bwNET2020+

Overview

BelWü TechDay 2024

Project Partners

universität ulm

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Hochschule Karlsruhe
University of
Applied Sciences

Research groups and data center operators work closely together.

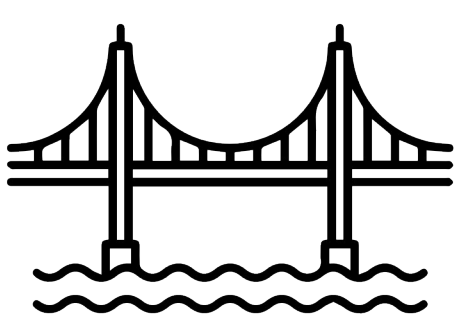
Basic Infos

- Previous project: bwNET100G+
- Funded by MWK Baden-Württemberg
- Original duration 4 years (Jan ’20 — Dec ‘23)
  - Successful midterm-review (June 2022)
  - Project extension: Until March ‘24
- Successful end review (September 2023)
- Follow-up project bwNET2.0

How to integrate research and operation?

Operation

- Reliable and robust operation is key
- Changes only if absolutely necessary
- Minimise risk
- No significant time budget for innovation



Research

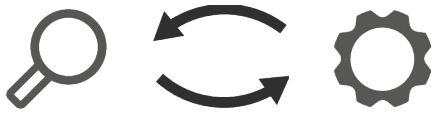
- Innovation focus
- Open for disruptive changes
- Early adaption or invention of new technologies
- Dedicated time budget for new ideas and solutions

Vision of bwNET2020+: **Self-driving Networks**

- **Autonomously adapt** to new situations
  - Little (if at all) human intervention
  - Adapt quickly (in the order of minutes)
  - Follow high-level intent

Picked up, e.g., by Google:
ORION-Paper
ACM USENIX 2021

- **Focus** of bwNET2020+
  - Specific, carefully selected topics
  - Use cases and show cases
  - Interplay of research and operation



BelWue and campus networks should be prepared

Research Topics in bwNET2020+

Policy Control

- Pragmatic approach
  - architecture
  - … applicability
- Use cases

Service Function Chaining

- Explore tunneling technologies
- Explore P4
- Deployment of service functions

Security

- Self-defending network
- Zero trust security
- Fast reaction
- Tailored security architecture

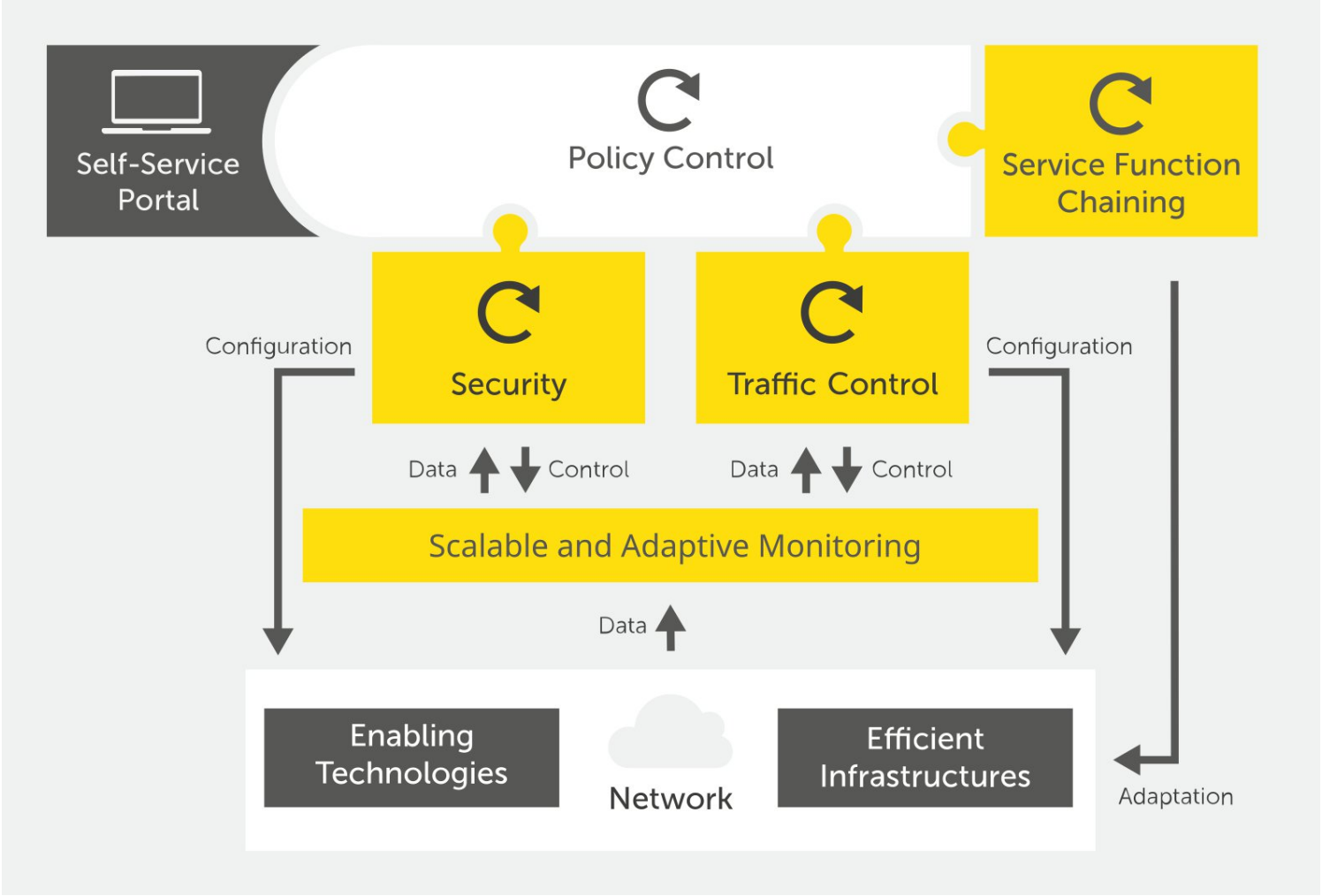
Traffic Control

- (Coordinated) congestion control
- Reinforcement-based mechanisms
- Wireless access: explore multipath capabilities

Scalable and Adaptive Monitoring

- Decentralized monitoring infrastructure
- Scalability and adaptivity
- Suitable interfaces

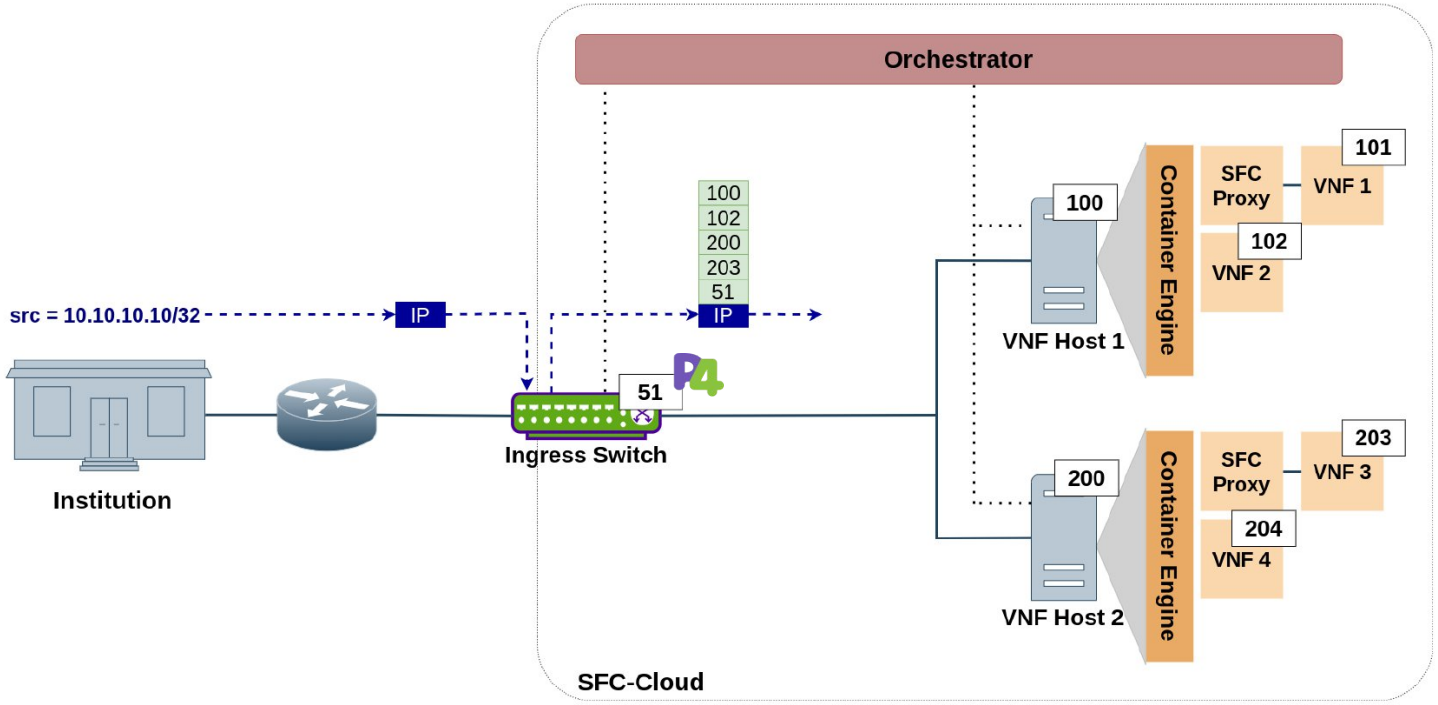
High-level Architecture



Example: SR-MPLS Data Plane

P4-SFC: Exemplary SFC architecture implementation

Task 4.3



Service Function Chaining Based on Segment Routing Using P4 and SR-IOV (P4-SFC)*

Andreas Stockmayer, Stephan Hinselmann, Marco Häberle, and Michael Menth

Chair of Communication Networks, University of Tuebingen, Tuebingen, Germany
{andreas.stockmayer,marco.haeberle,menth}@uni-tuebingen.de,
stephan.hinselmann@student.uni-tuebingen.de

Abstract. In this paper we describe P4-SFC to support service function chaining (SFC) based on a single P4-capable switch and off-the-shelf components. It utilizes MPLS-based segment routing for traffic forwarding in the network and SR-IOV for efficient packet handling on hosts. We describe the P4-SFC architecture and demonstrate its feasibility by a prototype using the Tofino Edgecore Wedge 100BF-32X as P4 switch. Performance test show that L2 throughput for VNFs on a host is significantly larger when connected via SR-IOV with the host's network interface card instead of over a software switch.

1 Introduction

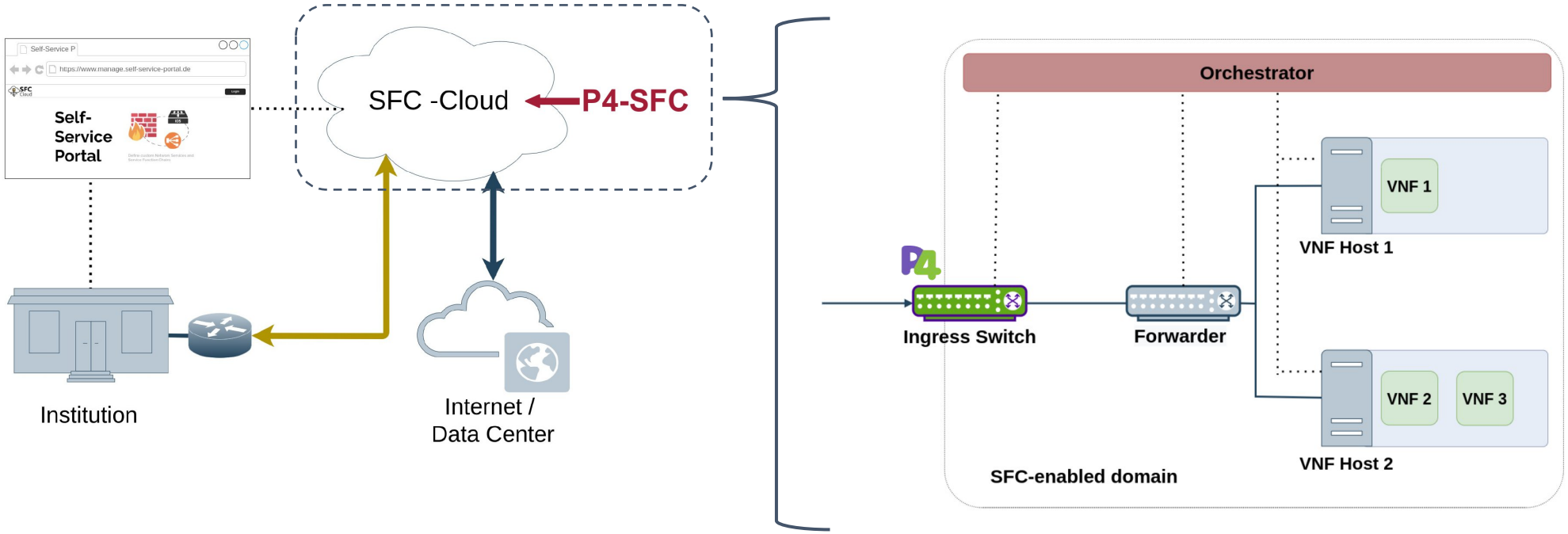
Packet processing at the network ingress or egress typically requires network functions (NFs) such as firewalls, IDS/IPS, NAT, and others. In the past, these functions have been provided as hardware appliances. To reduce costs, many of them are today implemented as applications running on standard server hardware as so-called virtual NFs (VNFs). Complex services may be composed of multiple VNFs. This is called service function chaining (SFC). As the VNFs of a SFC are generally located on different hosts, SFC requires forwarding support in the network. That means, a packet which is classified for a specific SFC must be forwarded along a path that visits all VNFs of the respective SFC in a predefined order.

The IETF has proposed various approaches for this problem. One approach requires per-SFC state in the network, the other requires the ability of a node – we call it the SFC ingress node – to encode a source route in the packet header. Segment routing based on MPLS is one preferred option for source route encoding. It requires the SFC ingress node to push a stack of MPLS labels, but intermediate SFC forwarders just need to pop individual labels. While the latter is simple, pushing a large header stack is hardly supported by today's affordable hardware.

* This work was supported by the bwNET2020+ project which is funded by the Ministry of Science, Research and the Arts Baden-Württemberg (MWK). The authors

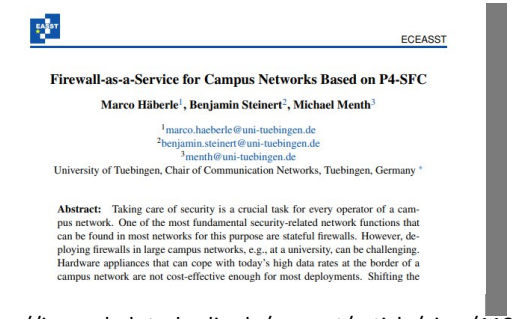
https://link.springer.com/chapter/10.1007/978-3-030-59851-8_19

Firewall as a Service for Campus Networks Based on P4-SFC



- Web-based management of Service Functions and Chains
- Automated deployment in SFC Cloud
- 100G ready

Use Case 1


ECEASST

Firewall-as-a-Service for Campus Networks Based on P4-SFC

Marco Häberle[1], Benjamin Steinert[2], Michael Menth[3]

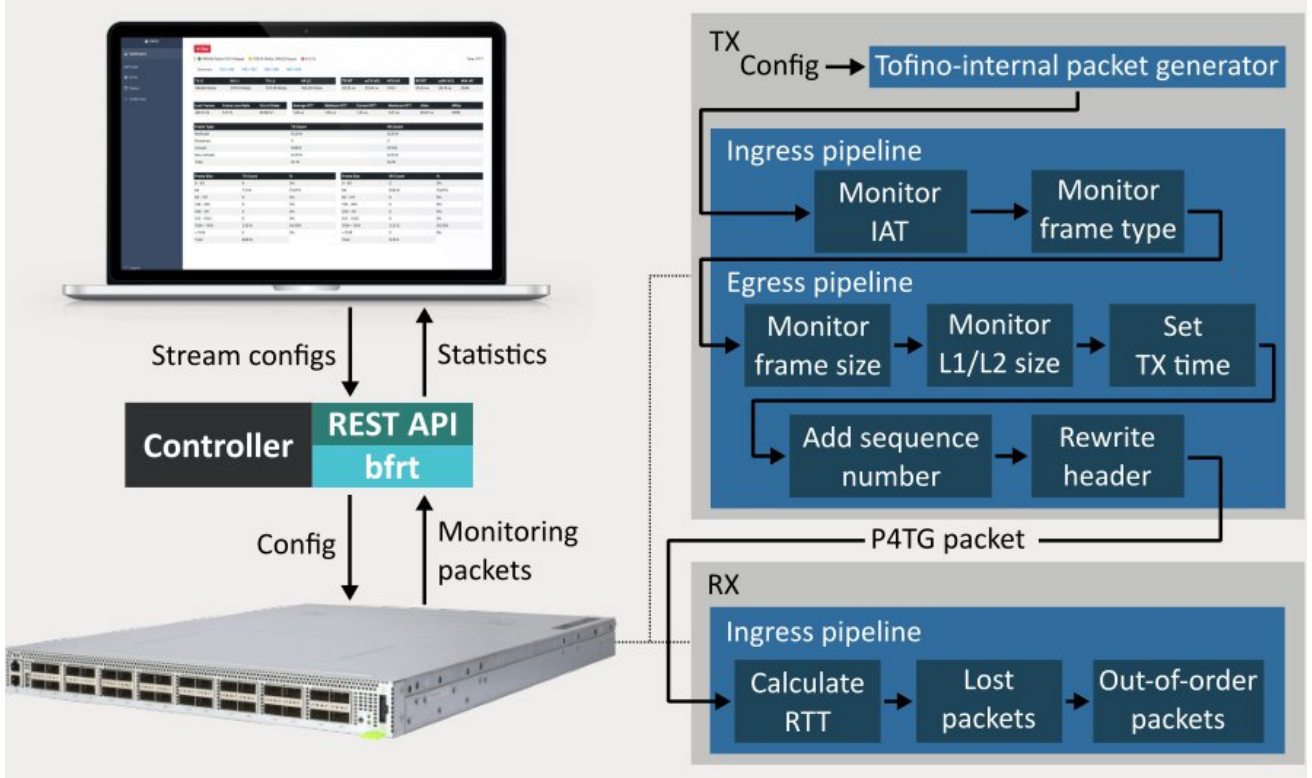
[1]marco.haeberle@uni-tuebingen.de
[2]benjamin.steinert@uni-tuebingen.de
[3]menth@uni-tuebingen.de
University of Tuebingen, Chair of Communication Networks, Tuebingen, Germany *

Abstract: Taking care of security is a crucial task for every operator of a campus network. One of the most fundamental security-related network functions that can be found in most networks for this purpose are stateful firewalls. However, deploying firewalls in large campus networks, e.g., at a university, can be challenging. Hardware appliances that can cope with today's high data rates at the border of a campus network are not cost-effective enough for most deployments. Shifting the

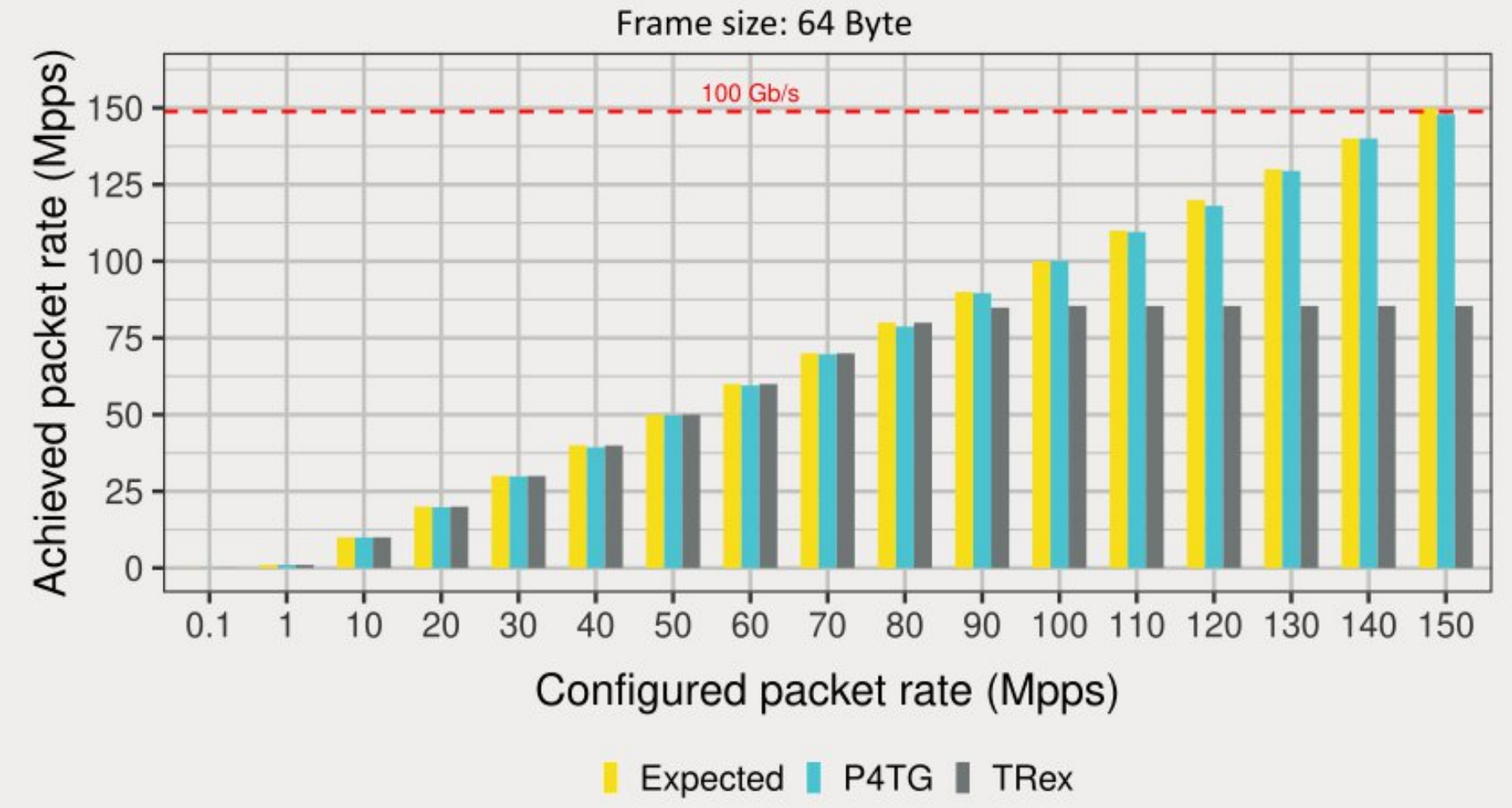
https://journal.ub.tu-berlin.de/eceasst/article/view/1185

P4 Traffic Generator

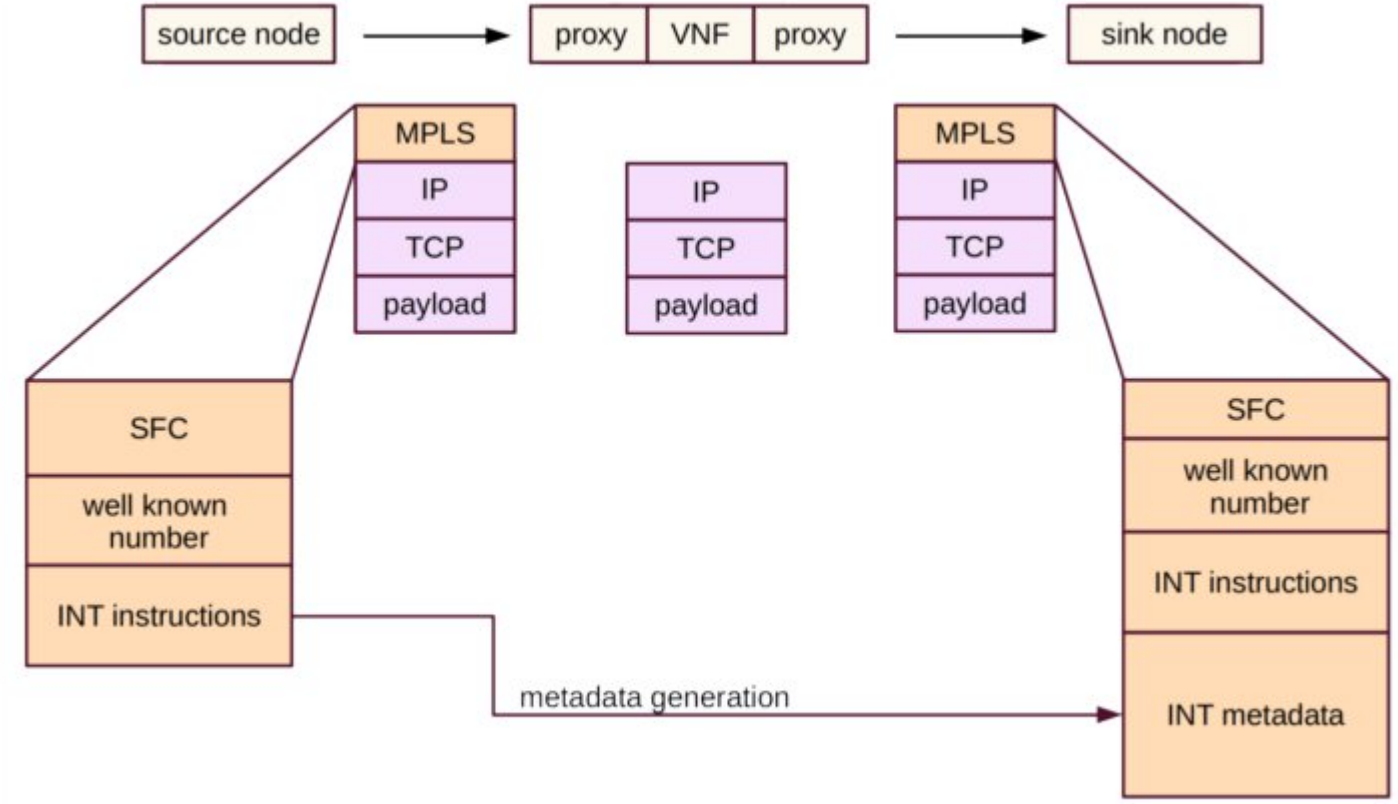
- Motivation
  - 100+ Gbit/s difficult to generate with software
  - Need hardware acceleration
  - Hardware based TGs very expensive
- Features
  - Traffic generation on single port (up to 100 Gbit/s)
  - Constant Bit-Rate and random (poisson) traffic



P4 Traffic Generator



In-band Network Telematry (INT) Proxy



CPU-Scheduler and (TCP-)Performance

- Solution: Enable „amd-pstate“ kernel module

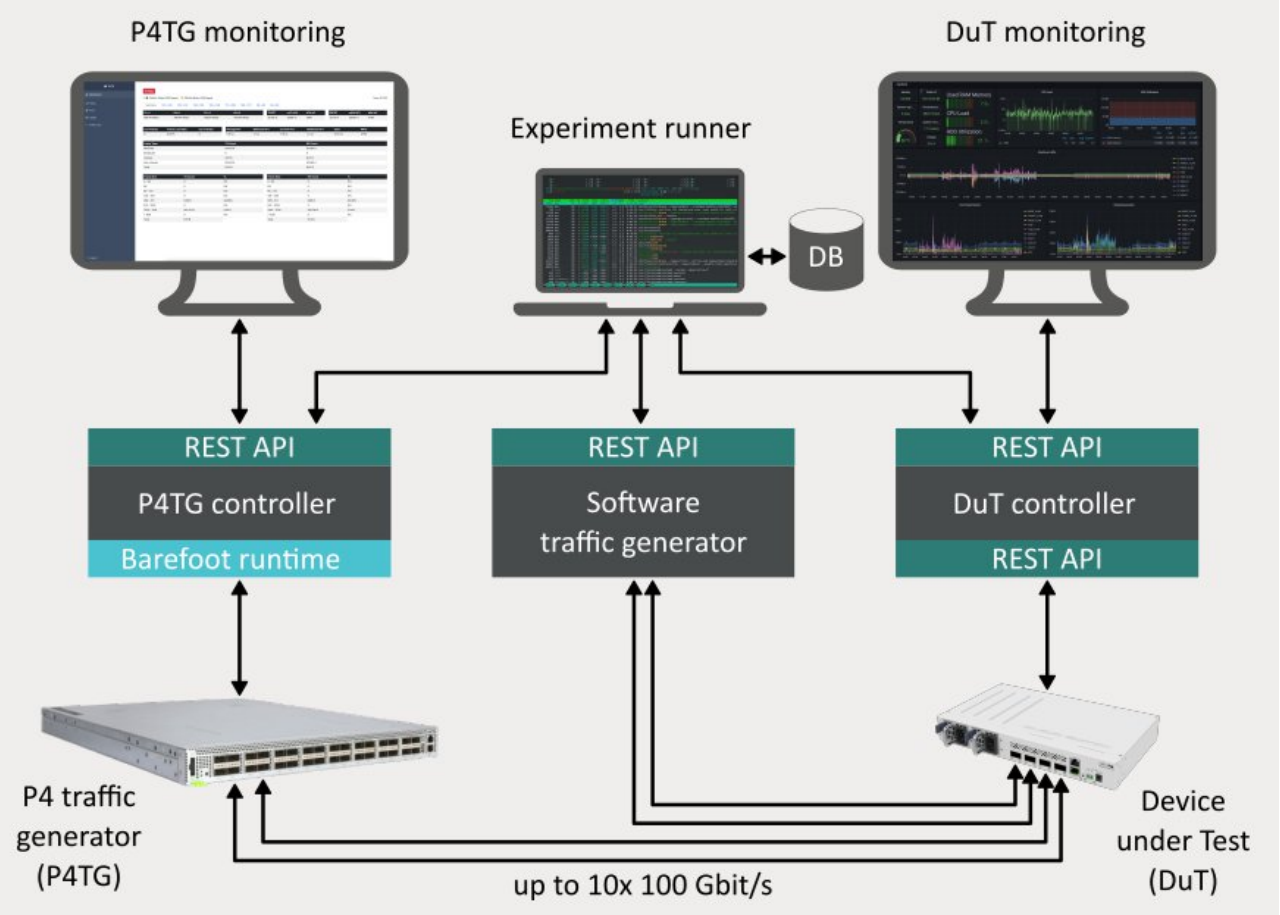
```
root@hannibal:~# cat /etc/kernel/cmdline
root=ZFS=rpool/ROOT/pve-1 boot=zfs amd_pstate=active
```

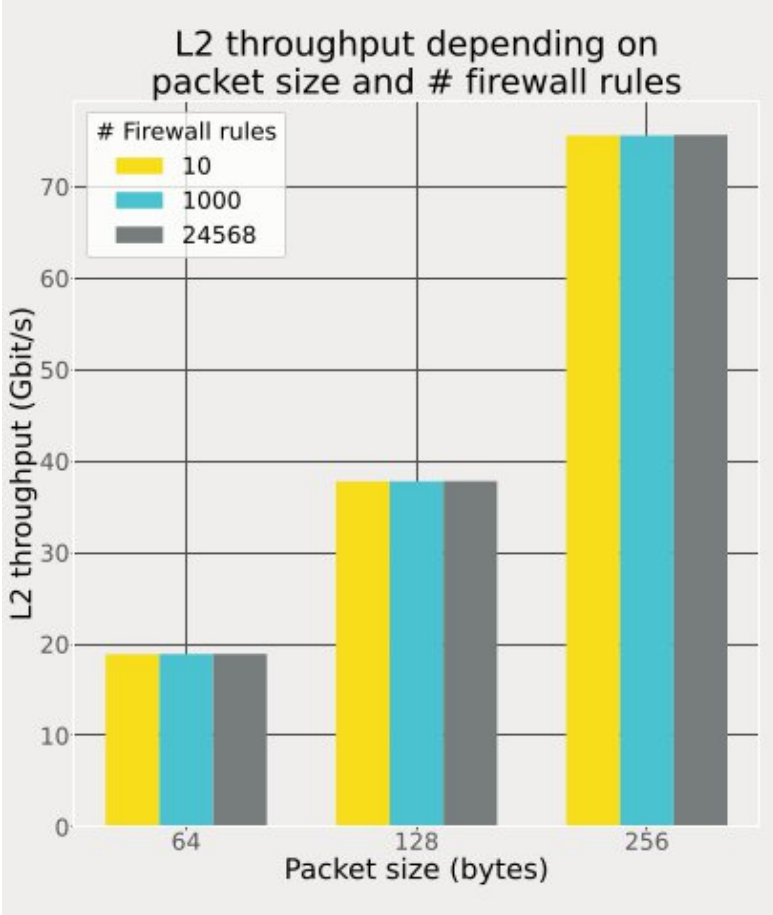
- TCP with iperf3

Server \ OS	Ubuntu 20.04	Ubuntu 22.04
Intel	60 Gbit/s	30 Gbit/s
AMD (before fix)	35 Gbit/s	20 Gbit/s
AMD (after fix)	65 Gbit/s	75 Gbit/s

- → Minor issue, maximum impact

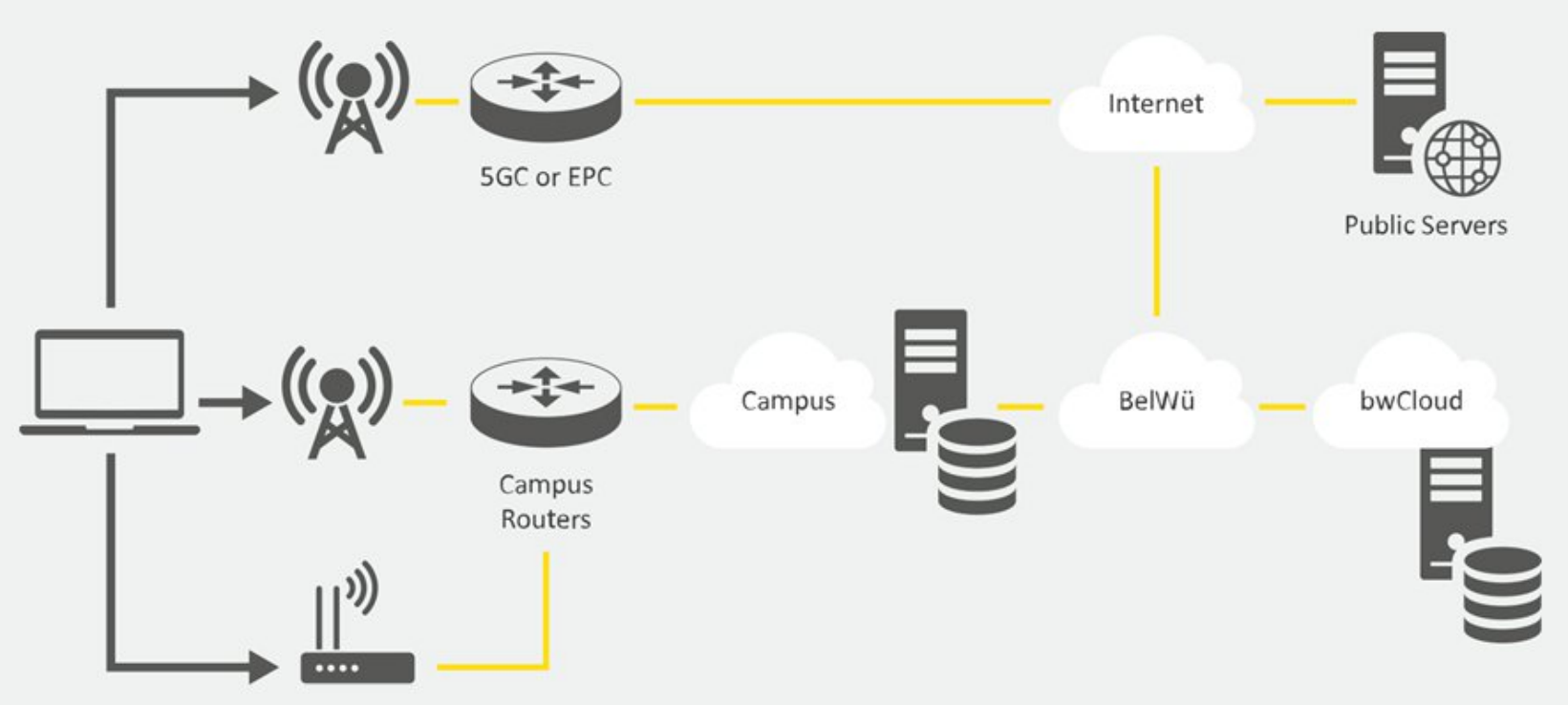
Automated Testbench





Use Case Scenario: Digital Classroom

- Mobile devices with Wi-Fi 5/6 and 4G/5G connection
  - 20 - 40 students accessing digital learning materials



5GC: 5G Core Network EPC: Evolved Packet Core (LTE)

Coordinated Congestion Control (C³)

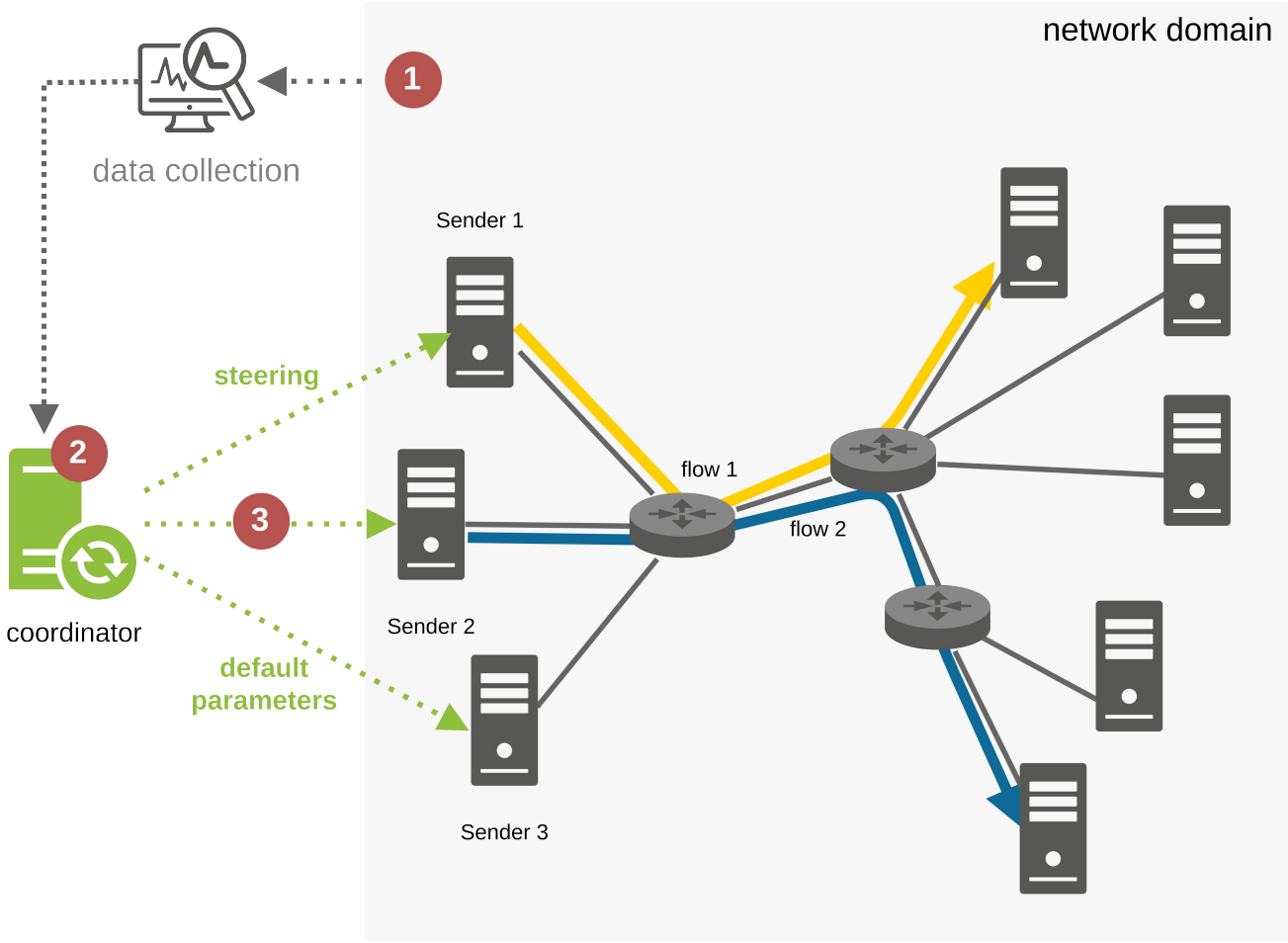
"Congestion Control with SDN-like global view and actions"

With

- Per-flow steering
- Good default parameters

Goal

- Accelerated start-up behavior
- Faster convergence



Coordinated Congestion Control (C^3)

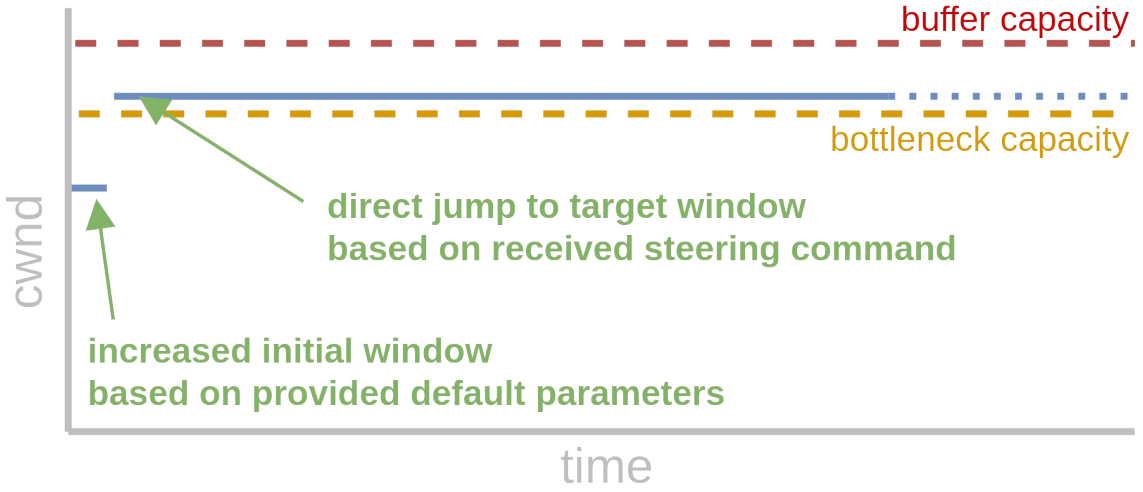
"Congestion Control with
SDN-like global view and actions"

With

- Per-flow steering
- Good default parameters

Goal

- Accelerated start-up behavior
- Faster convergence



Coordinated Congestion Control (C^3)

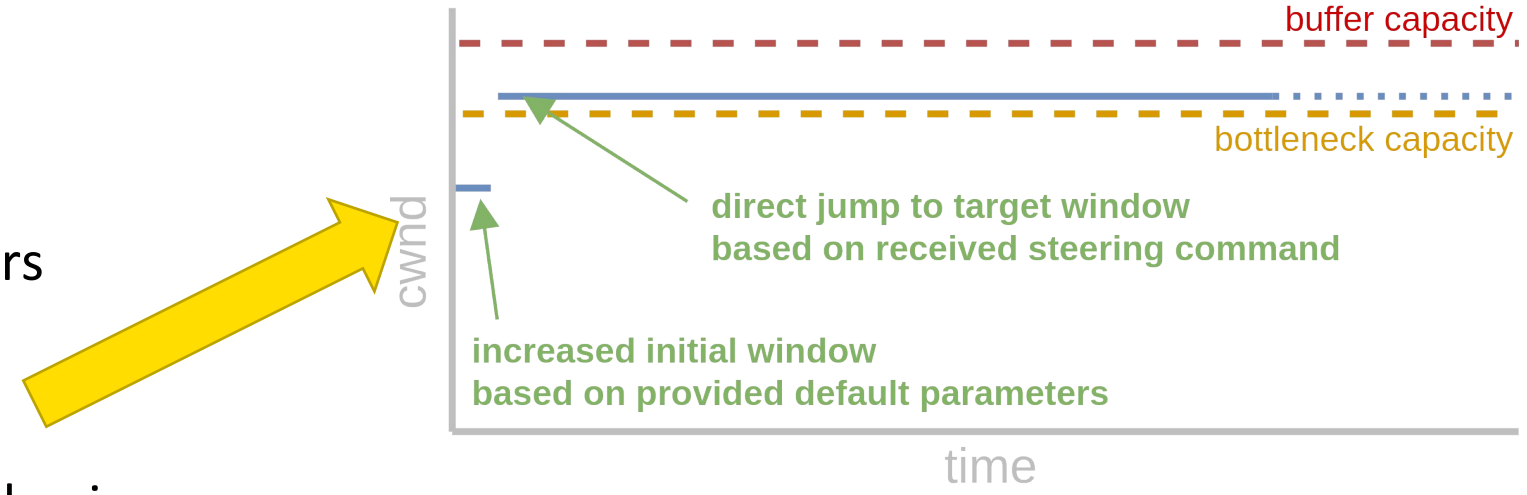
"Congestion Control with
SDN-like global view and actions"

With

- Per-flow steering
- Good default parameters

Goal

- Accelerated start-up behavior
- Faster convergence



Coordinated Congestion Control (C^3)

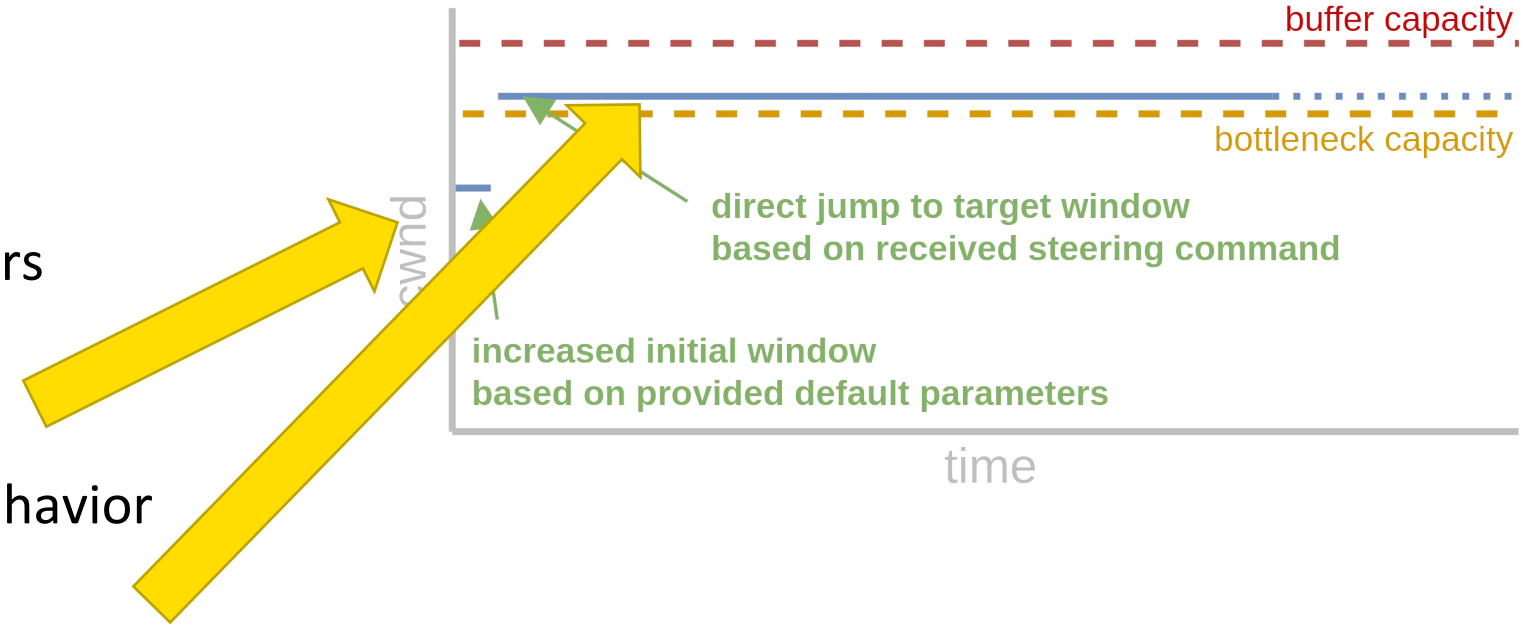
"Congestion Control with
SDN-like global view and actions"

With

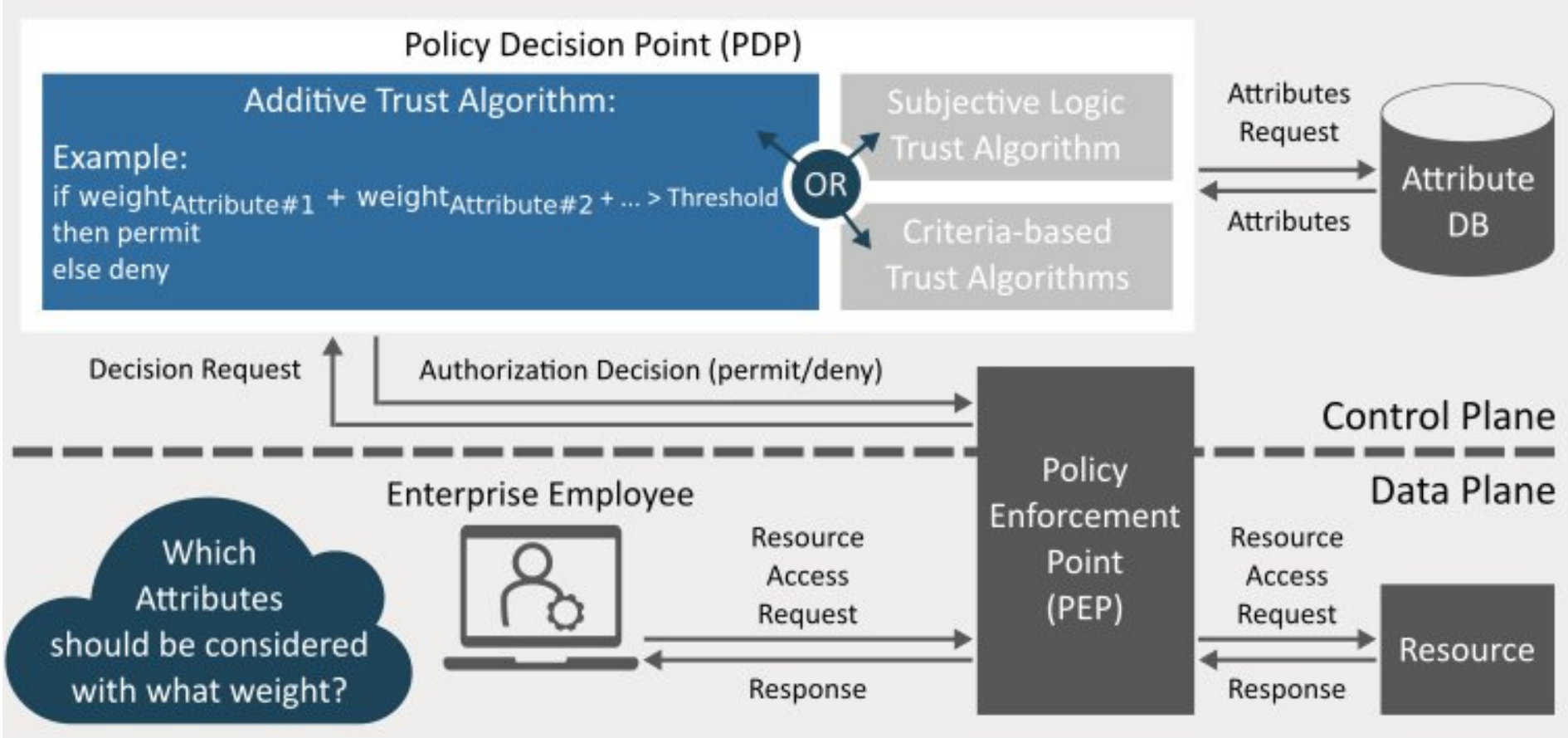
- Per-flow steering
- Good default parameters

Goal

- Accelerated start-up behavior
- Faster convergence

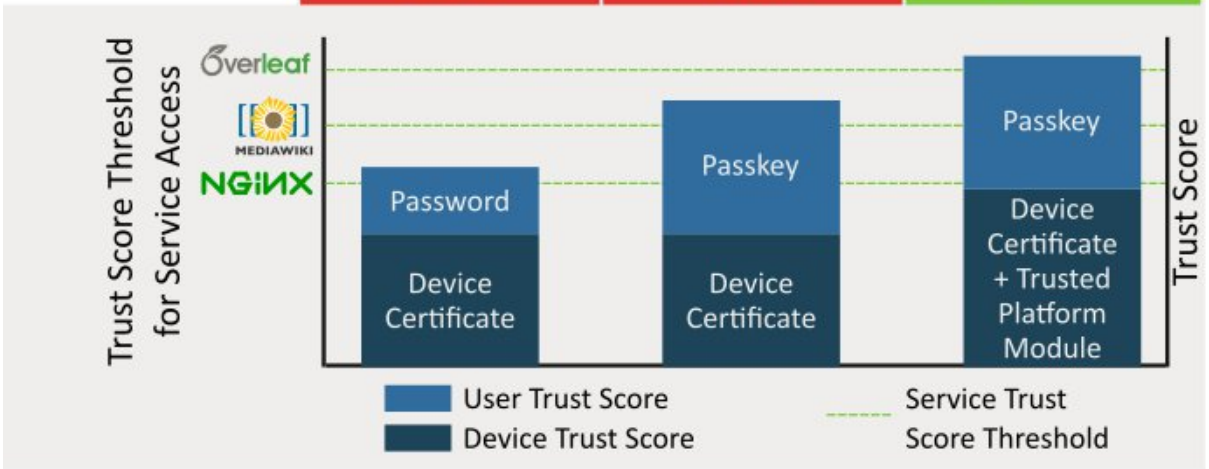


Zero Trust Service Function Chaining

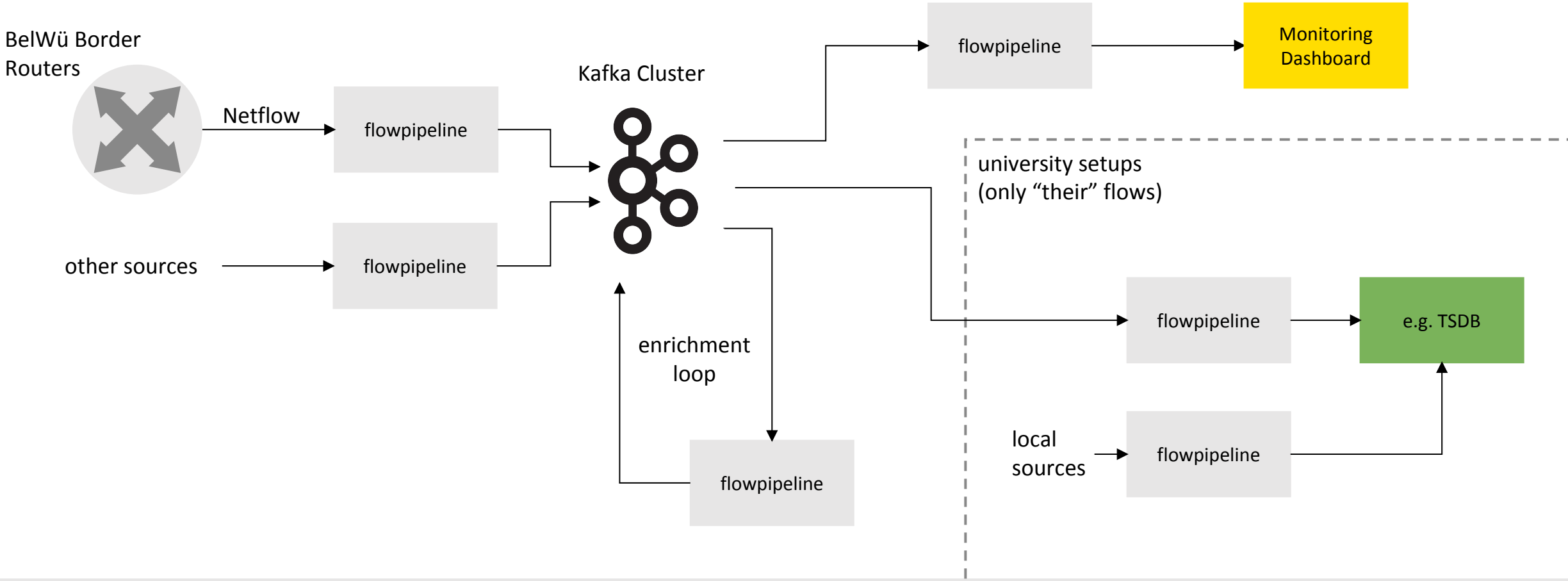


Zero Trust Service Function Chaining

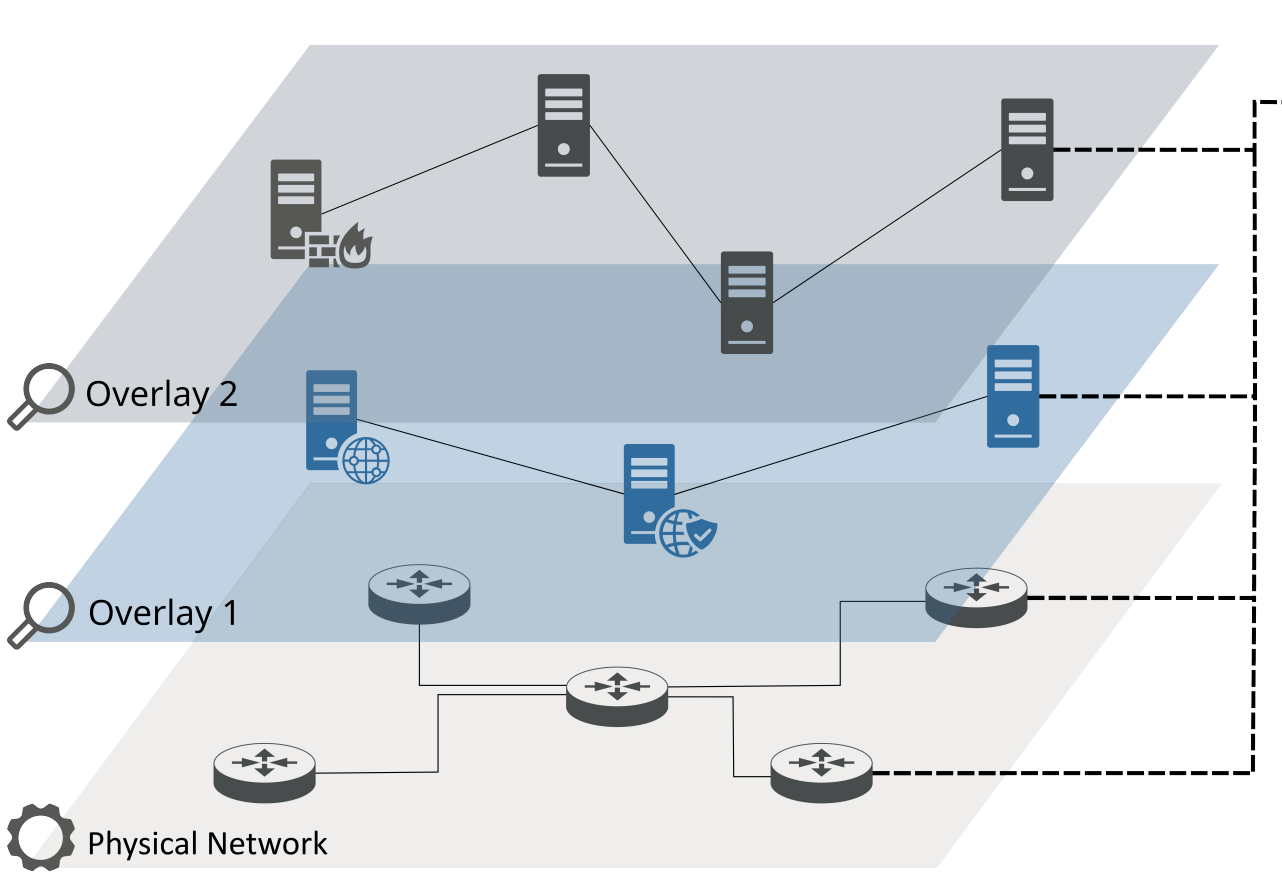
Username:	Alice	Alice	Alice
Password:	*********		
Passkey:		5AC43D9E	5AC43D9E
Device Certificate:	Alice Laptop	Alice Laptop	Alice Laptop
Trusted Platform Module:			TPM
NGINX	Access granted	Access granted	Access granted
MEDIAWIKI	Access denied	Access granted	Access granted
Overleaf	Access denied	Access denied	Access granted

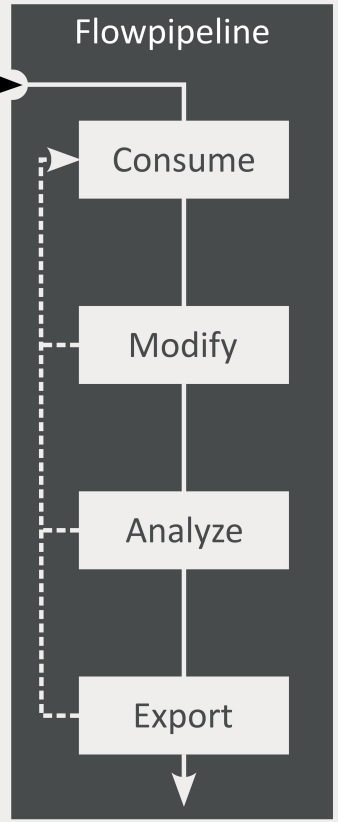


Flowpipeline - Flow Data Processing Platform



Flow Monitoring - Concept



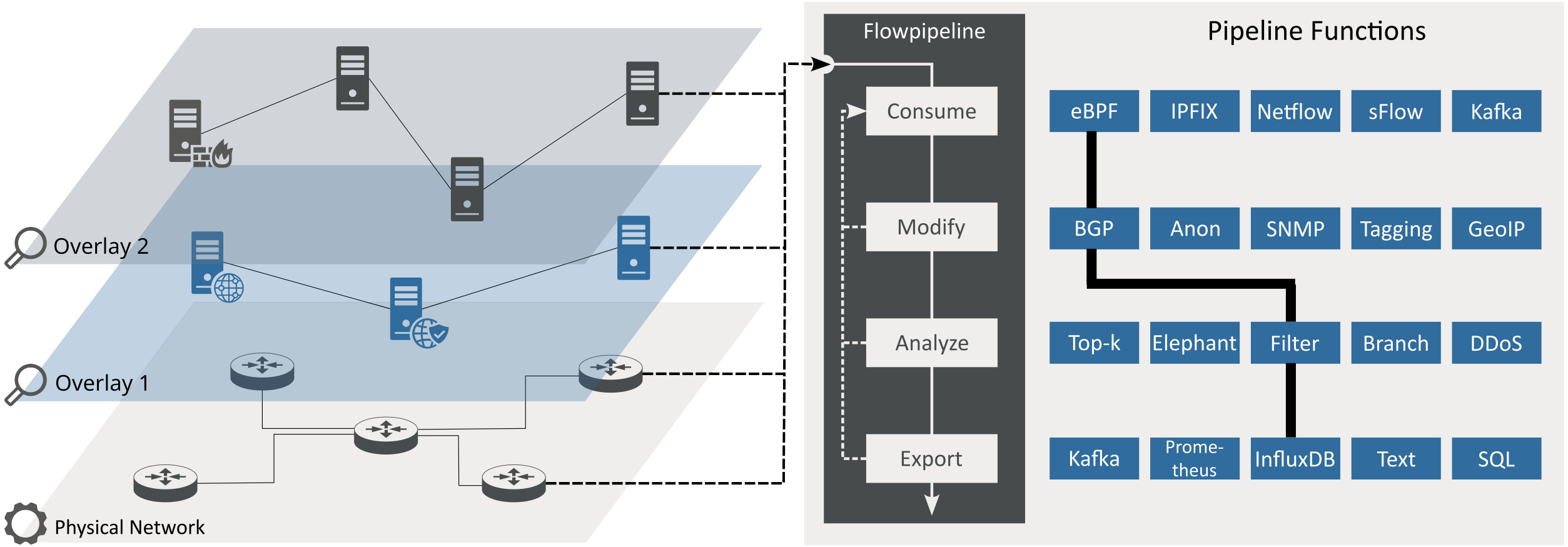


Pipeline Functions

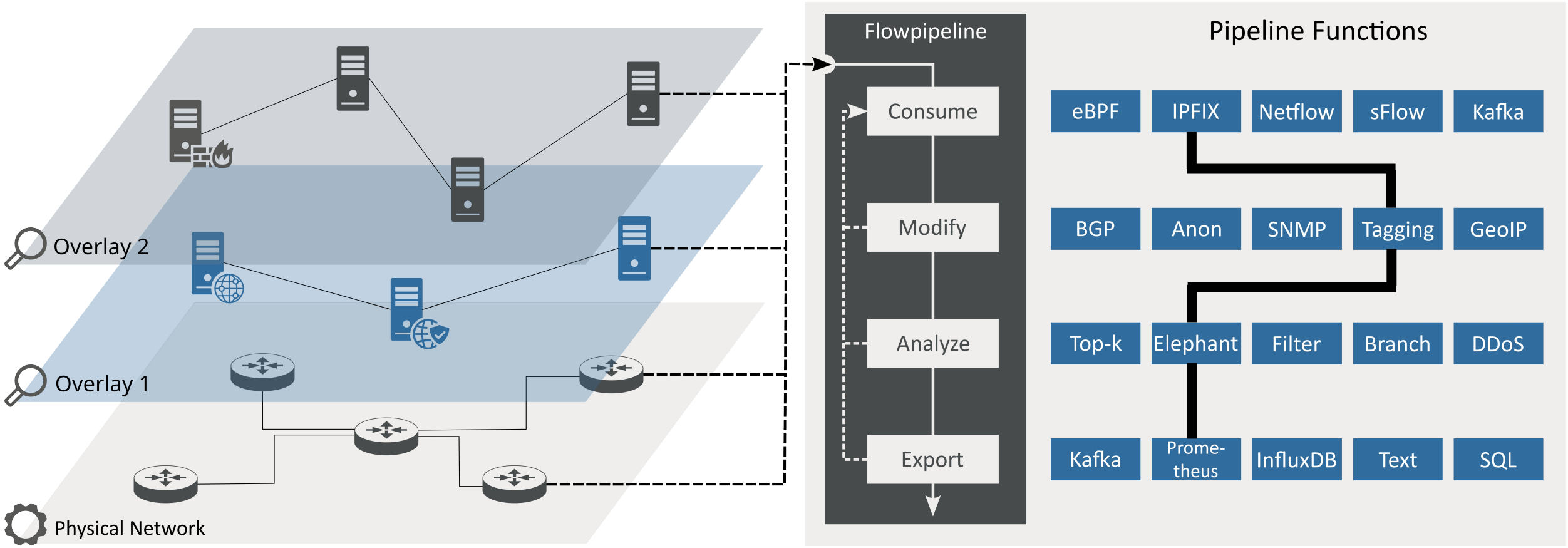
eBPF	IPFIX	Netflow	sFlow	Kafka
BGP	Anon	SNMP	Tagging	GeoIP
Top-k	Elephant	Filter	Branch	DDoS
Kafka	Prome-theus	InfluxDB	Text	SQL

→ Task 2.2, 2.3, 2.4, 2.5, 2.6

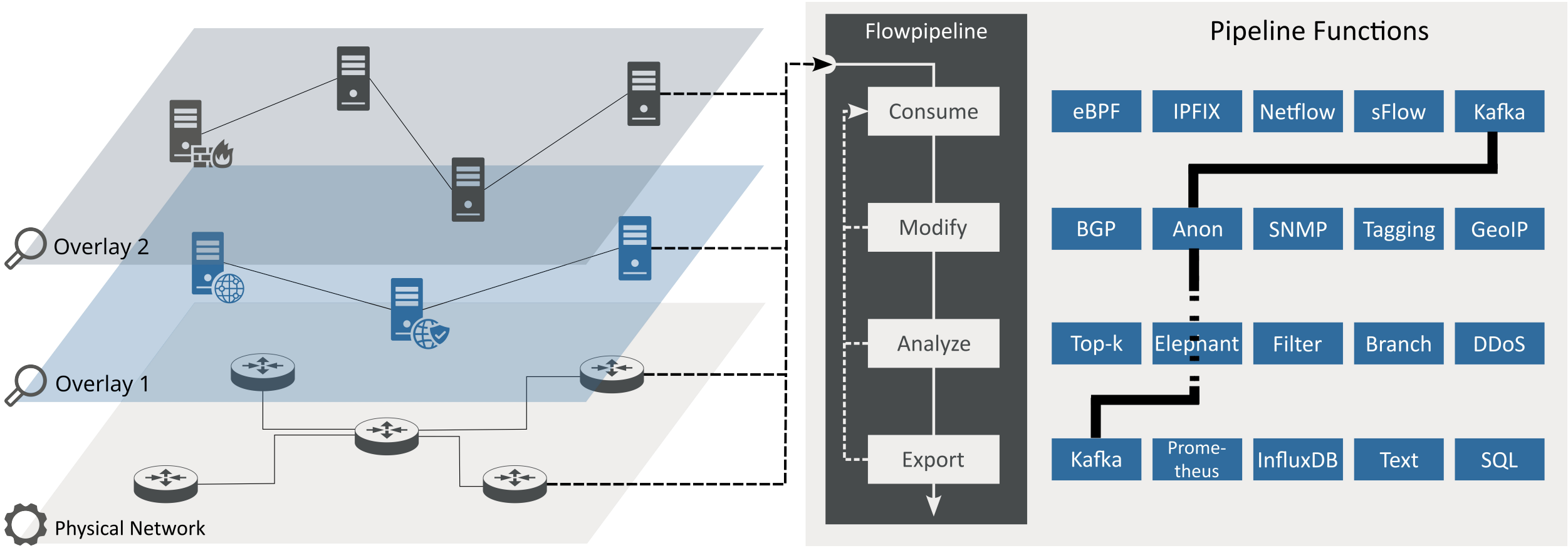
Flow Monitoring - Concept



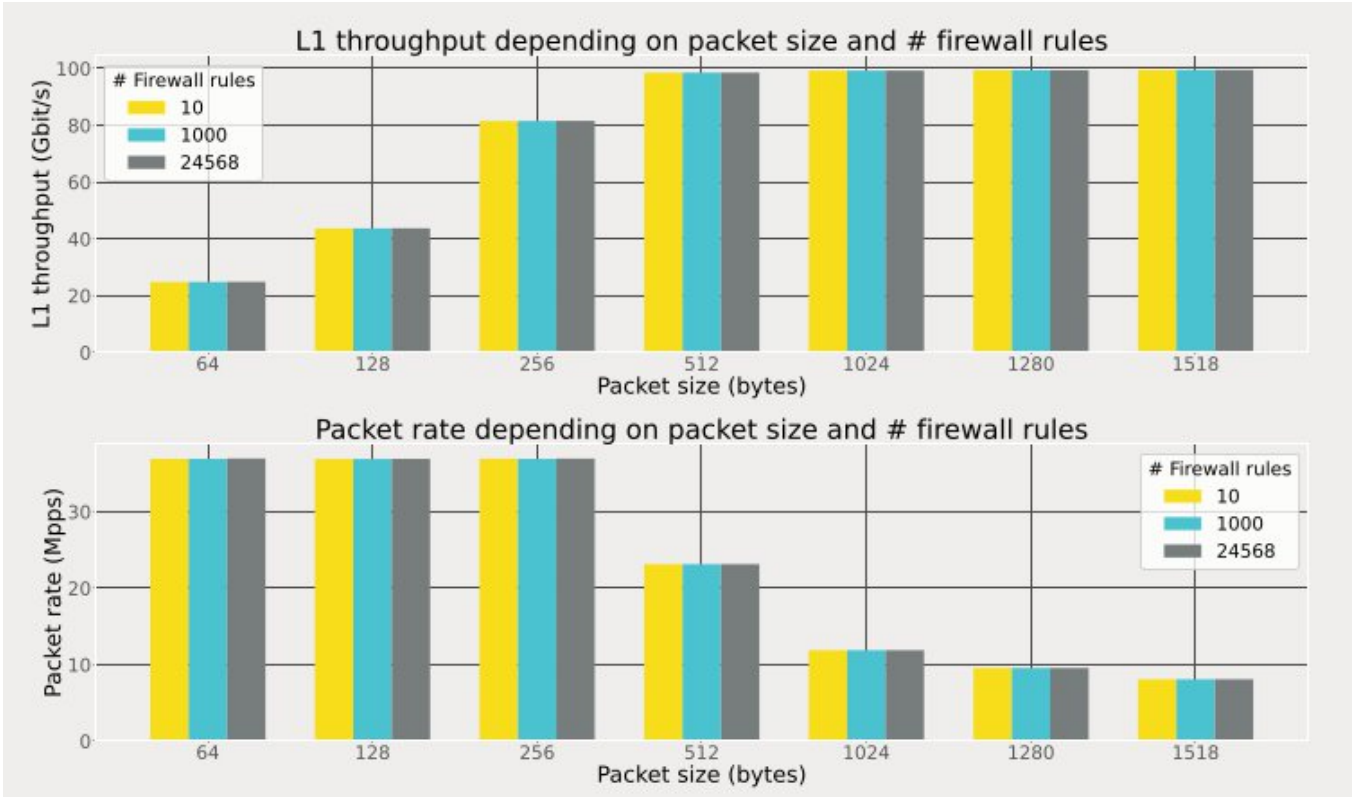
Flow Monitoring - Concept

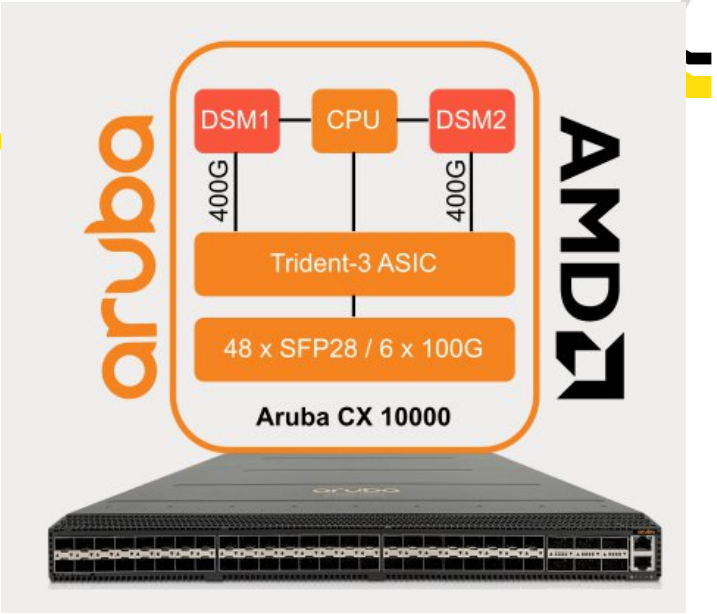


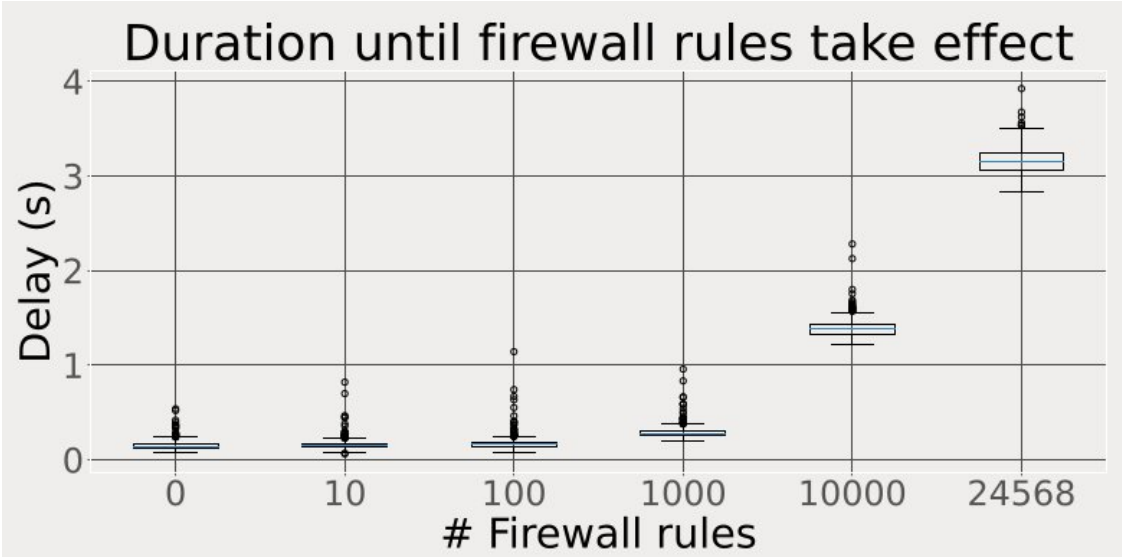
Flow Monitoring - Concept



Hardware Evaluation







bwNET2020+

22 PUBLISHED PAPERS

2 DISER-TATIONS

10+ TECHNICAL REPORTS

15+ DISSEM-INATIONS

23 BA/MA THESES