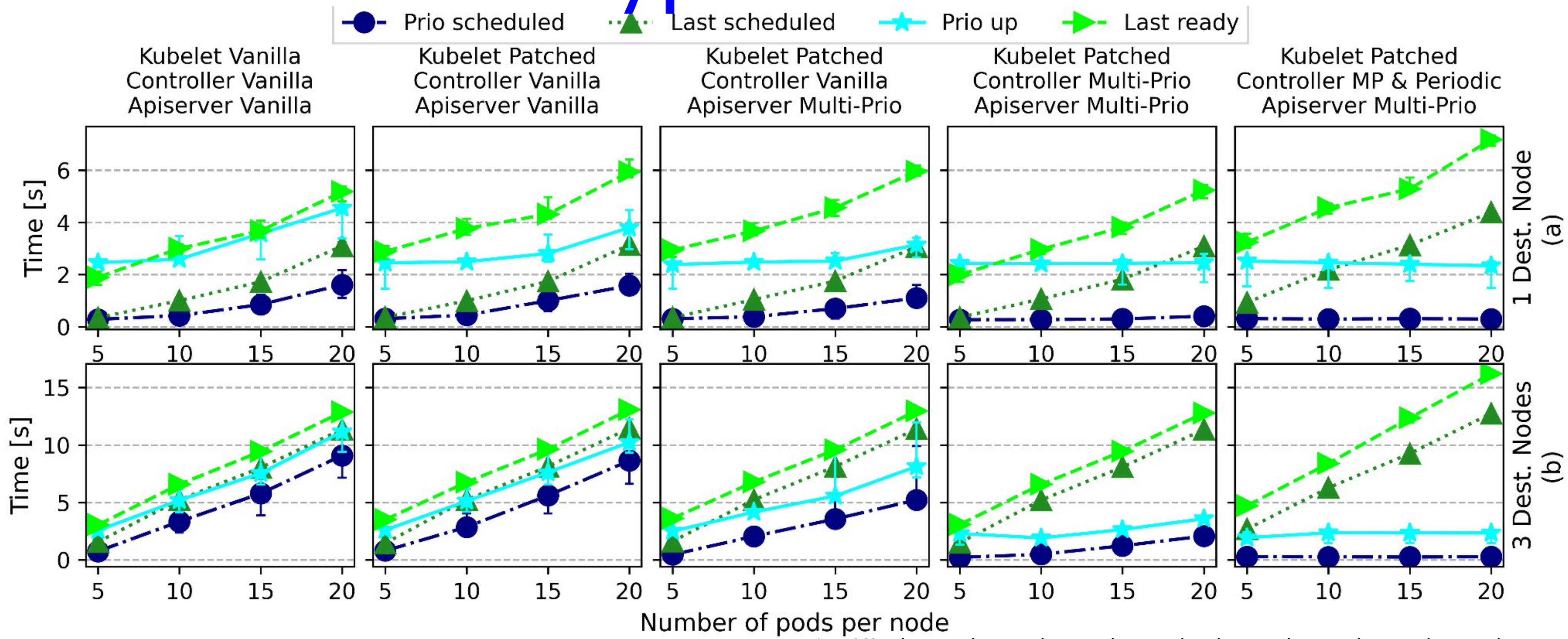
Special purpose hardware for real-time critical tasks

Different degrees of:

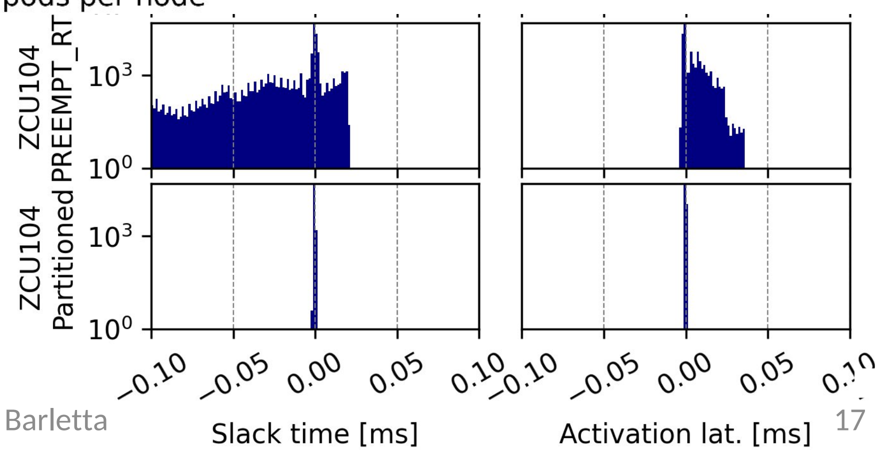
- Isolation from timing interference
- Isolation from failures



Prototype Evaluation



- Bounded orchestration times
- Isolation from interference
- Deterministic execution times



Icon credits

"Icon made by Surang from www.flaticon.com"

"Icon made by Muhammad Ali from www.flaticon.com"

"Icon made by mynamepong from www.flaticon.com"

"Icon made by Stockio from www.flaticon.com"

"Icon made by smashingstocks from www.flaticon.com"

"Icon made by vectorsmarket15 from www.flaticon.com"

"Icon made by juicy_fish from www.flaticon.com"

"Icon made by Freepik from www.flaticon.com"

"Icon made by pause08 from www.flaticon.com"