SOCC ACM Symposium on Cloud Computing

Quadrant: A Cloud-Deployable NF Virtualization Platform

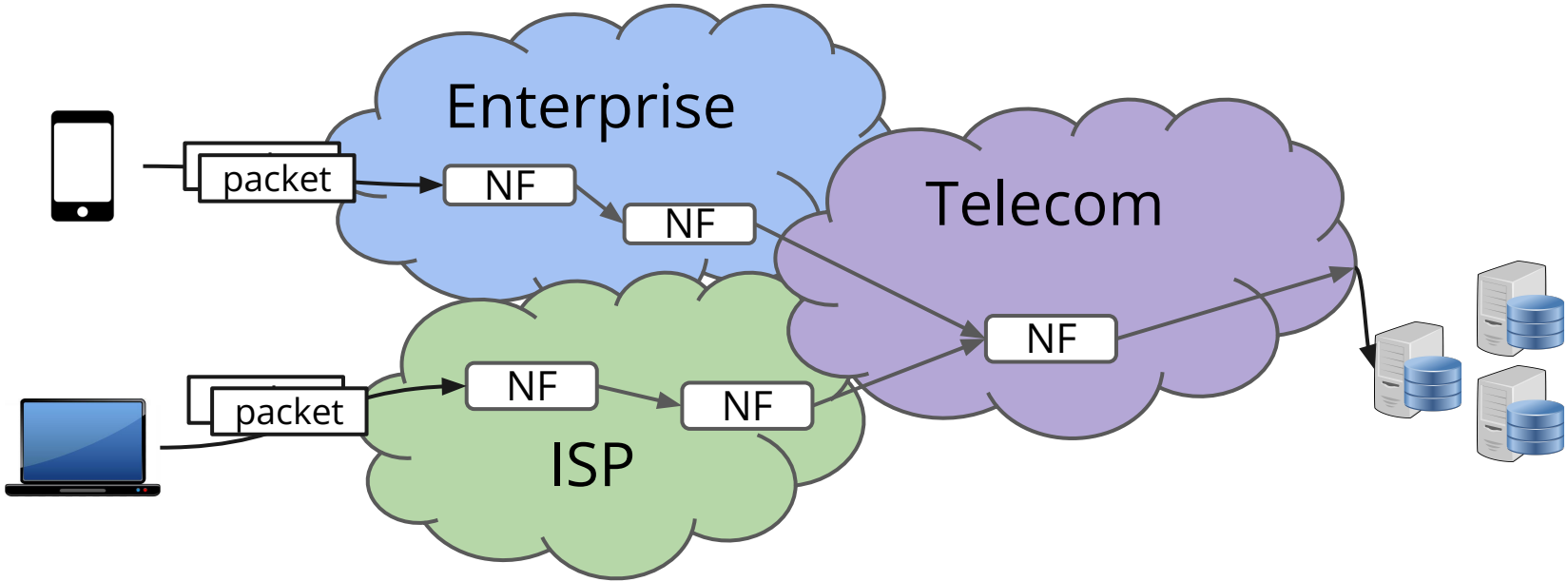
Jianfeng Wang*, Tamás Lévai, Zhuojin Li,
Marcos A. M. Vieira, Ramesh Govindan and Barath Raghavan

USC Viterbi School of Engineering

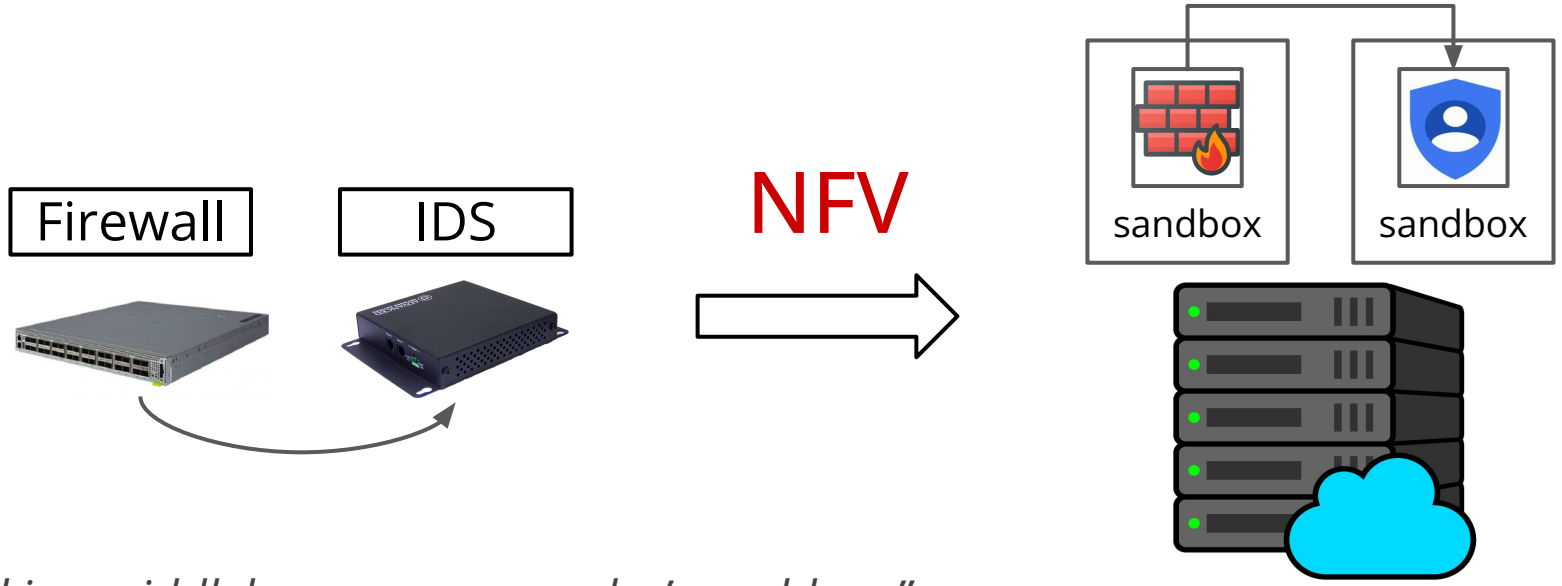
MŰEGYETEM 1782

UFMG UNIVERSIDADE FEDERAL DE MINAS GERAIS

Network Functions (NFs) Perform Packet Processing Tasks

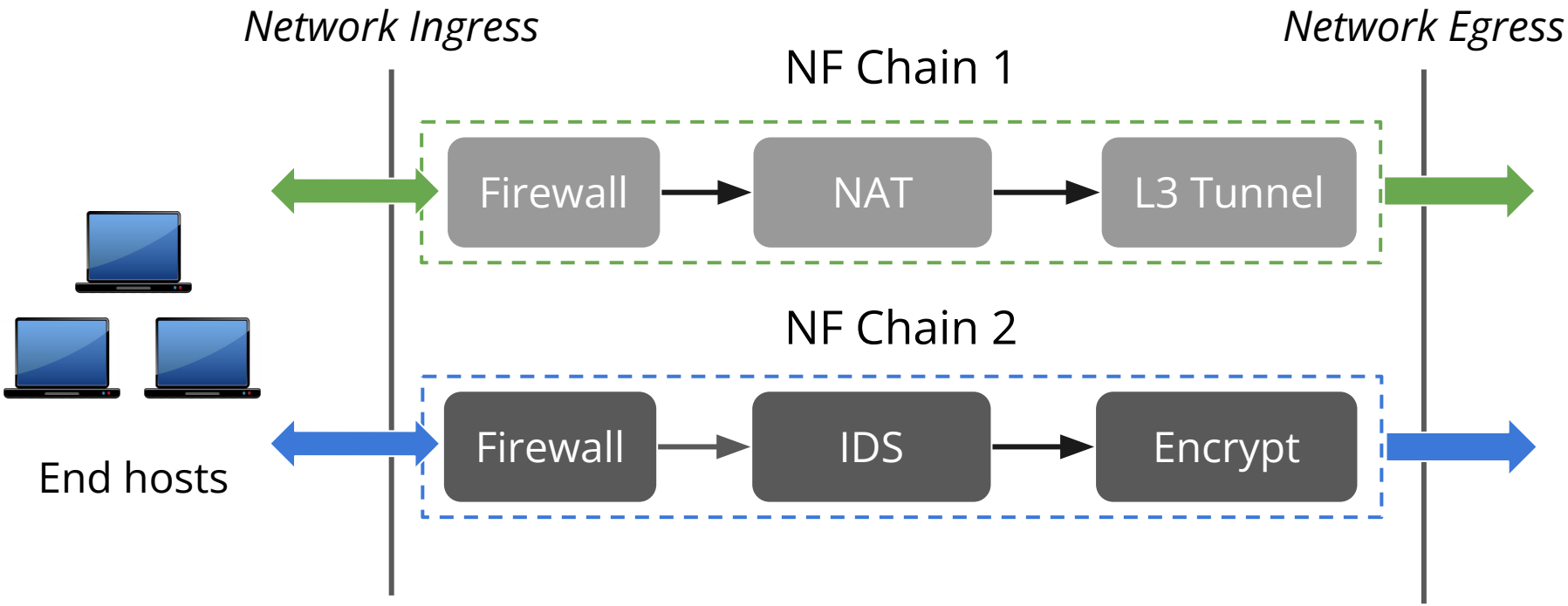


Network Function Virtualization (NFV) Replaces Hardware Middleboxes With Software NFs



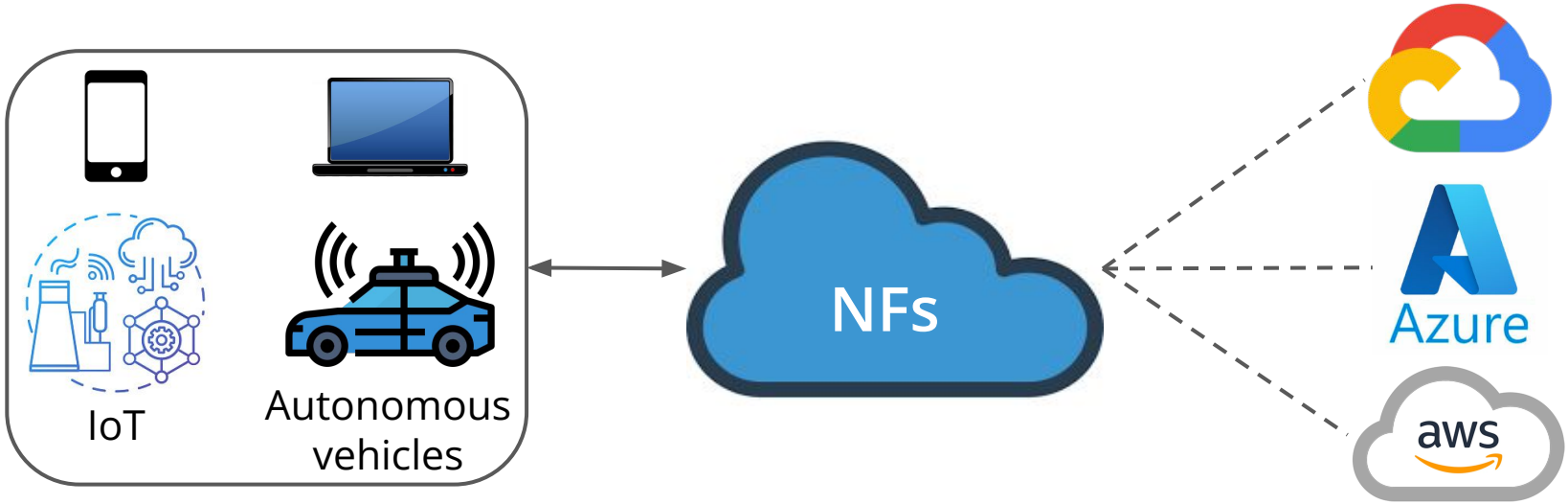
"Making middleboxes someone else's problem."
Aplomb [SIGCOMM '12])

NF Chain: A Sequence of NFs Applied To Traffic



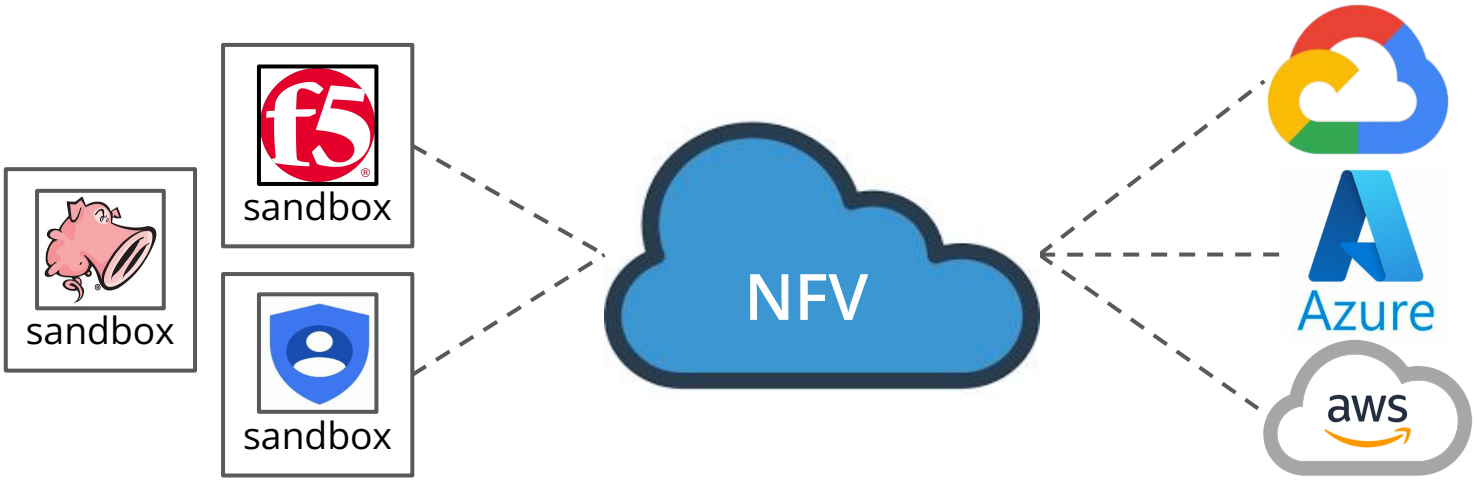
NFV Is Moving Towards Cloud Deployments

E.g. 5G moves towards cloud-hosted cellular NFs



Cloud-deployability Is Becoming A Necessity

To achieve *performance*, *generality*, and *ease of deployment* in the commodity cloud context

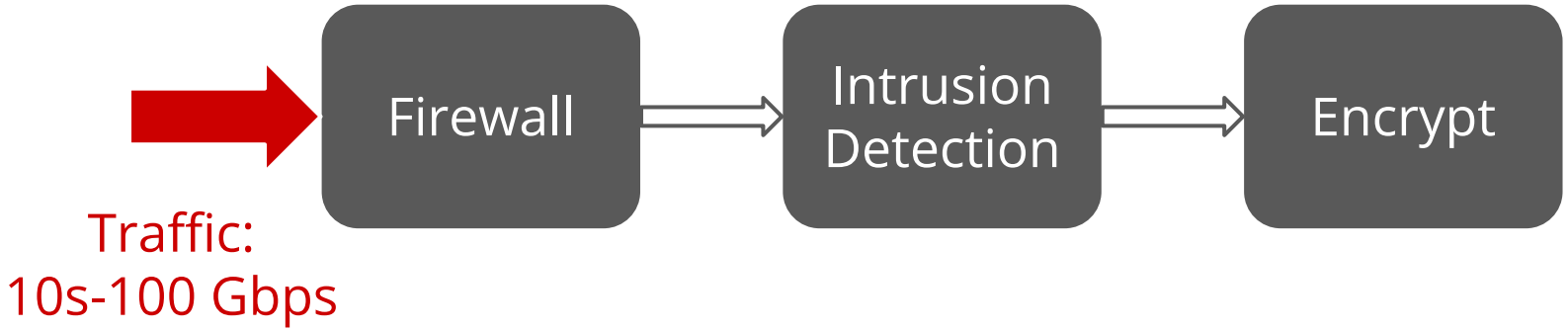


Third-party NFs From Vendors

Commodity Cloud Infra

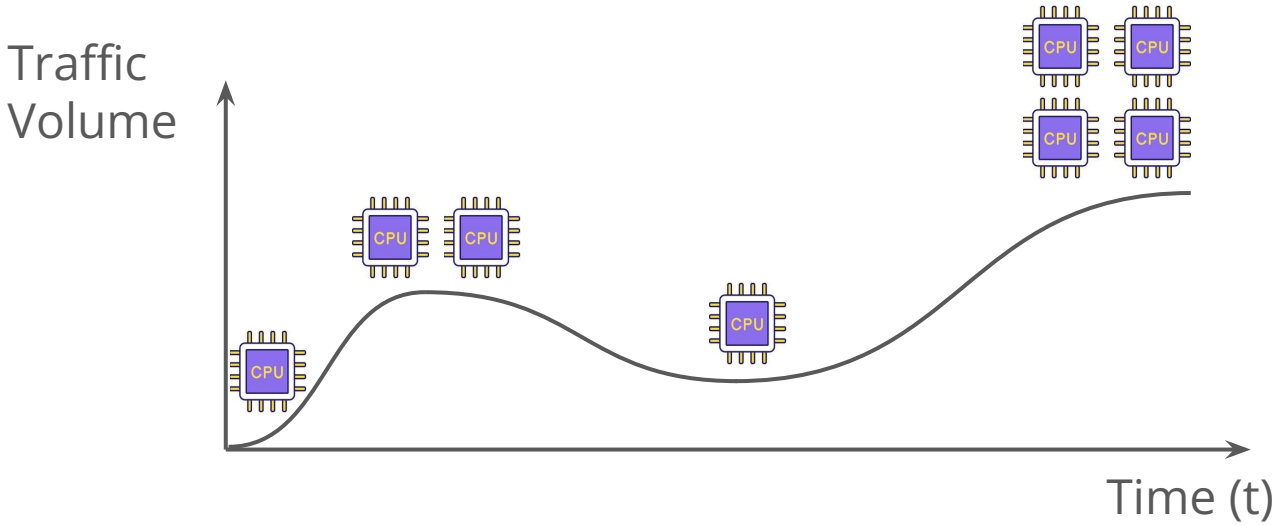
Cloud-deployable NFV Requirements

1. High-performance packet processing
   a. Line-rate throughput



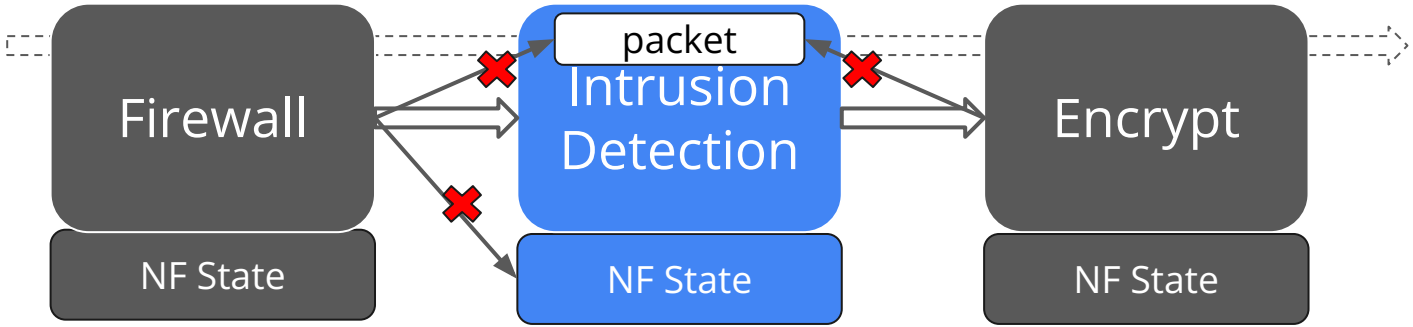
Cloud-deployable NFV Requirements

1. High-performance packet processing
2. Scaling of NF chains
    a. Throughput and latency SLO-adherence



Cloud-deployable NFV Requirements

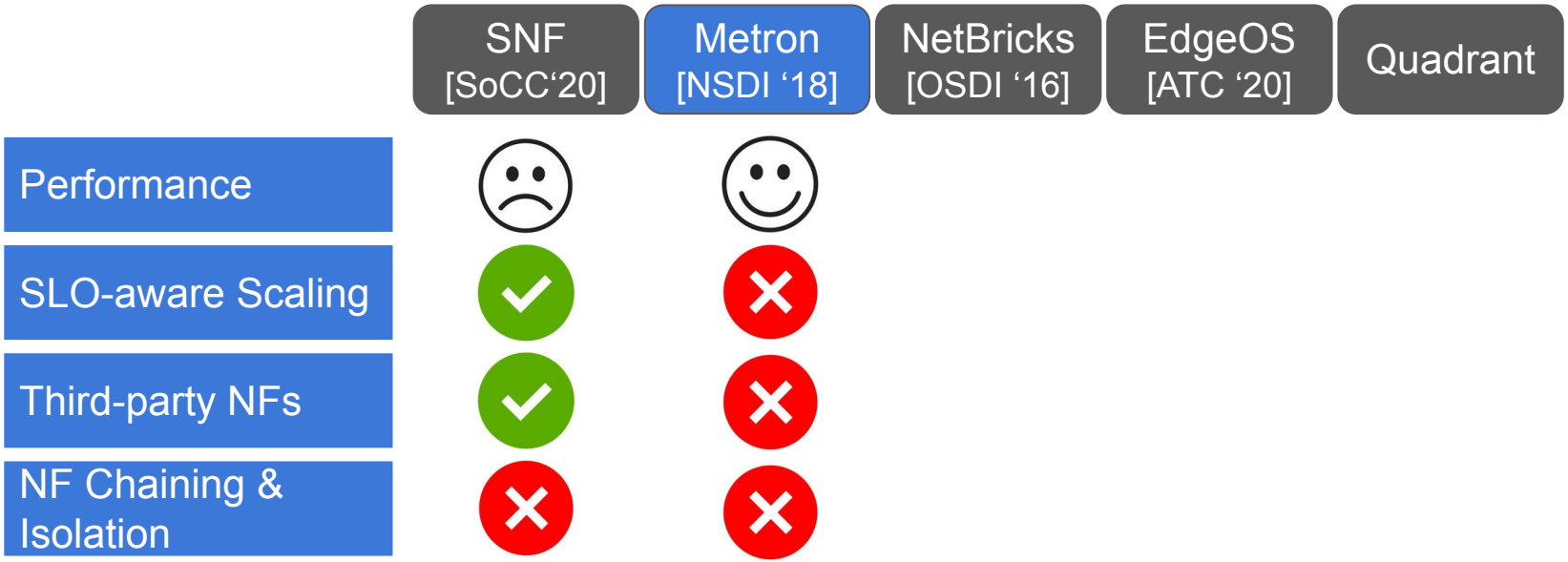
1. High-performance packet processing
2. Scaling of NF chains
3. NF Memory and packet isolation
   a. Each NF has exclusive access to internal state and packets being processed



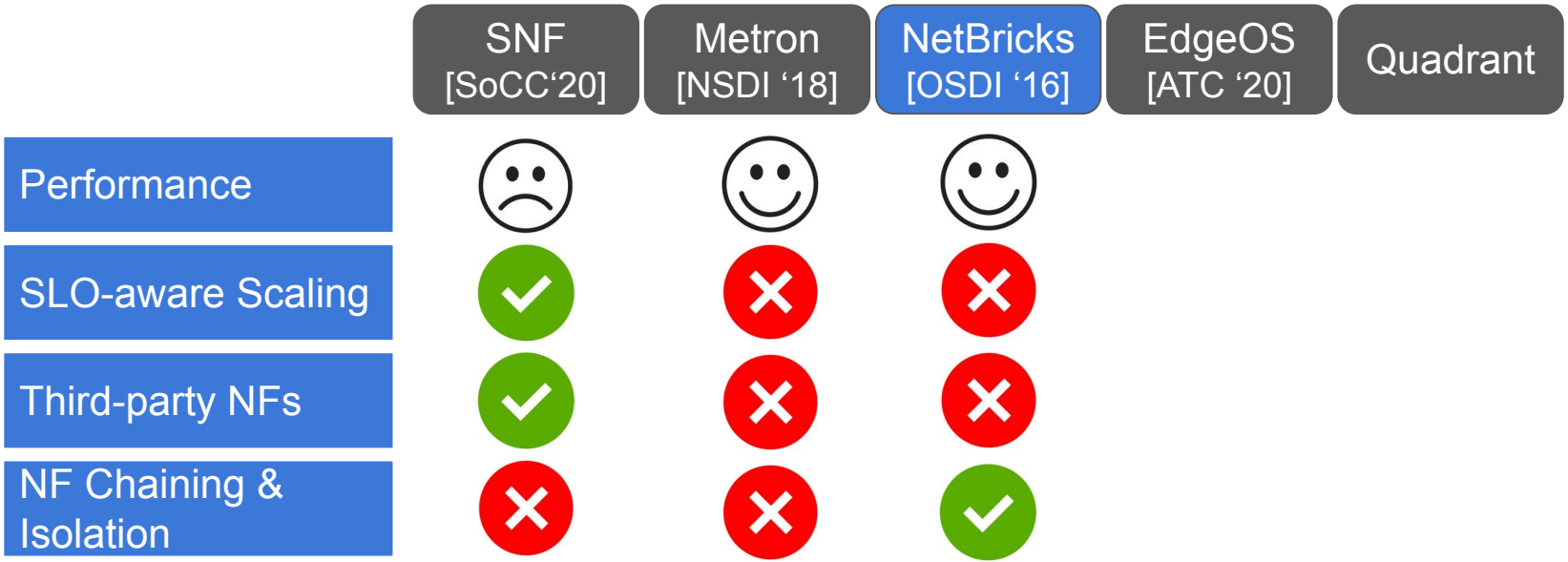
Comparing Cloud-deployable NFV Properties

	SNF [SoCC'20]	Metron [NSDI '18]	NetBricks [OSDI '16]	EdgeOS [ATC '20]	Quadrant
Performance					
SLO-aware Scaling					
Third-party NFs					
NF Chaining & Isolation					

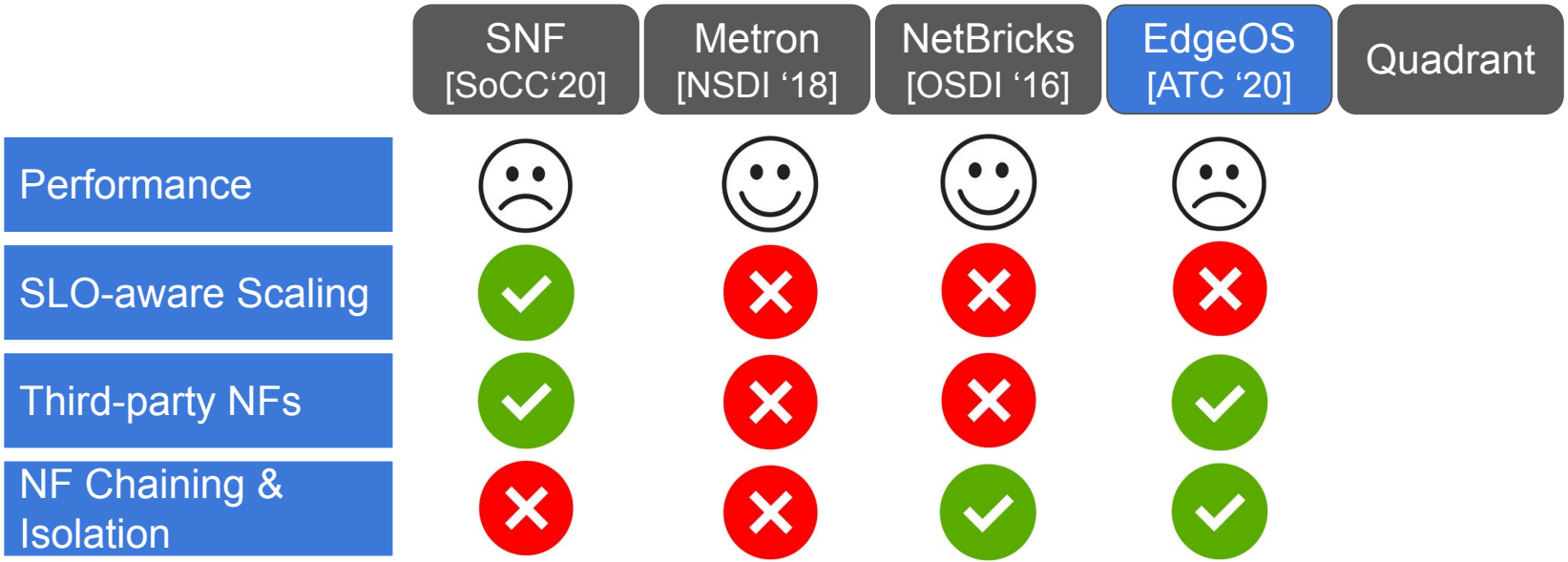
Comparing Cloud-deployable NFV Properties



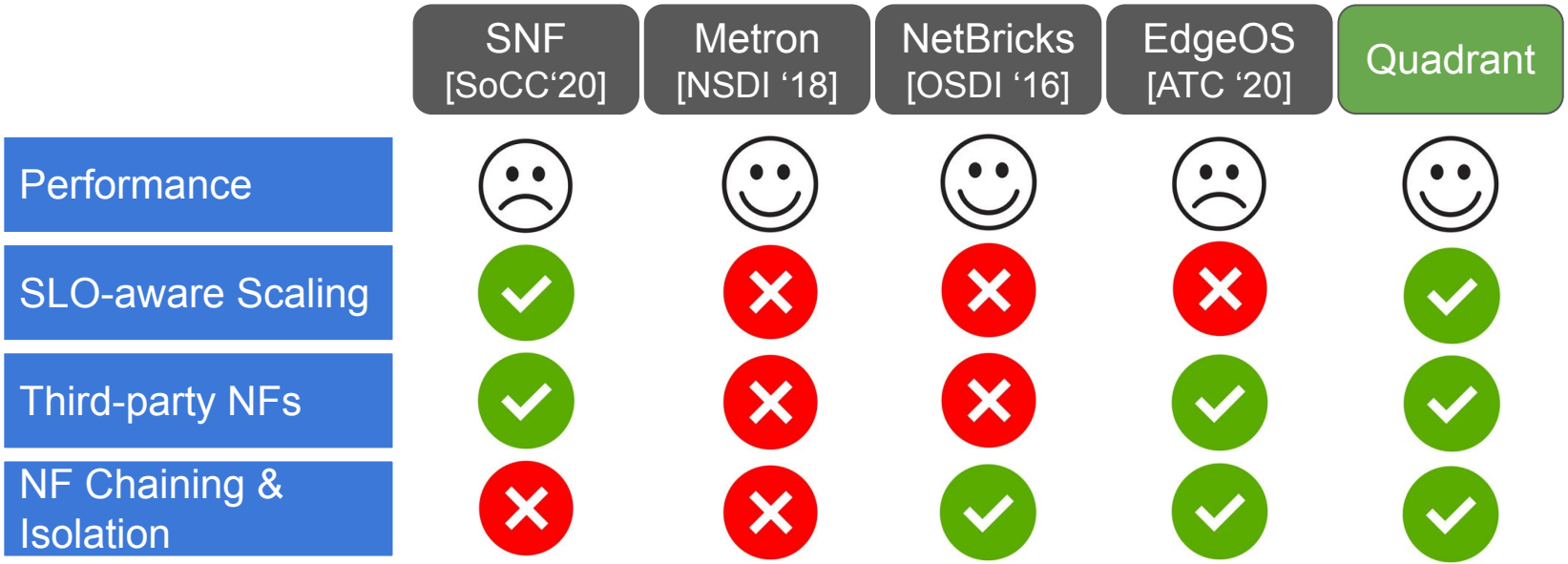
Comparing Cloud-deployable NFV Properties



Comparing Cloud-deployable NFV Properties



Comparing Cloud-deployable NFV Properties



Goal: Achieving Cloud-deployability In NFV

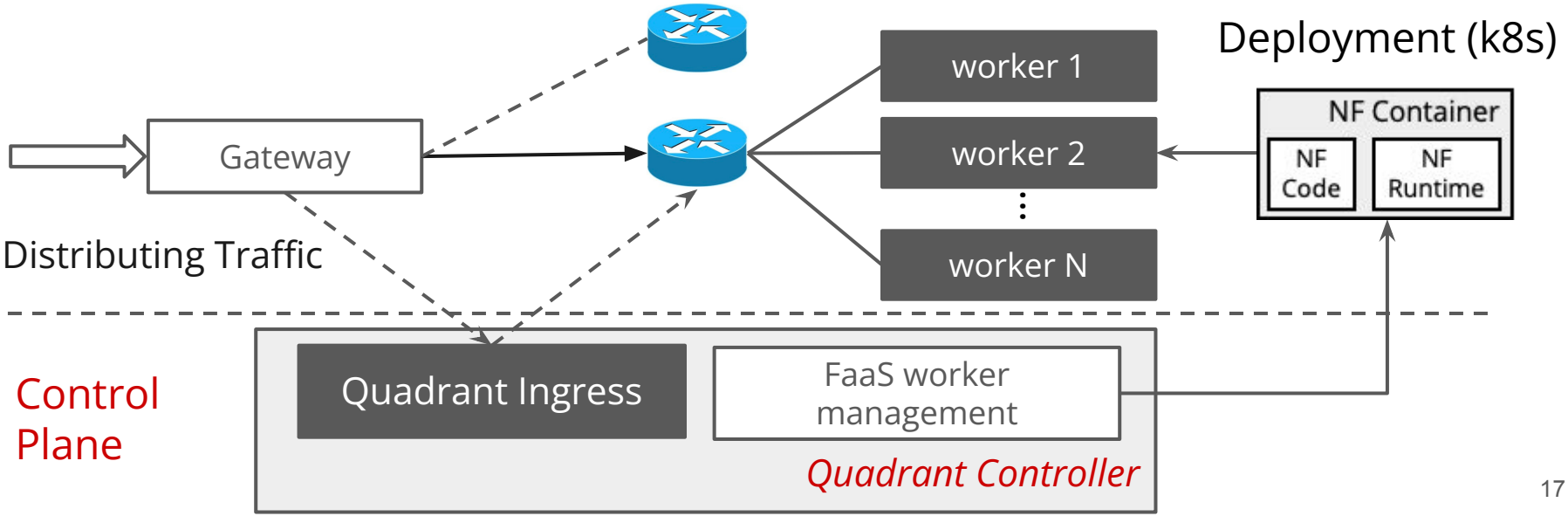
Adapt existing cloud infrastructure to achieve NF chaining, scaling and isolation with high-performance and generality

Quadrant's Contributions

1. High-performance spatiotemporal packet isolation
2. Performance-aware NF scheduling
3. SLO-aware auto-scaling

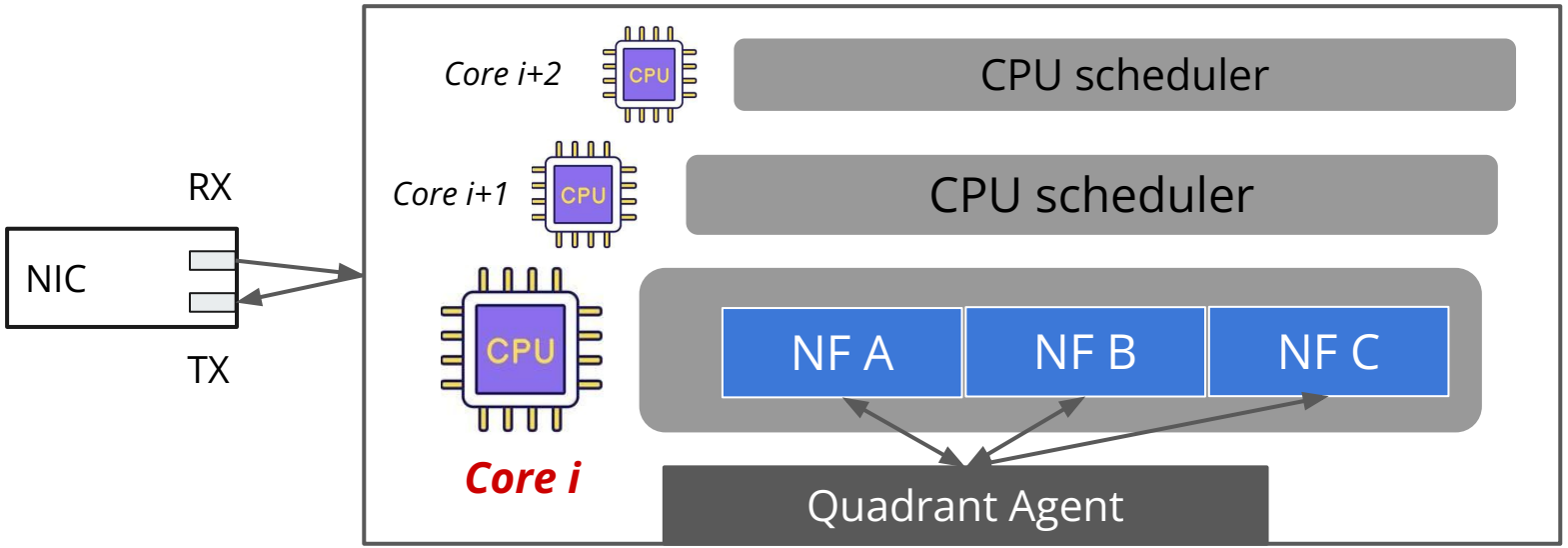
Quadrant Design Overview: Cluster-level

Quadrant deploys NFs as containers, and is built on top of OpenFaaS, reusing Kubernetes, Linux kernel, NICs and switches



Quadrant Design Overview: Server-level

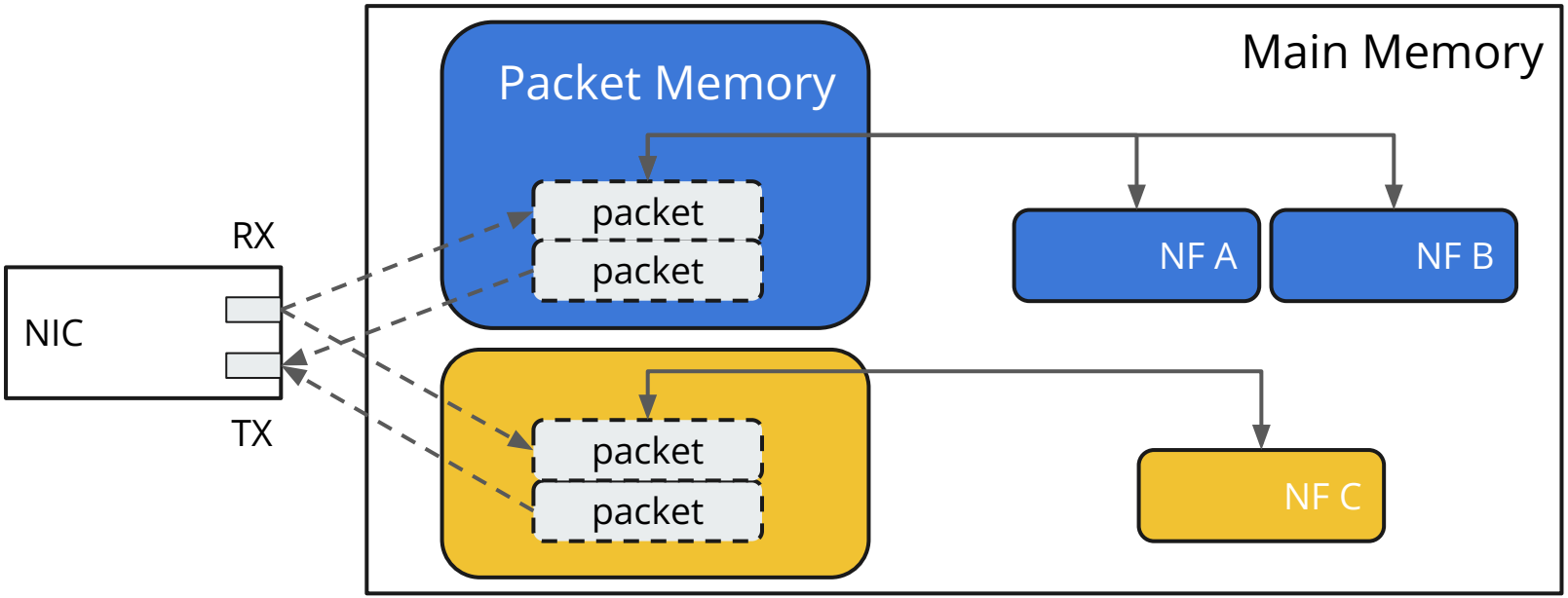
Each Quadrant worker dedicates a core to each chain, applies run-to-completion scheduling to each chain



Quadrant's Contributions

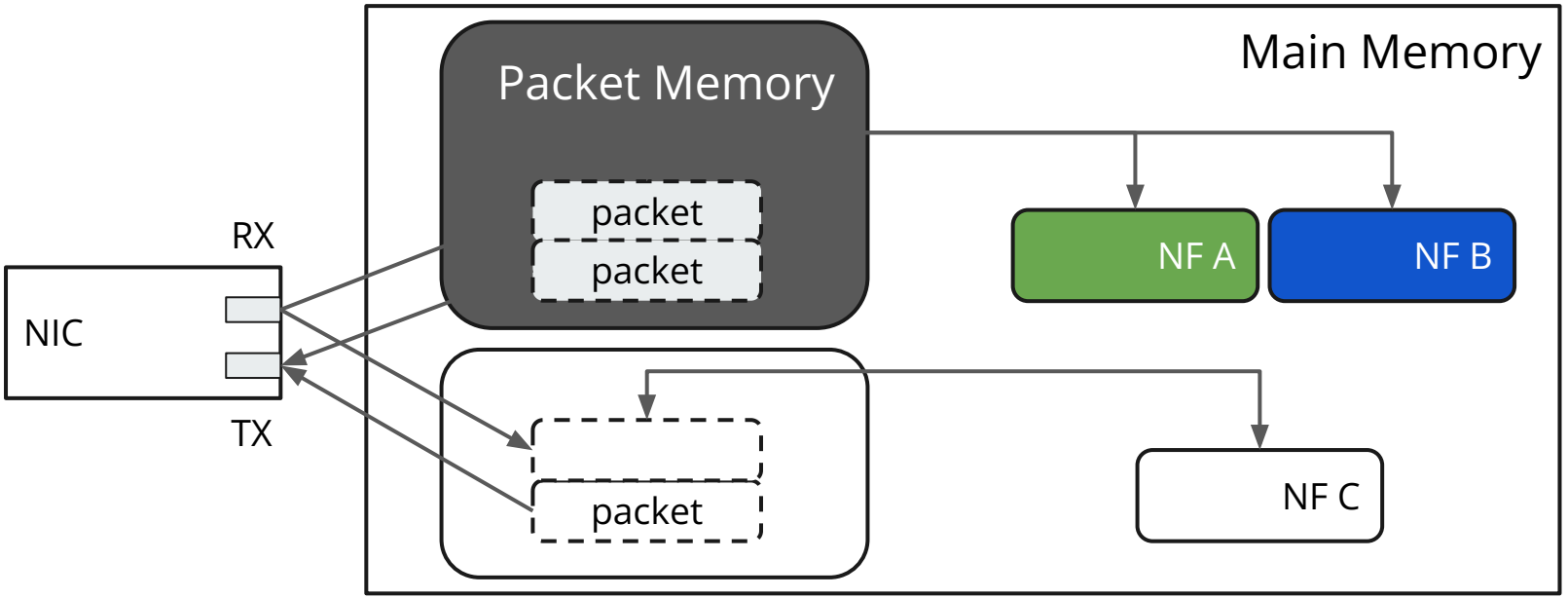
1. High-performance spatiotemporal packet isolation
2. Performance-aware NF scheduling
3. SLO-aware auto-scaling

Our Observation: NFs Can Share Packet Memory When In The Same NF Chain



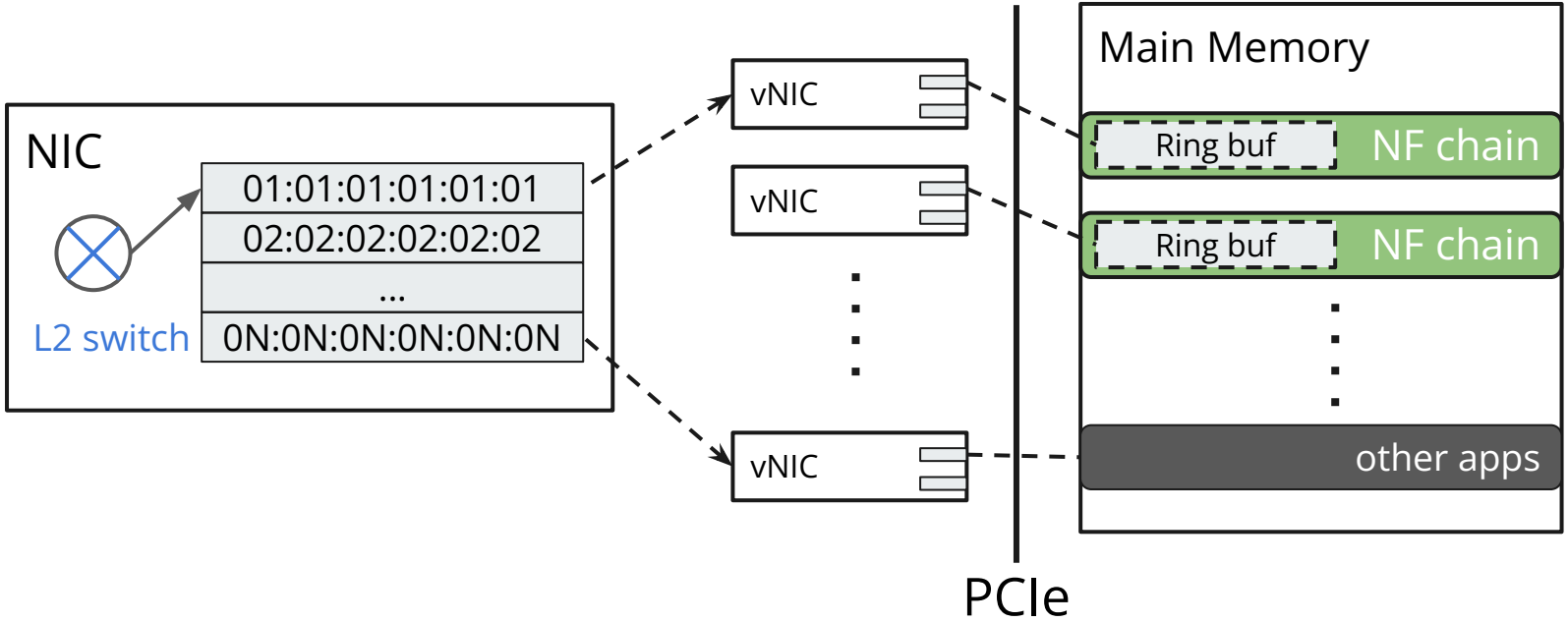
NF Memory And Packet Isolation

NFs are containerized processes with unique NF state memory



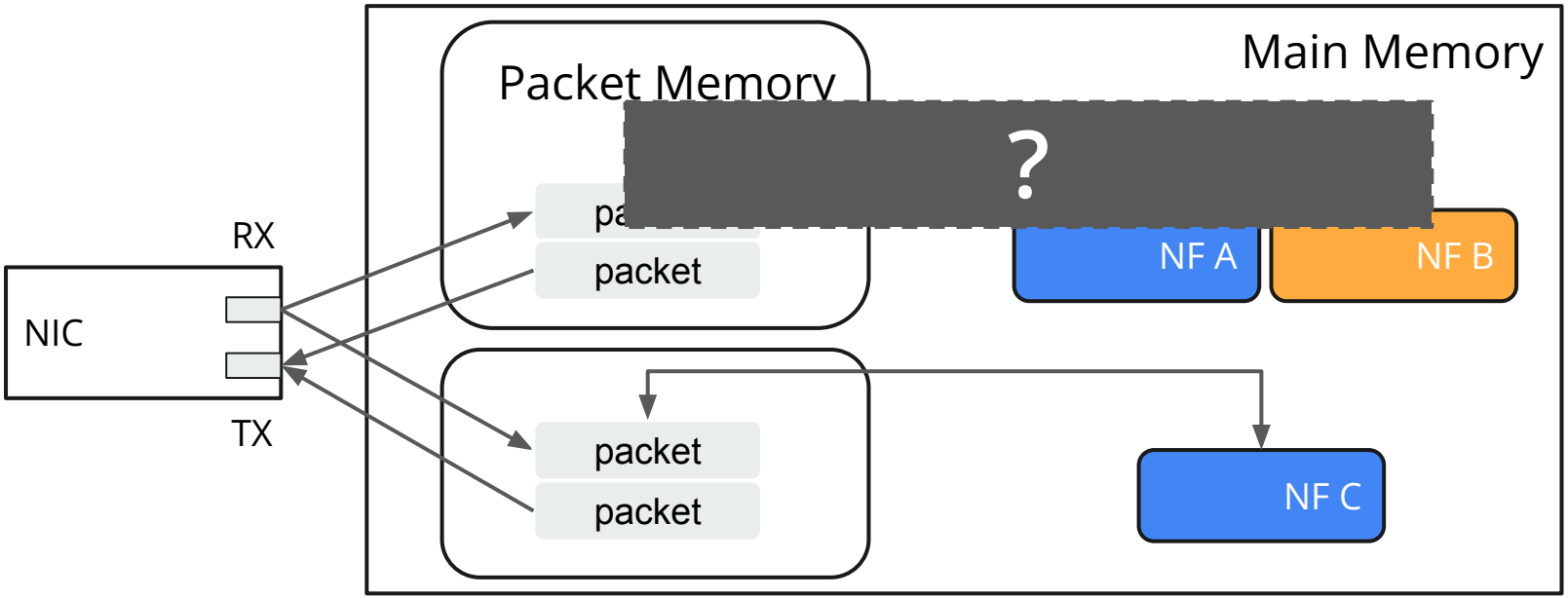
Design: Spatiotemporal Packet Isolation

- Per-chain packet memory: *SR-IOV + per-packet L2 tagging at ToR*



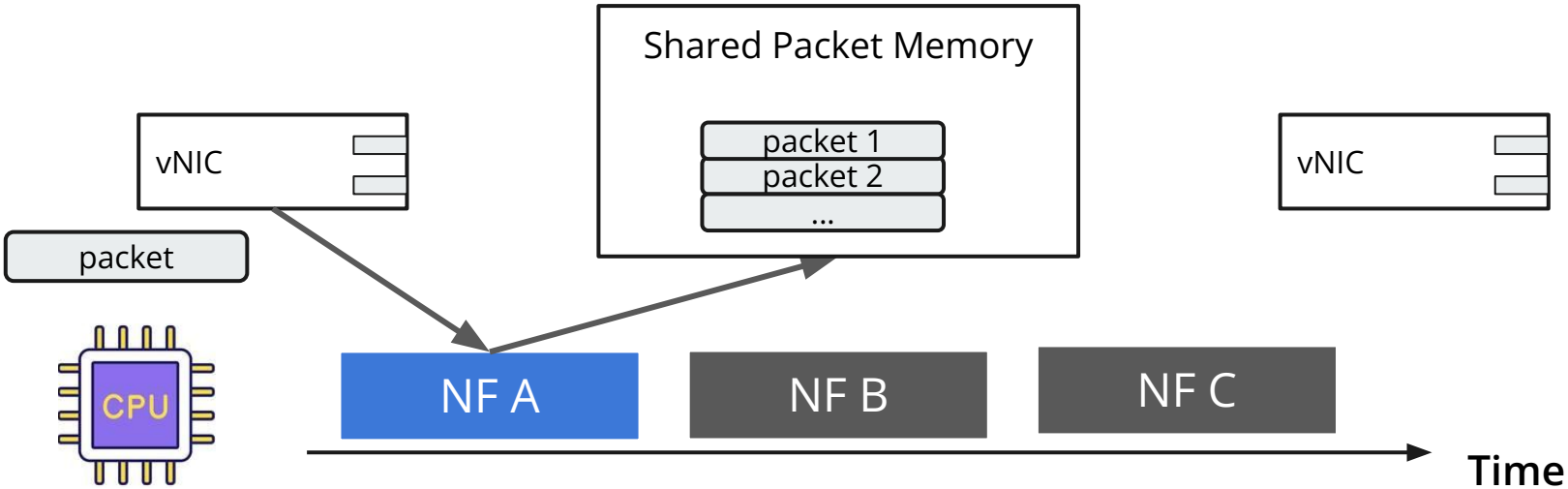
NF Memory And Packet Isolation

How can NFs have unique access to the shared memory?



Design: Spatiotemporal Packet Isolation

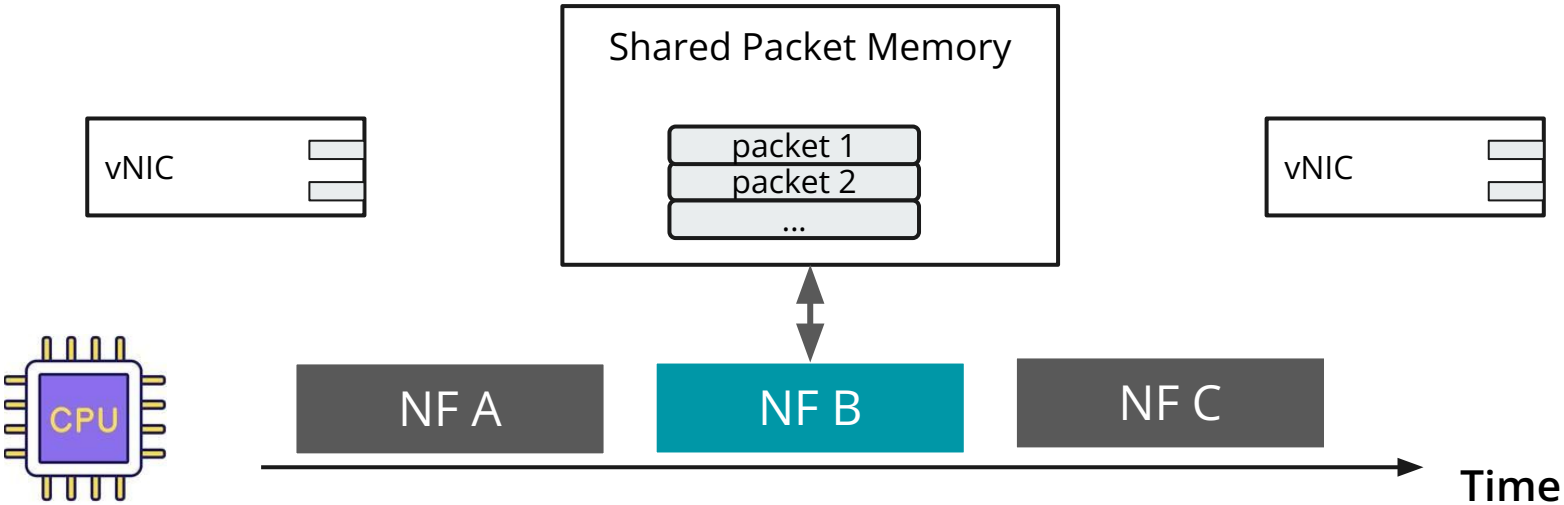
- Controlled memory accesses via *Cooperative Scheduling*



A cooperative scheduler runs at each core

Design: Spatiotemporal Packet Isolation

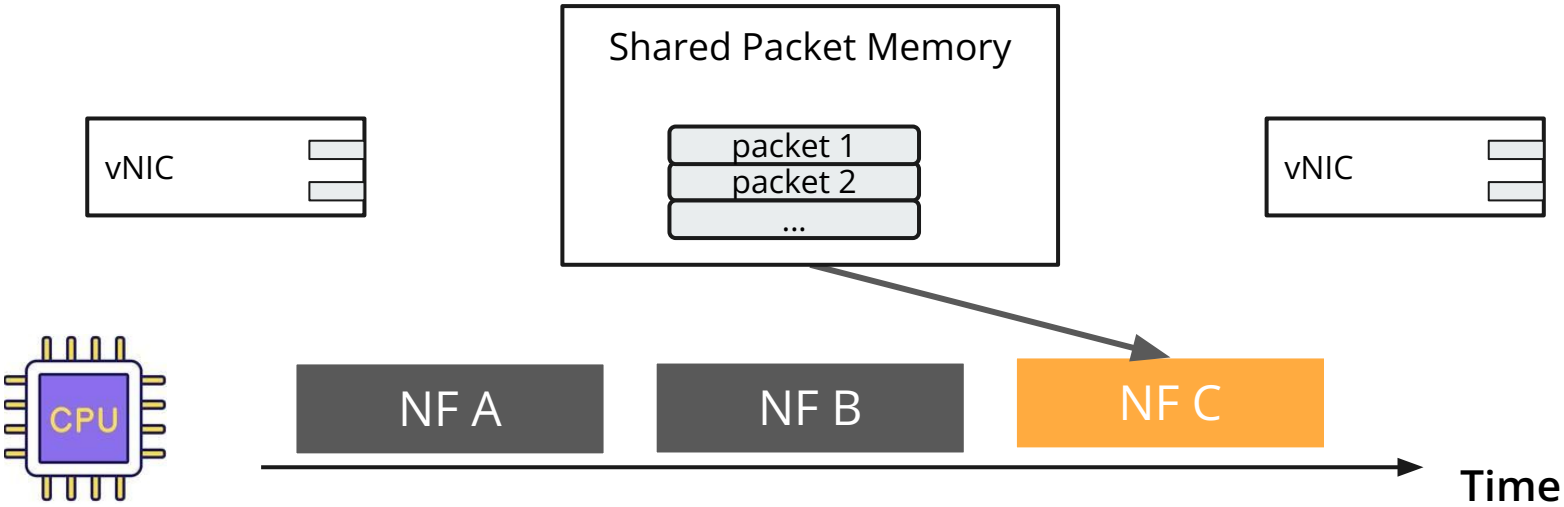
- Controlled memory accesses via *Cooperative Scheduling*



A cooperative scheduler runs at each core

Design: Spatiotemporal Packet Isolation

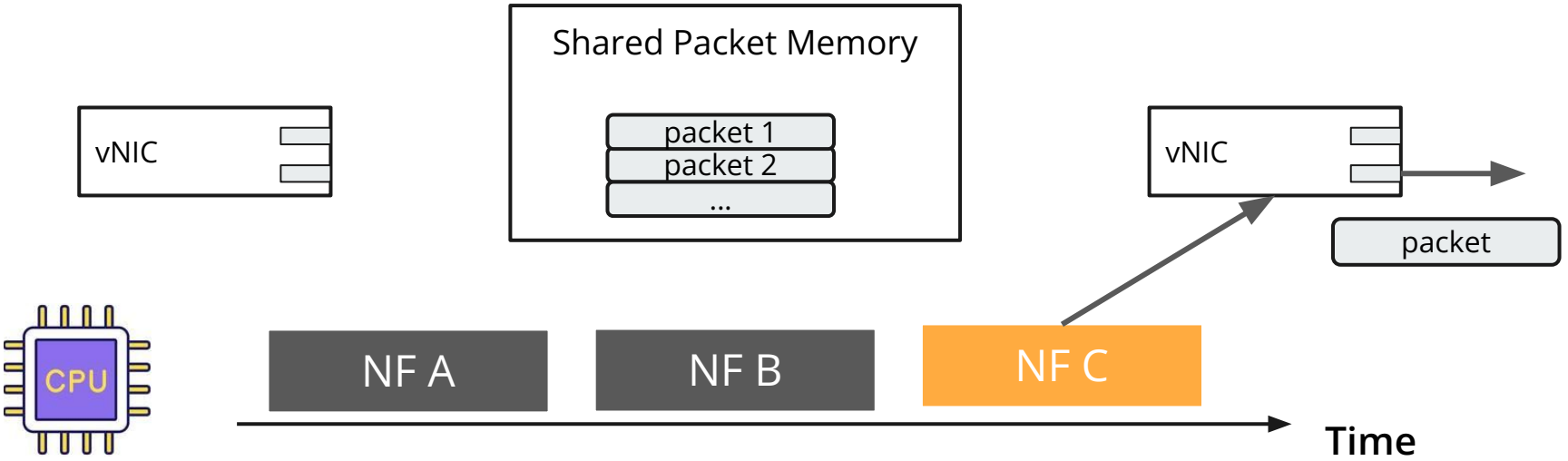
- Controlled memory accesses via *Cooperative Scheduling*



A cooperative scheduler runs at each core

Design: Spatiotemporal Packet Isolation

- Controlled memory accesses via *Cooperative Scheduling*



Quadrant reduces expensive software packet copying

Quadrant's Contributions

1. High-performance spatiotemporal packet isolation
2. Performance-aware NF scheduling
3. SLO-aware auto-scaling

Evaluation

- (Deployability) How many LoCs does Quadrant require?
- (Isolation) How does Quadrant's NF isolation mechanism perform?
- (Scaling) Can Quadrant adapt to dynamic traffic and meet SLOs?
- (Overhead) What are different overheads in Quadrant?
- (Failure) How long does Quadrant recover a failed chain?
- (Ingress) How can Quadrant's ingress mask control-plane latency?
- (Sensitivity) How sensitive is Quadrant to design parameters?

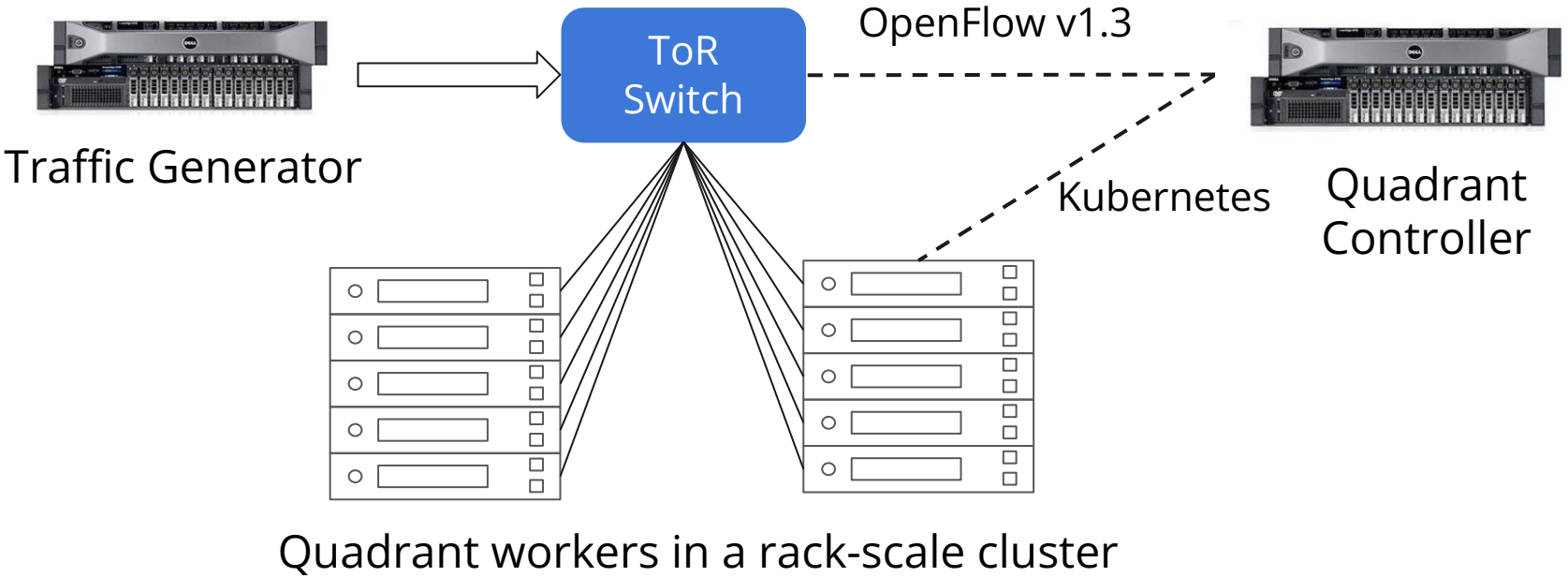
Evaluation

- (Isolation) How does Quadrant's isolation mechanism perform compared to state-of-the-art NFV platforms?

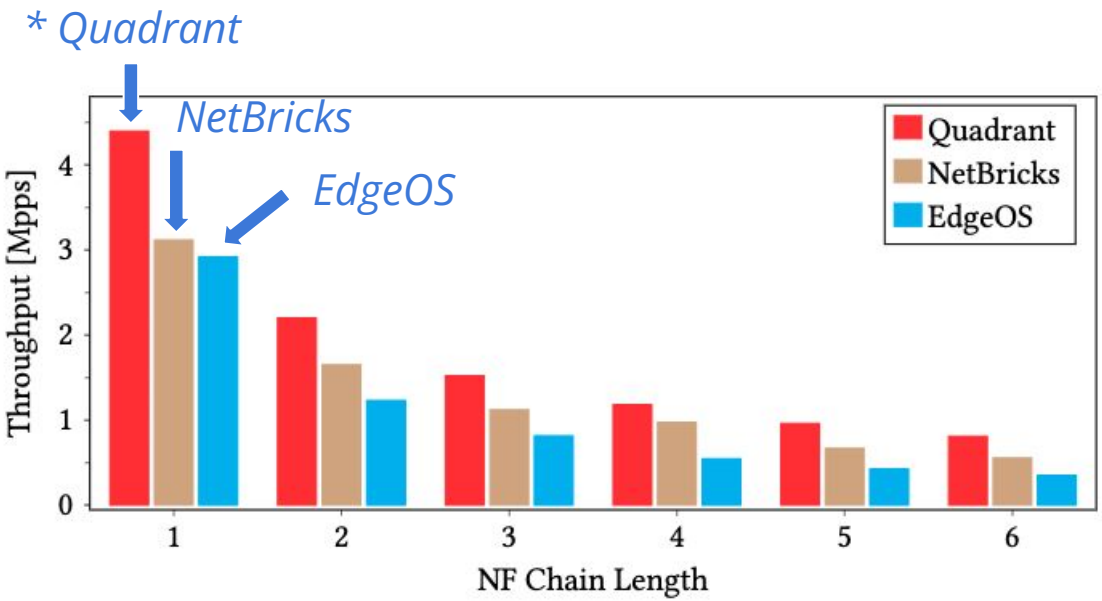
Cloudlab: Build Scientific Infrastructure for Research:
Link: https://www.cloudlab.us/

Implementation

Quadrant is built on top of OpenFaaS (+2.7% LoC)



Isolation: Comparing Per-core Throughput

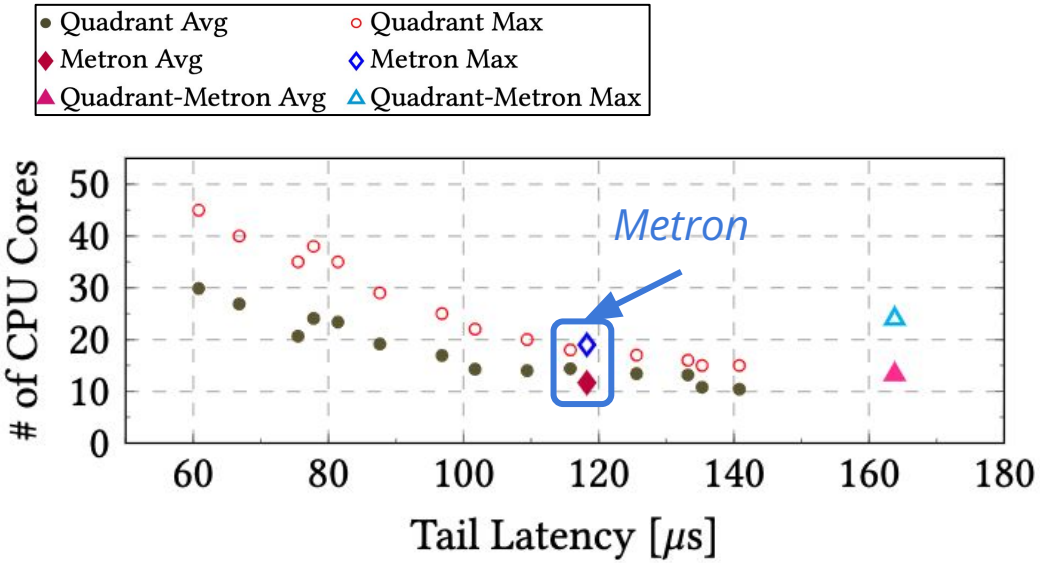


Quadrant outperforms:

- NetBricks: Rust
  - By 1.2 - 1.5x
- EdgeOS: packet copying
  - By 1.6 - 2.3x

Quadrant outperforms SOTA NFV platforms with different isolation mechanisms

Scaling Performance: CPU Core Usage



Quadrant:

- *Avg.: 14.4* cores (+23%)

Metron (no isolation):

- *Avg.: 11.7* cores

Quadrant achieves SLO-aware scaling,
enabling trade-off between latency and efficiency

Quadrant Summary

- Supports NFV in cloud with commodity software and hardware
- Proposes high performance spatiotemporal packet isolation
  - Lightweight inter-NF isolation for third-party NFs
- Proposes SLO-aware auto-scaling
  - Enabling flexible latency-efficiency tradeoffs
  - Less CPU core usage compared to alternatives

Thank you!

Email: jianfenw@usc.edu

Quadrant: A Cloud-Deployable NF Virtualization Platform

Jianfeng Wang, Tamás Lévai, Zhuojin Li,
Marcos A. M. Vieira, Ramesh Govindan and Barath Raghavan

USC Viterbi School of Engineering

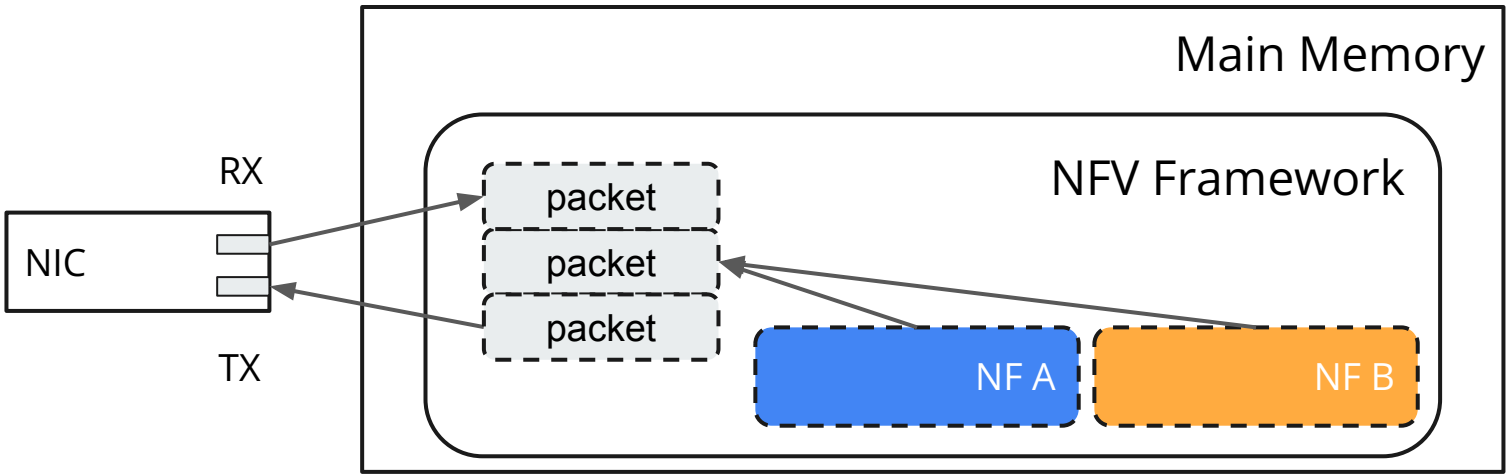
MŰEGYETEM 1782

UFMG UNIVERSIDADE FEDERAL DE MINAS GERAIS

Additional Slides

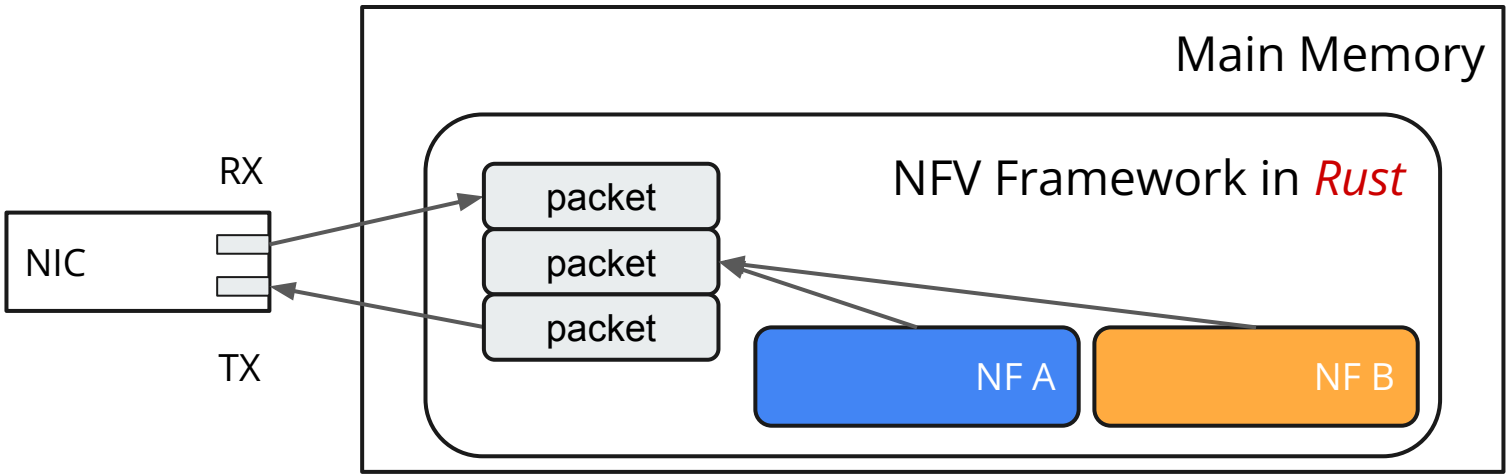
Chaining And Isolation: Limitations of Prior Work

1. High performance but without isolation
   a. Metron [NSDI ‘18]



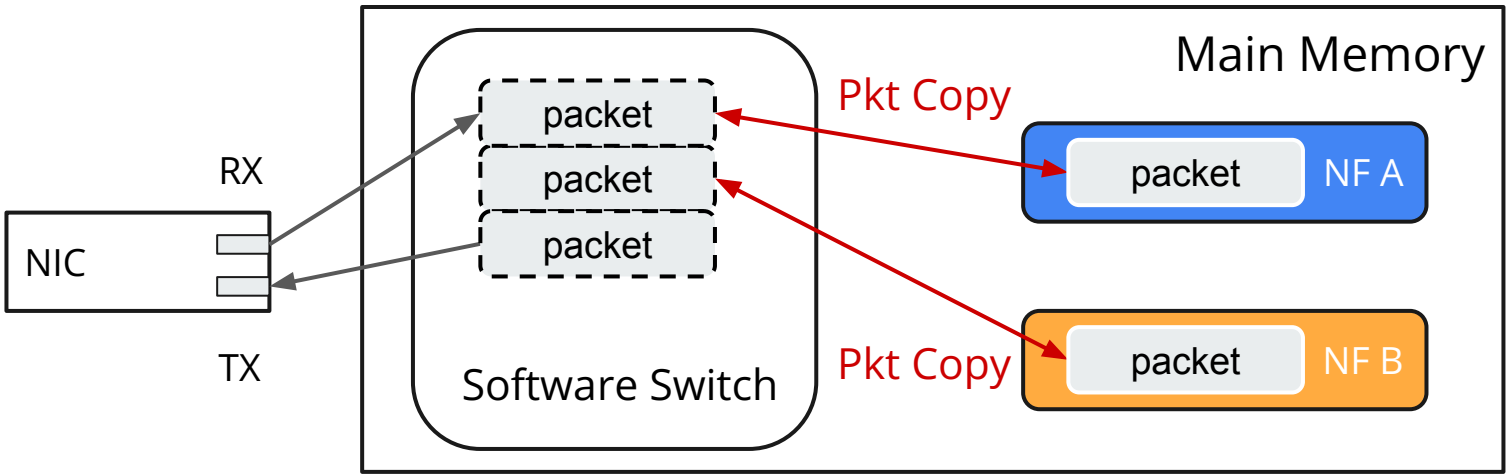
Chaining And Isolation: Limitations of Prior Work

1. High performance but without isolation
2. Isolation without generality / performance
   a. NetBricks [OSDI ‘16], SafeBricks [NSDI ‘18]: using Rust



Chaining And Isolation: Limitations of Prior Work

1. High performance but without isolation
2. Isolation without generality / performance
   a. EdgeOS [ATC '20]: copying packets per NF-hop



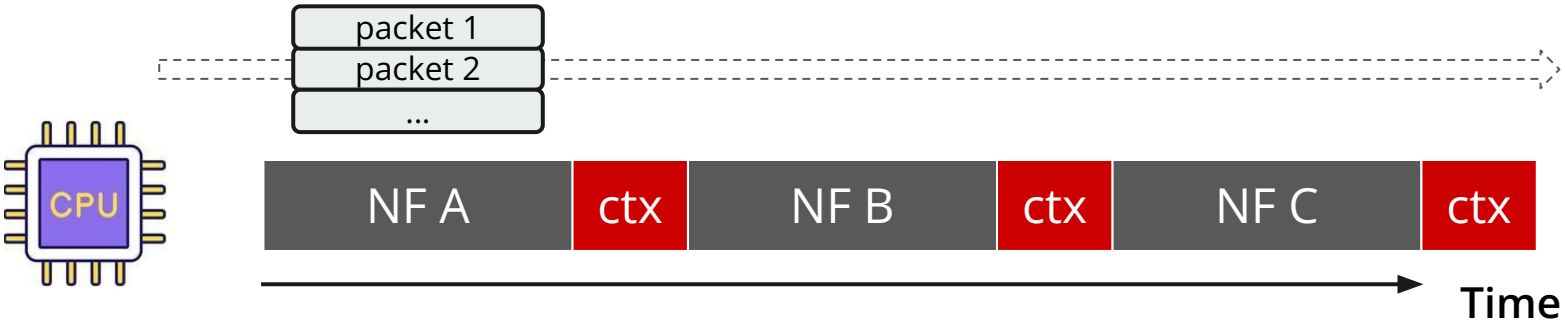
Quadrant's Contributions

1. High-performance spatiotemporal packet isolation
2. Performance-aware NF scheduling
3. SLO-aware auto-scaling

Design: Performance-aware Scheduling

Problem: frequent thread context switches in Cooperative Scheduling

- Key Idea: adjust batch size to bound context switch overhead

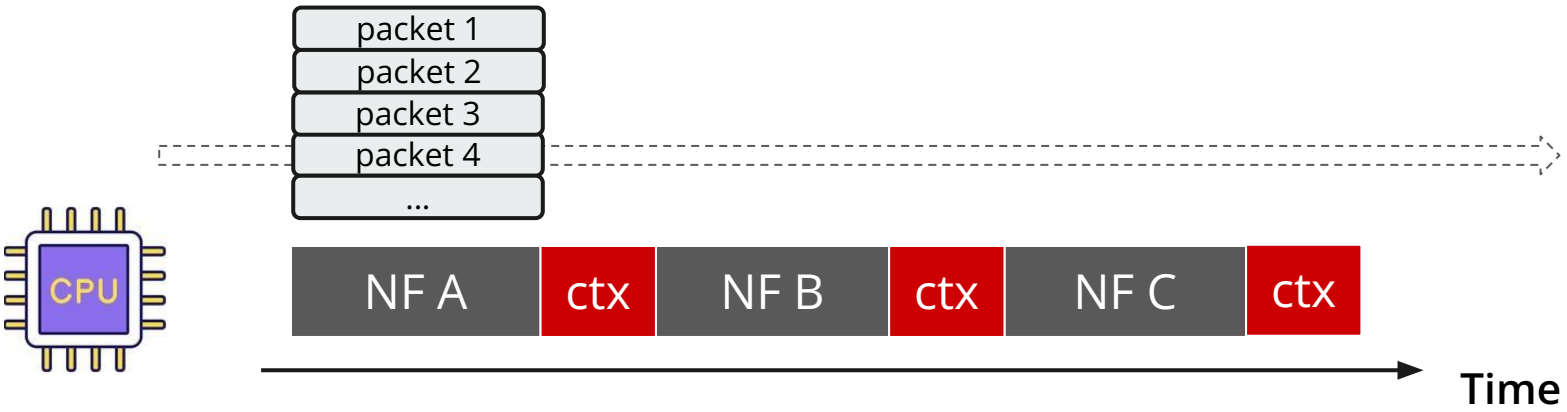


Quadrant executes an NF chain in *run-to-completion* mode.

Design: Performance-aware Scheduling

Problem: frequent thread context switches in Cooperative Scheduling

- Key Idea: adjust batch size to bound context switch overhead



Large batch size when (NF processing time) / (ctx time) decreases

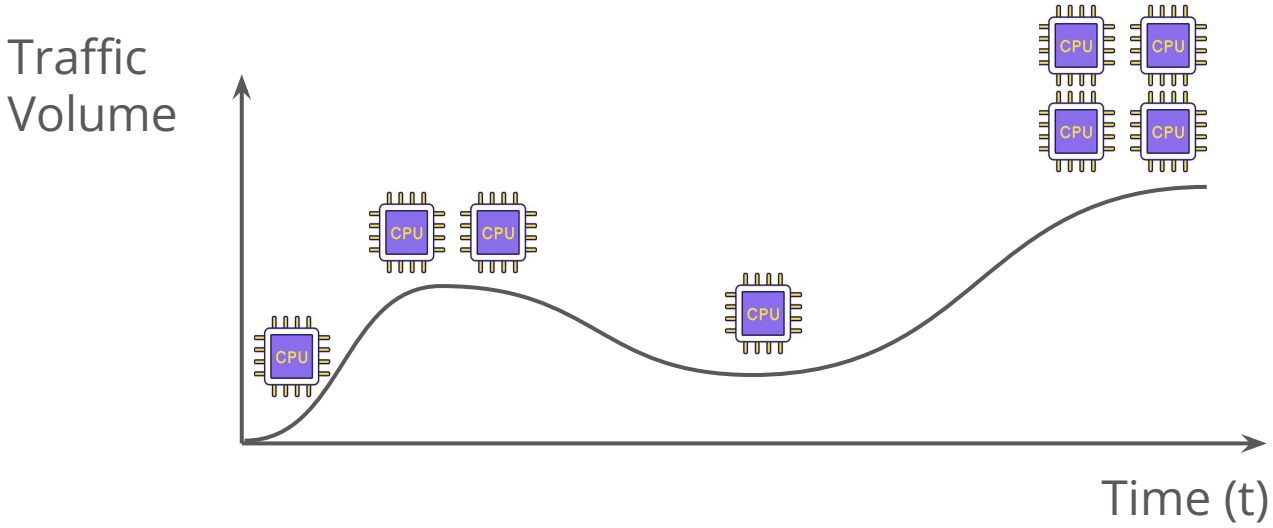
Quadrant's Contributions

1. High-performance spatiotemporal packet isolation
2. Performance-aware NF scheduling
3. SLO-aware auto-scaling

Quadrant Design: SLO-aware Auto-scaling

Auto-scaling is to adapt resources allocated to each NF chain in response to traffic dynamics

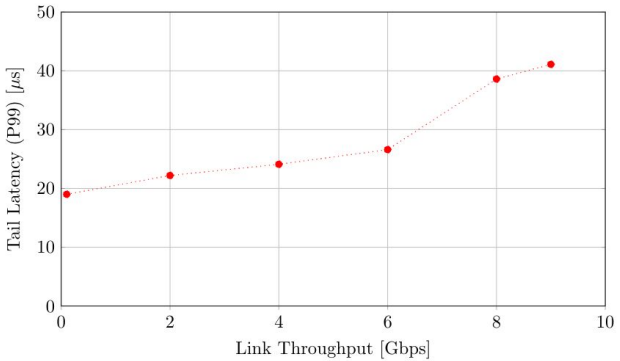
- SLO-aware: minimizing packet losses with a latency objective



Quadrant Design: SLO-aware Auto-scaling

Primary scaling signal: per-chain tail latency estimation

- Key Idea: the tail worker delay + the tail *network transmission delay*

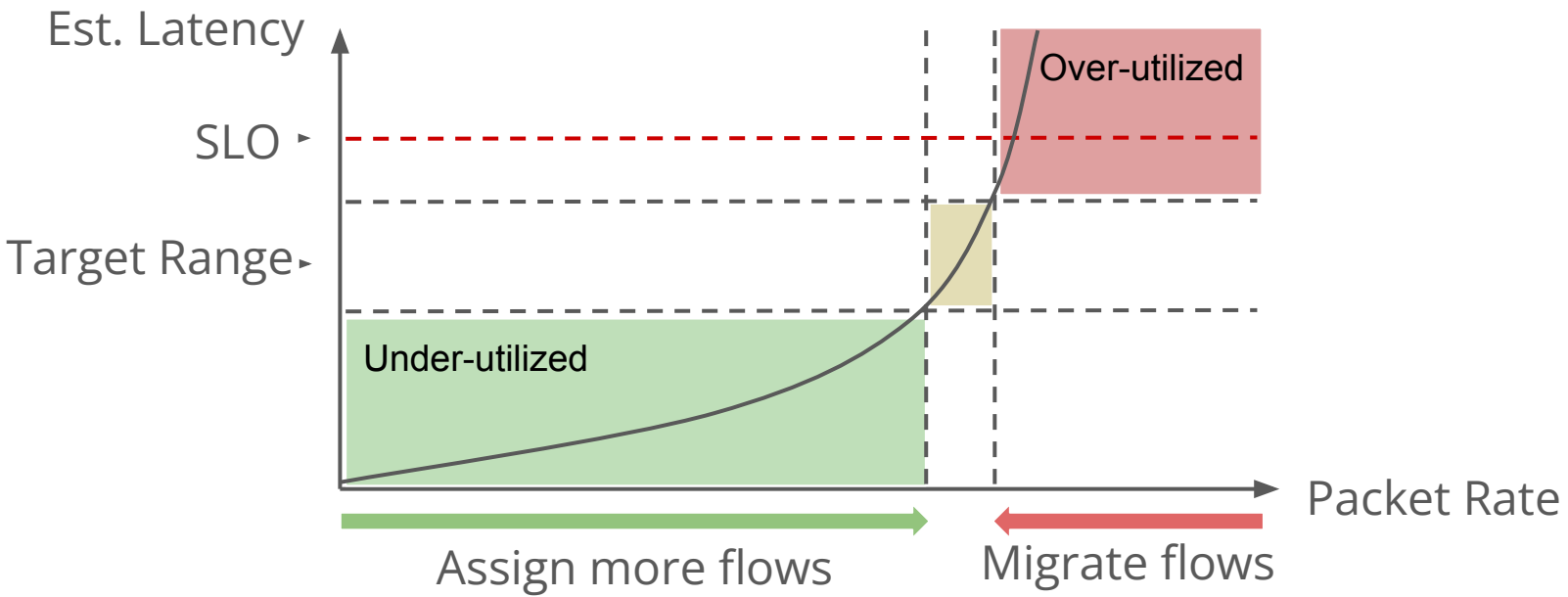


Est. tail worker delay
(based on worst-case per-batch delay)

Est. tail network transmission delay
(worst-case offline-profile)

Quadrant Design: SLO-aware Auto-scaling

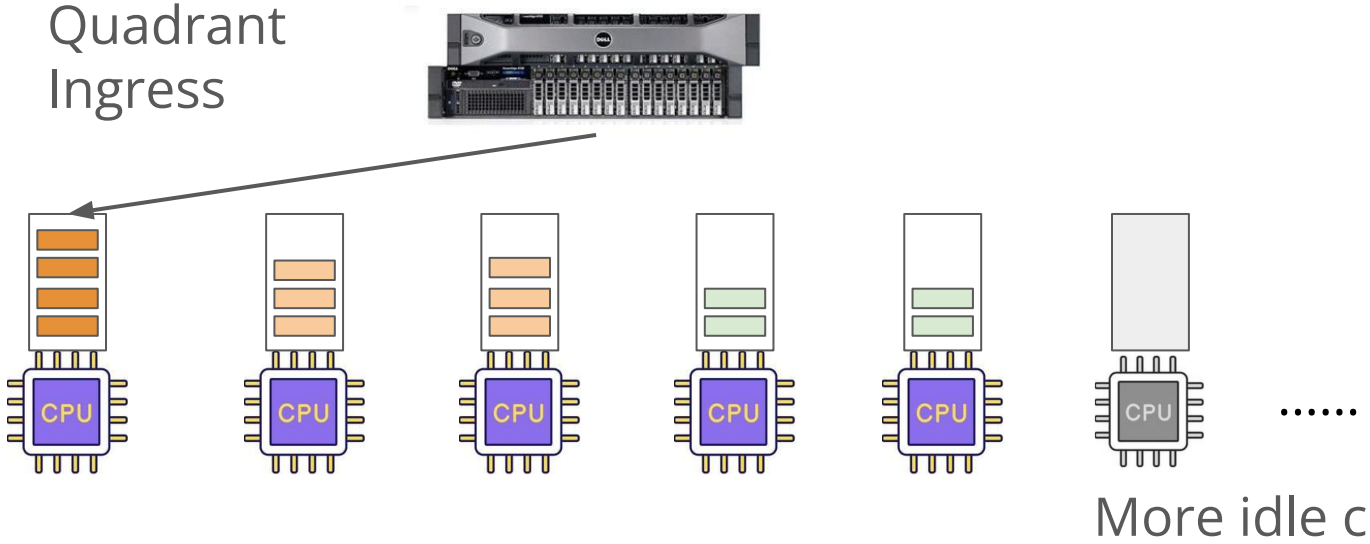
Cluster-scale NF chain auto-scaling: *minimizing # of CPU cores* while preventing latency SLO violations



Quadrant Design: SLO-aware Auto-scaling

Cluster-scale NF chain auto-scaling: *minimizing # of CPU cores*

- Key idea: prioritize the under-utilized core with the highest load



Thank you!

Email: jianfenw@usc.edu

Quadrant: A Cloud-Deployable NF Virtualization Platform

Jianfeng Wang, Tamás Lévai, Zhuojin Li,
Marcos A. M. Vieira, Ramesh Govindan and Barath Raghavan

USC Viterbi School of Engineering

MŰEGYETEM 1782

UFMG UNIVERSIDADE FEDERAL DE MINAS GERAIS