Universität Rostock Traditio et Innovatio

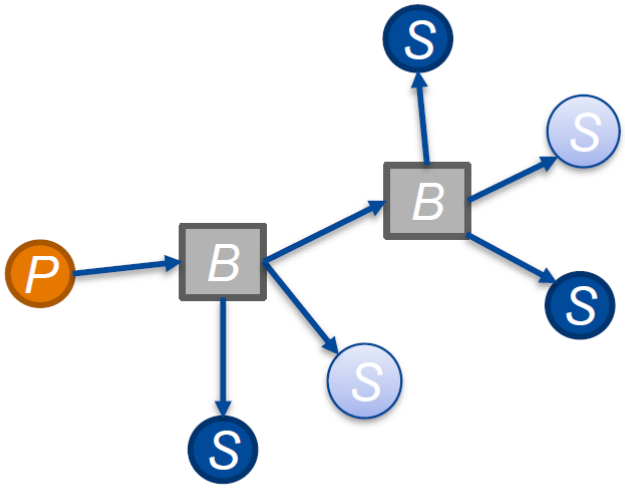
3rd KuVS Fachgespräch "Network Softwarization"

Fast Publish/Subscribe Using Linux eBPF

Michael Tatarski
Gero Mühl
Helge Parzyjegla
Peter Danielis

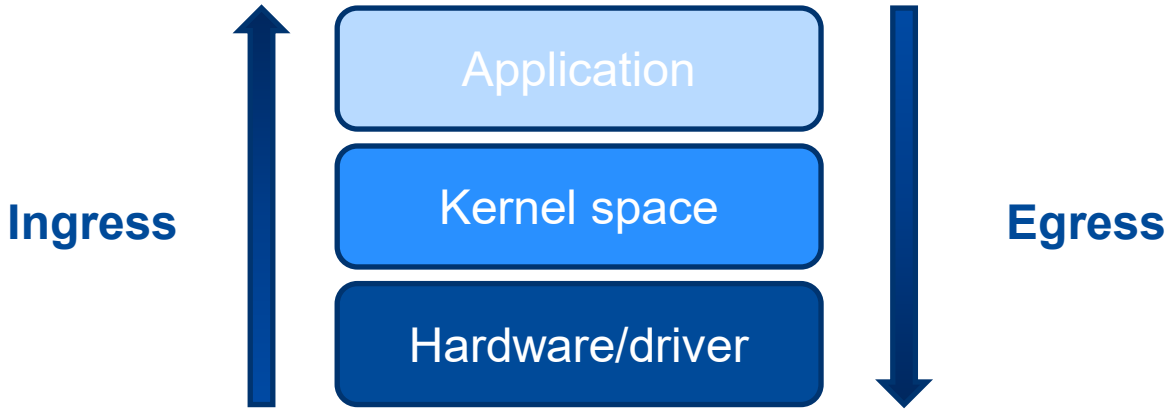
Architektur von Anwendungssystemen (AVA)
Fakultät für Informatik und Elektrotechnik (IEF)
Universität Rostock

Publish/Subscribe Systems

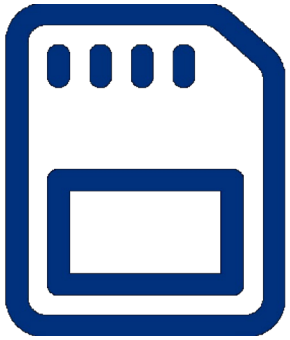


✓ Flexible

✗ Slow

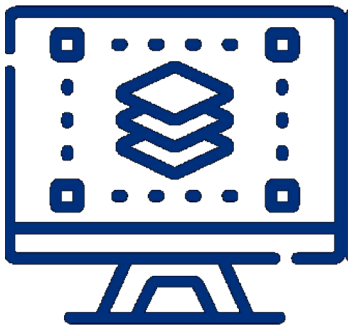


Problem Areas of the Linux Network Stack



Data copy and
memory allocation

Context switches

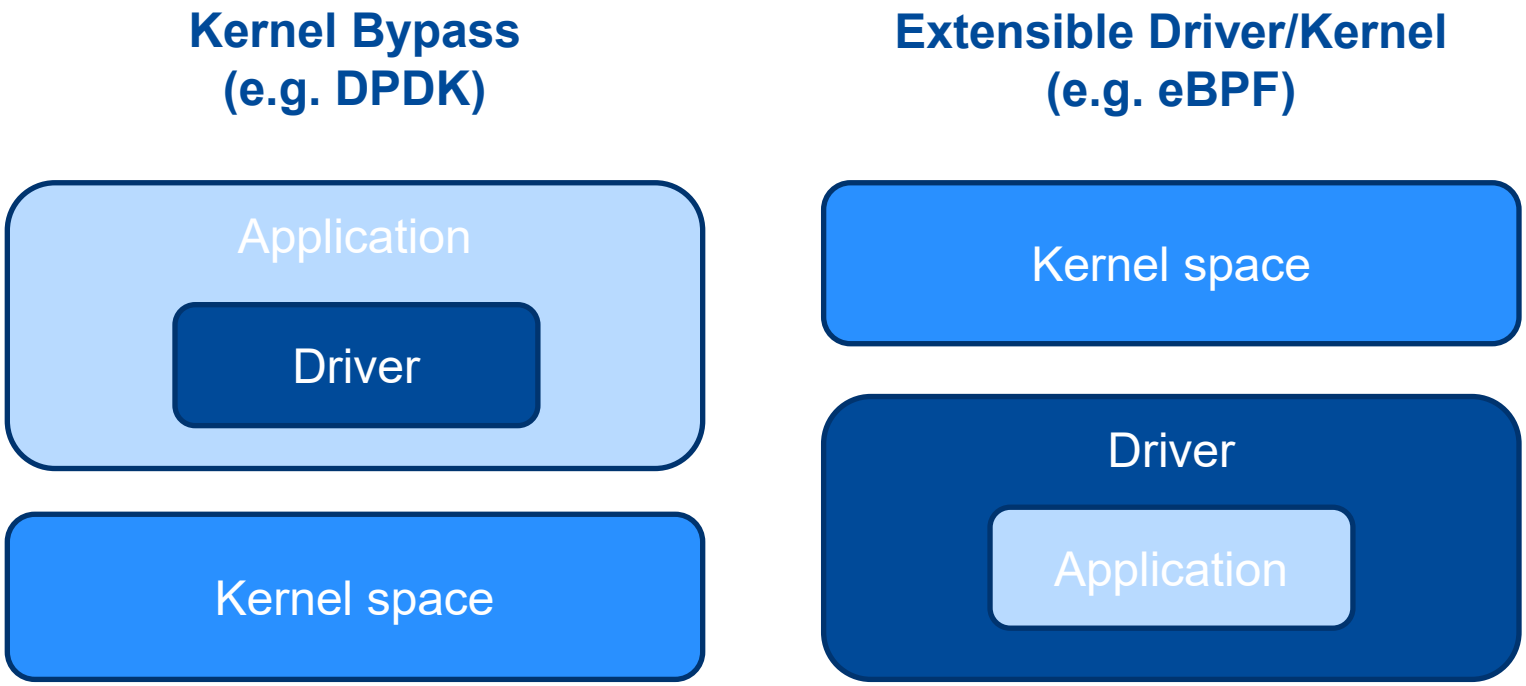


Numerous software
and network layers

Agenda

- Motivation
  - Optimizing the data plane of pub/sub brokers
- Development and design
  - Limitations and challenges
  - System architecture
  - Design of filter techniques
- Evaluation
  - Data rate
  - Latency

Optimizing the data plane of pub/sub brokers



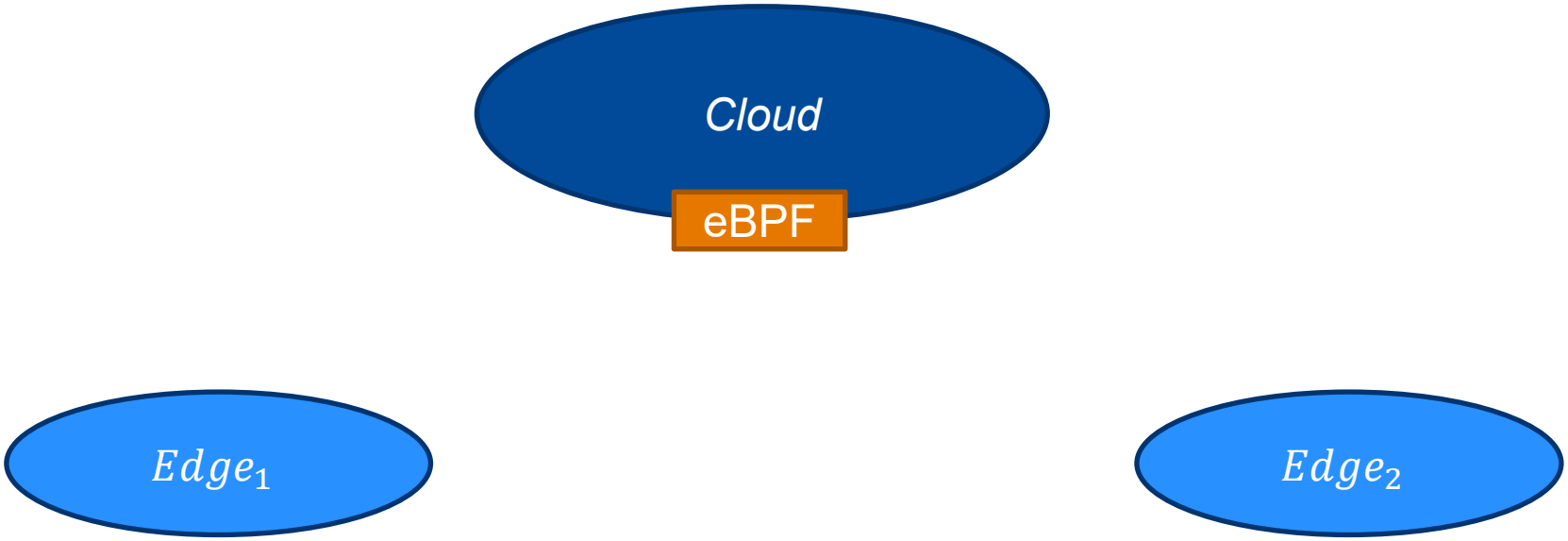
Limitations when using eBPF

- Loops and backward jumps: Usage of dynamic loops is not allowed → Fixed size of routing tables

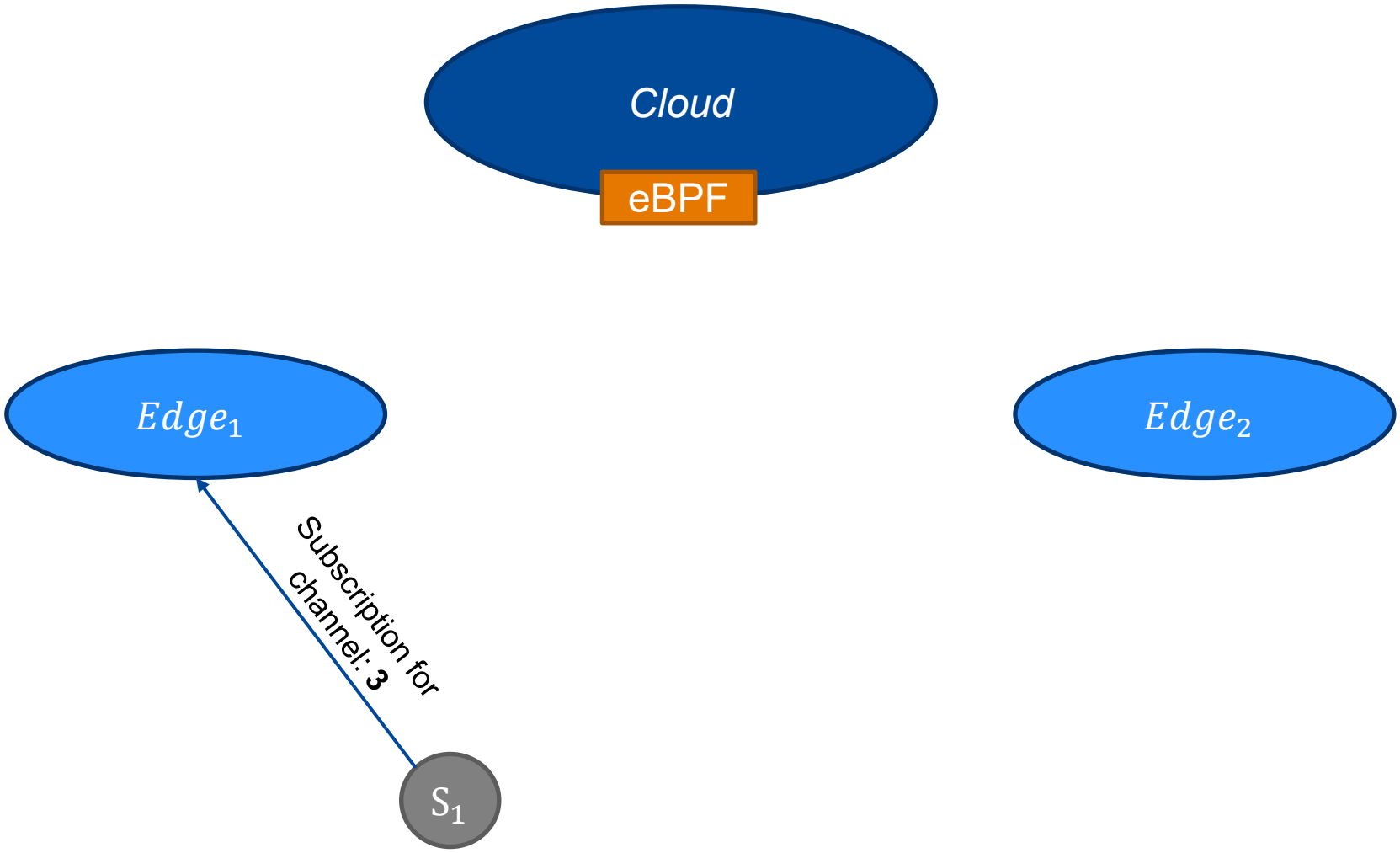
- Events: Only the reception and sending of data packets trigger the execution of eBPF code → Program must dispense with any other form of events such as timeouts

- Delay of network packets: eBPF provides no functionality to delay the delivery of a network packet → pub/sub clients cannot be fully decoupled

System architecture

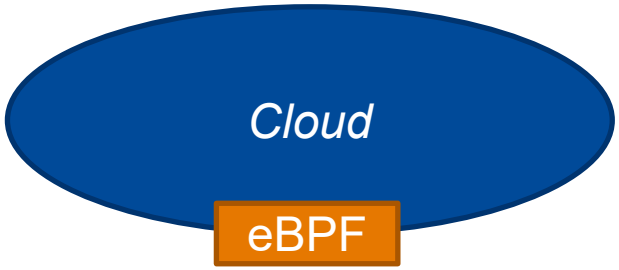


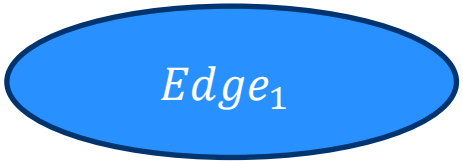
System architecture



System architecture

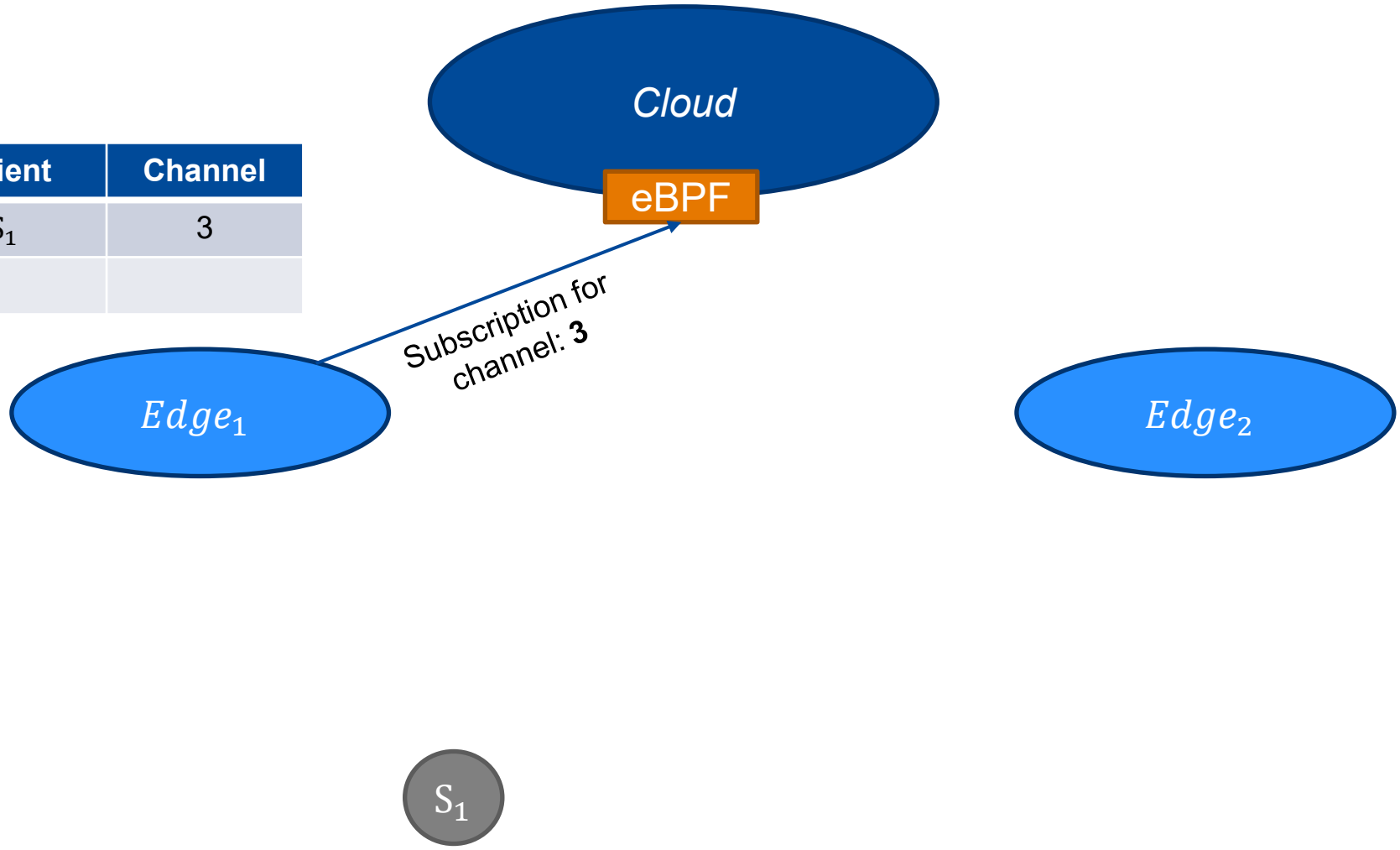
Client	Channel
S_1	3





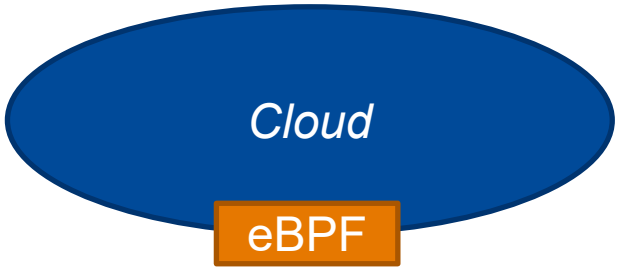
System architecture

Client	Channel
S_1	3



System architecture

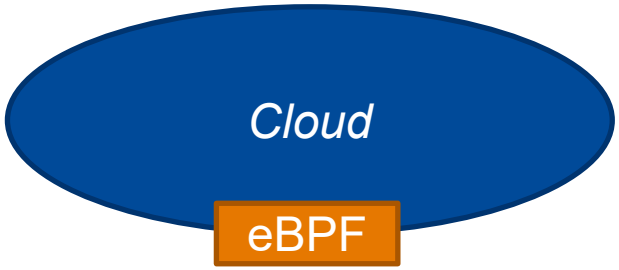
Client	Channel
$Edge_1$	3



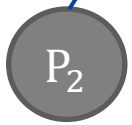
Client	Channel
S_1	3

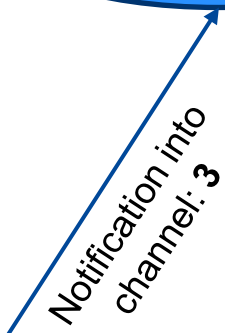
System architecture

Client	Channel
$Edge_1$	3



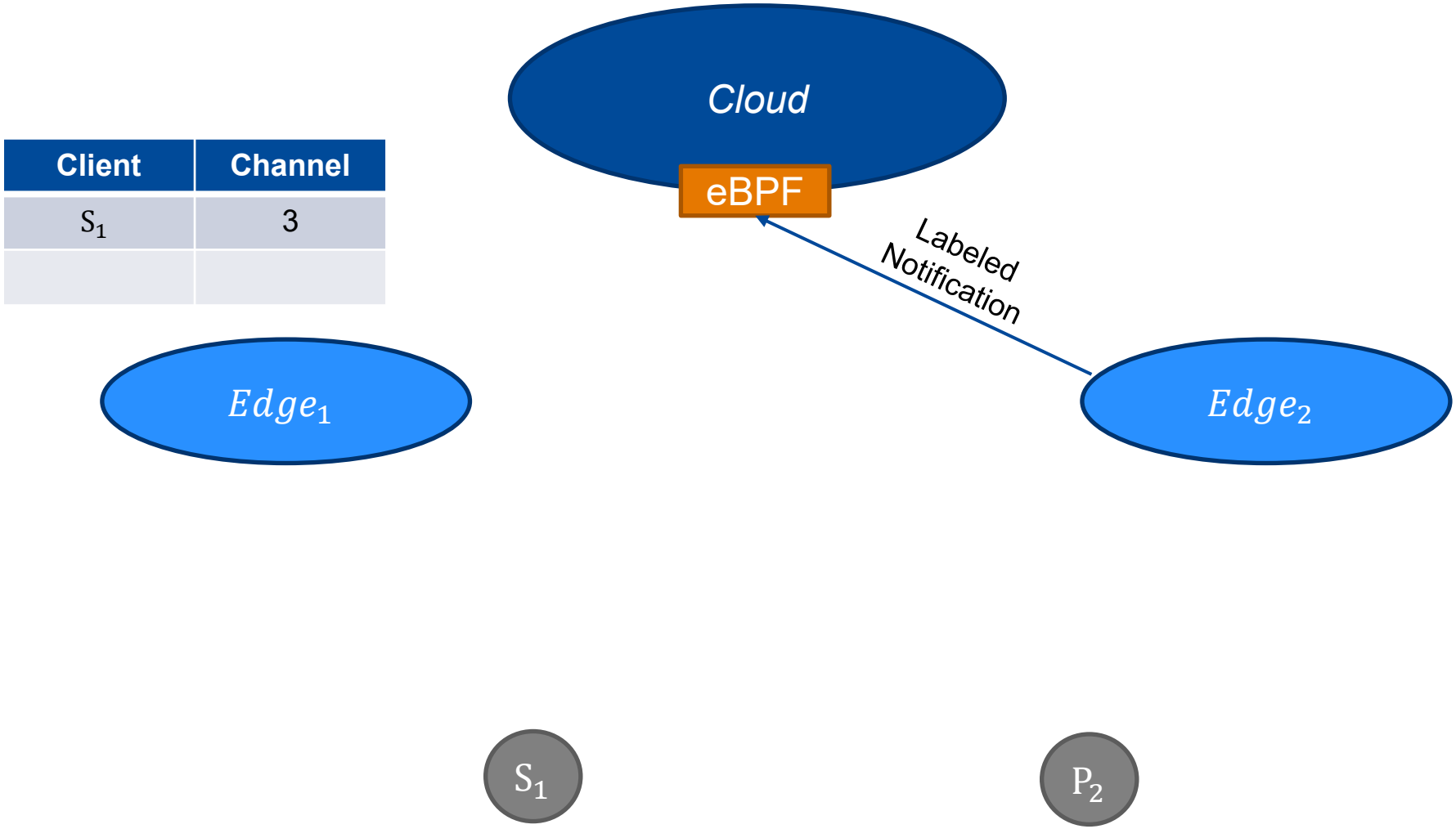
Client	Channel
S_1	3





System architecture

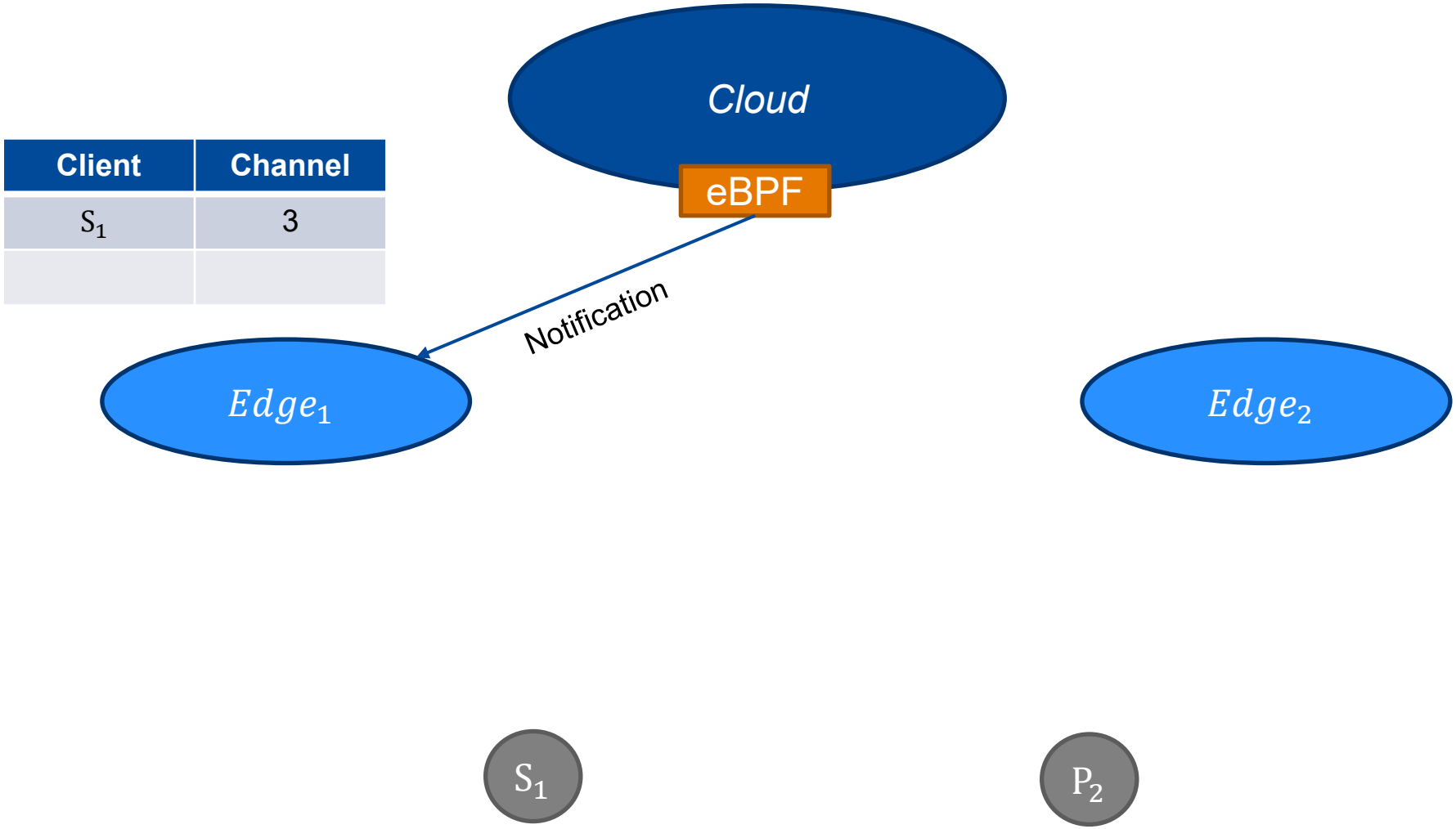
Client	Channel
$Edge_1$	3



Client	Channel
S_1	3

System architecture

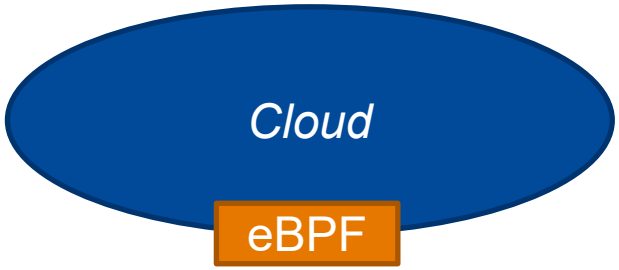
Client	Channel
$Edge_1$	3



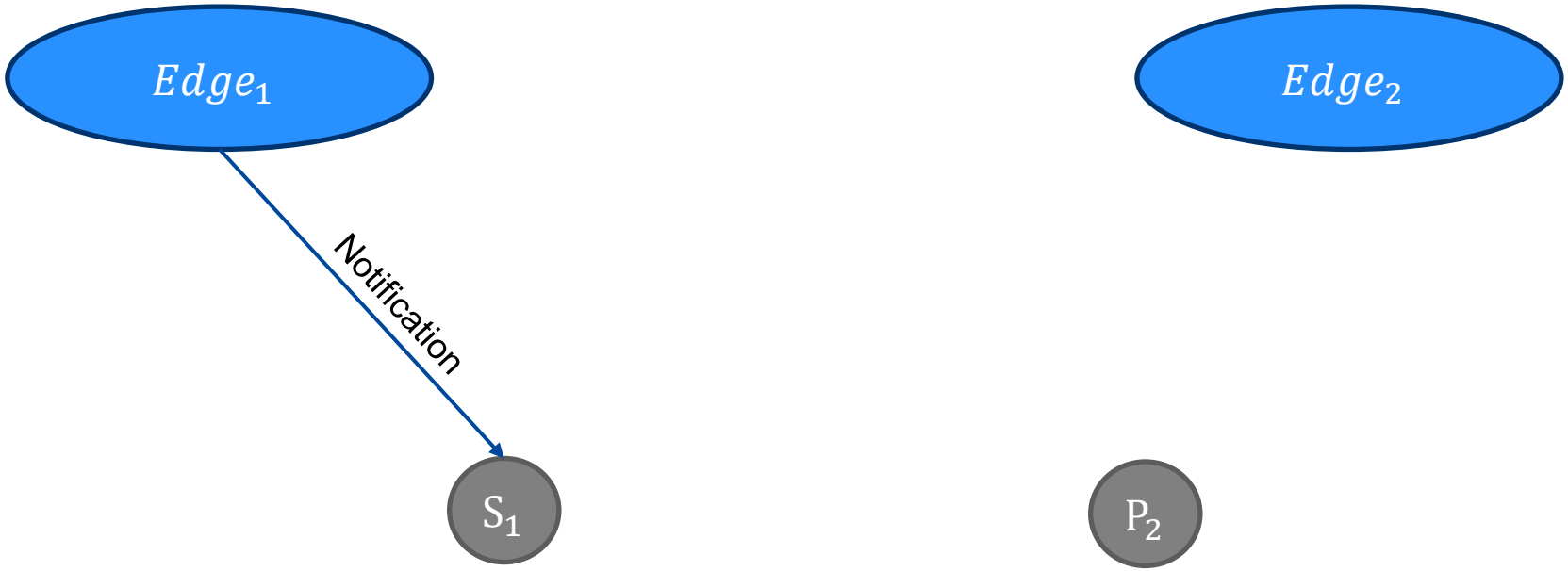
Client	Channel
S_1	3

System architecture

Client	Channel
$Edge_1$	3

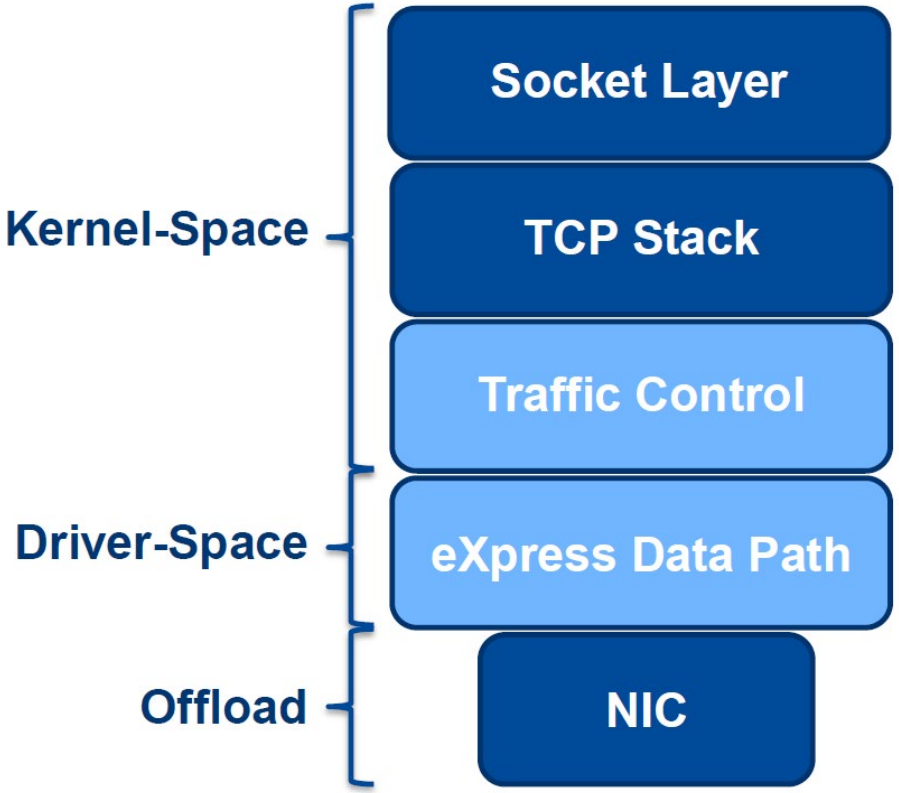


Client	Channel
S_1	3



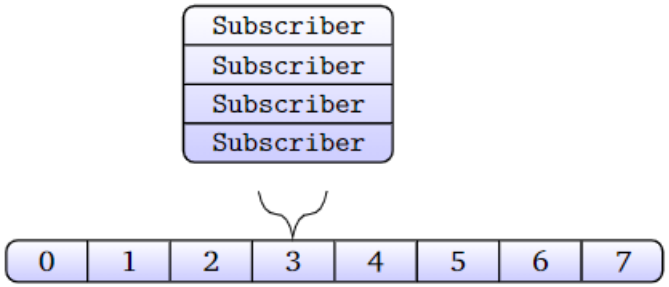
Selection of network hooks

- Traffic Control:
  - Calculation of checksums
  - Altering IP header
- eXpress Data Path:
  - Preprocessing
  - Checking if interested subscribers exist

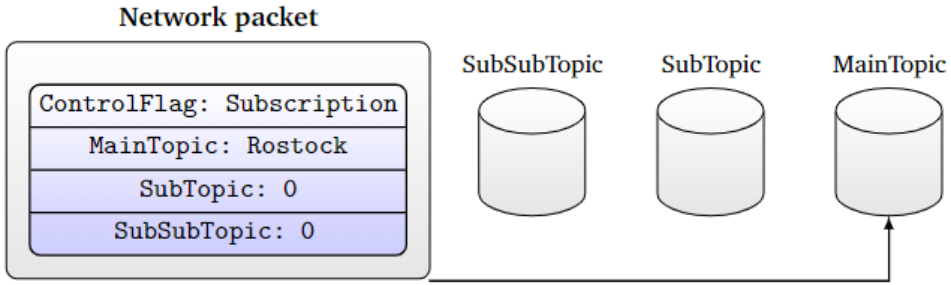


Design of filter techniques

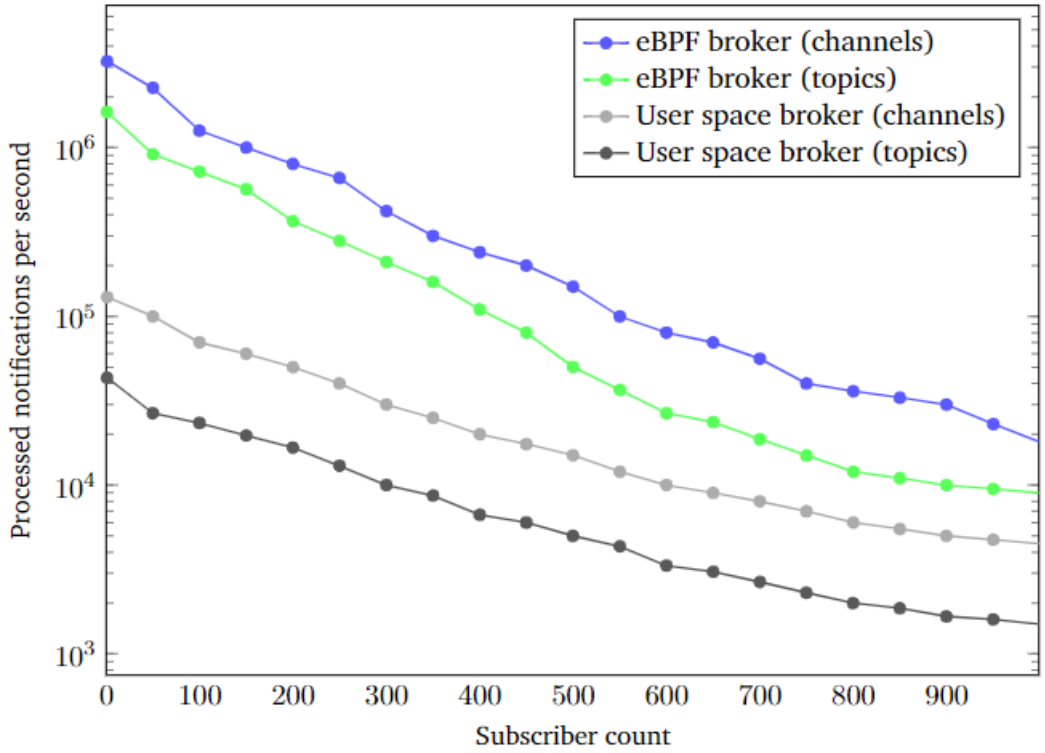
- Channels
  - Implemented with two nested arrays
  - Outer array represents channel number
  - Inner array is used as a routing table



- Topics
  - Similar data structure as channels
  - Every topic must be mapped to an index
  - Each hierarchy layer has its own data structure

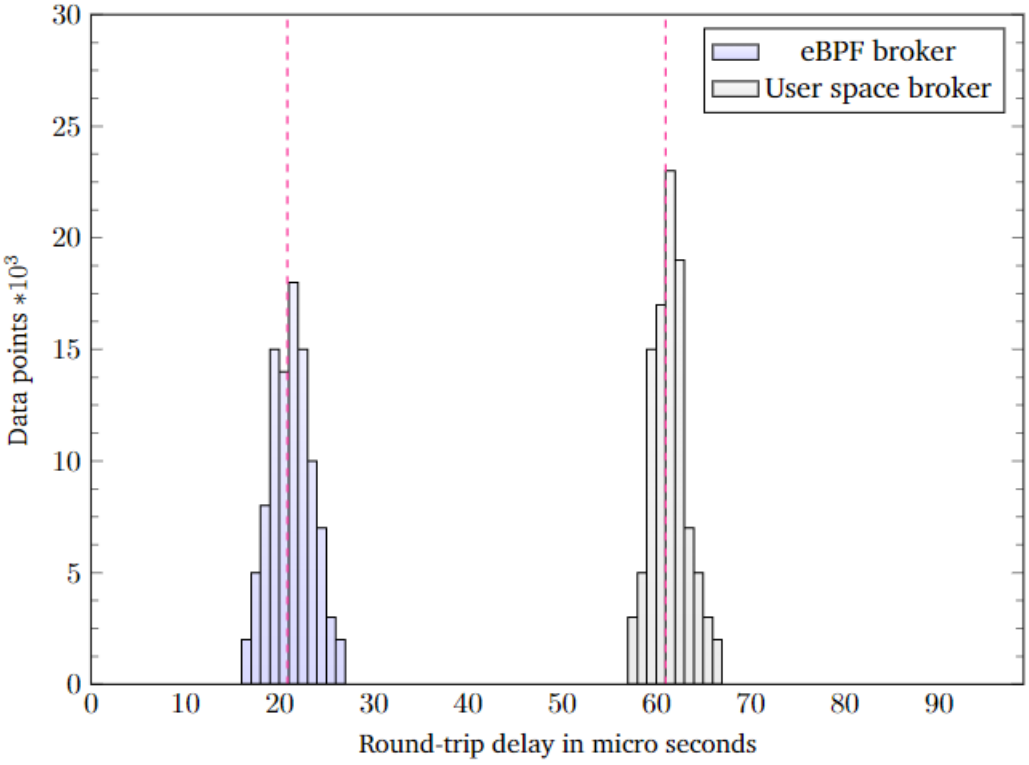


Forwarding



> Result: Increase of packet processing rate by a factor of ≈ 20
(for topic based as well as channel based filter mechanism)

Round Trip Time



> Result: Reduction of latency by a factor of ≈ 3

Conclusions

- Generic implementation of Linux network stack limits performance of broker systems
- Cloud/Edge broker architecture
  - eBPF enables application logic in kernel or driver with limitations
  - Separation in cloud and multiple edge brokers to remedy eBPF limitations
- Evaluation
  - Latency reduction by a factor of ≈ 3
  - Data rate improvement by a factor of ≈ 20
- Outlook
  - Scale out strategies
  - Reliable notification delivery

Thank you for your attention!

Michael Tatarski
michaeltatarski@yahoo.de