ADAPTIVE SOURCE ROUTING AND SPECULATIVE EXECUTION FOR MULTI-HOMED WIRELESS CLIENTS IN PRECLINICAL MEDICAL CARE

Bachelor Thesis

Oliver Michel
oliver.michel@univie.ac.at

November 2012

universität wien
MEDIZINISCHE UNIVERSITÄT WIEN
ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Outline

- Motivation
- Objectives and Measures
- Prototype
- Evaluation
- Conclusions

Motivation I

- tele-medicine becoming more popular in all medical disciplines and application scenarios

[Bergrath et.al. - PLOS ONE 2012]

Motivation 2

- not widely deployed in pre-hospital medical care due to deficient and constrained mobile connectivity
- design an easy to deploy framework usable for any application to enhance mobile network connectivity
- special needs in health-care scenario

Objective

What to achieve?

Objective

What to achieve?

- always available
- responsive
- fast
- secure

Objective

What to achieve?

- always available
- responsive
- fast
- secure

- resilience
- availability
- latency
- throughput
- transparency

Objective

What to achieve?

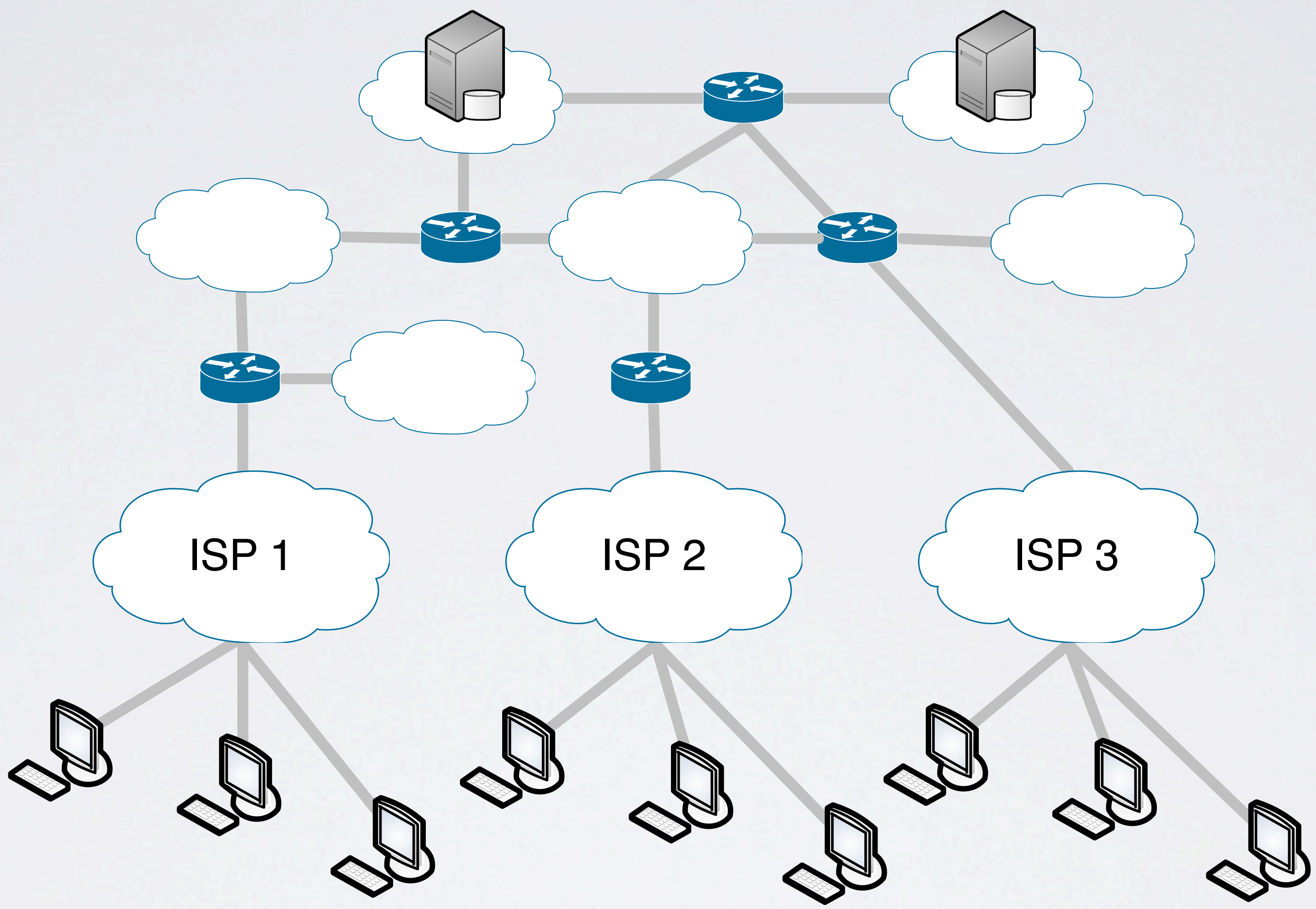
- always available
- responsive
- fast
- secure

- resilience
- availability
- latency
- throughput
- transparency

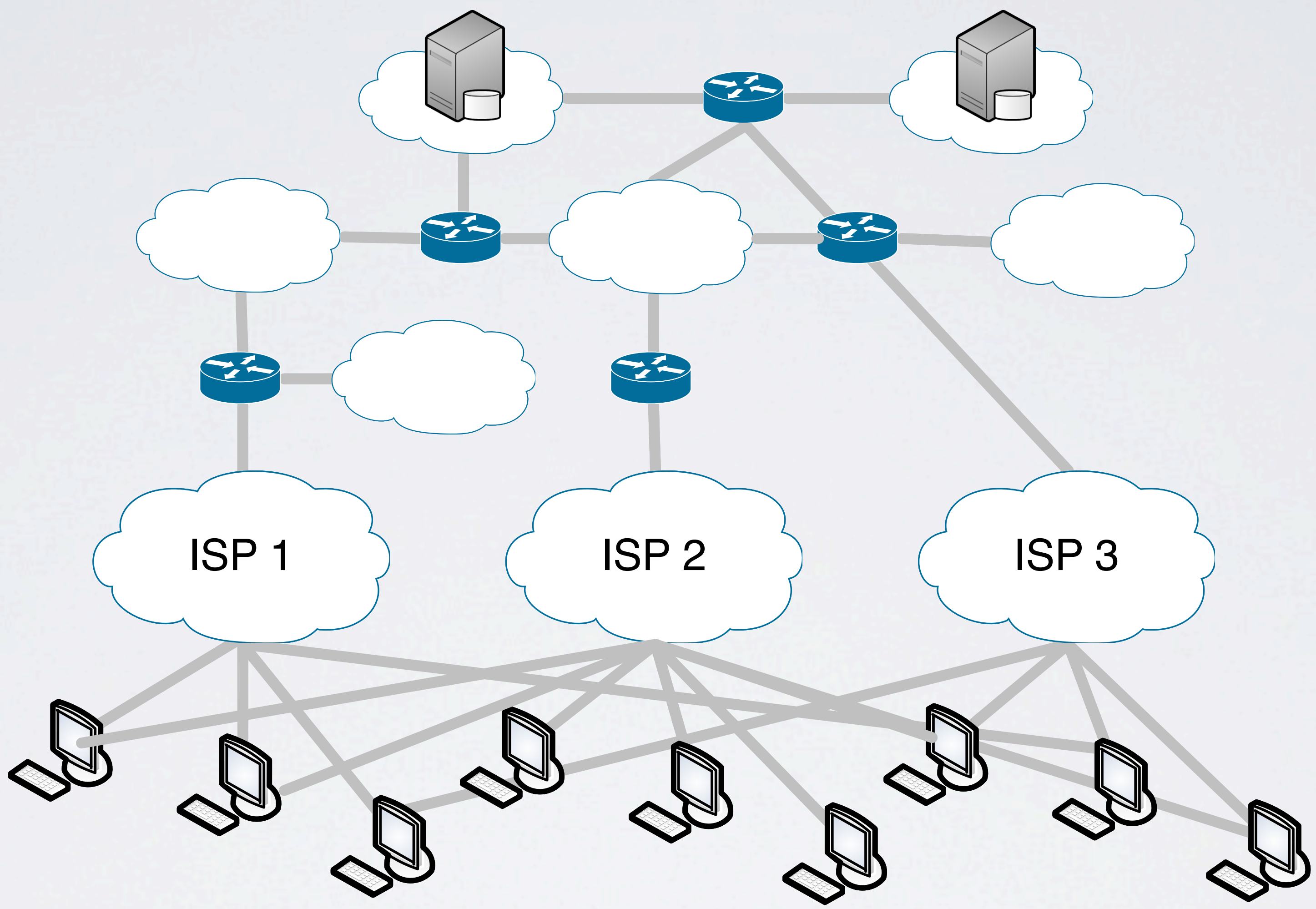
- multi-homing
- resource selection source-routing
- concurrent multipath transmission
- speculative execution

Multi-Homing

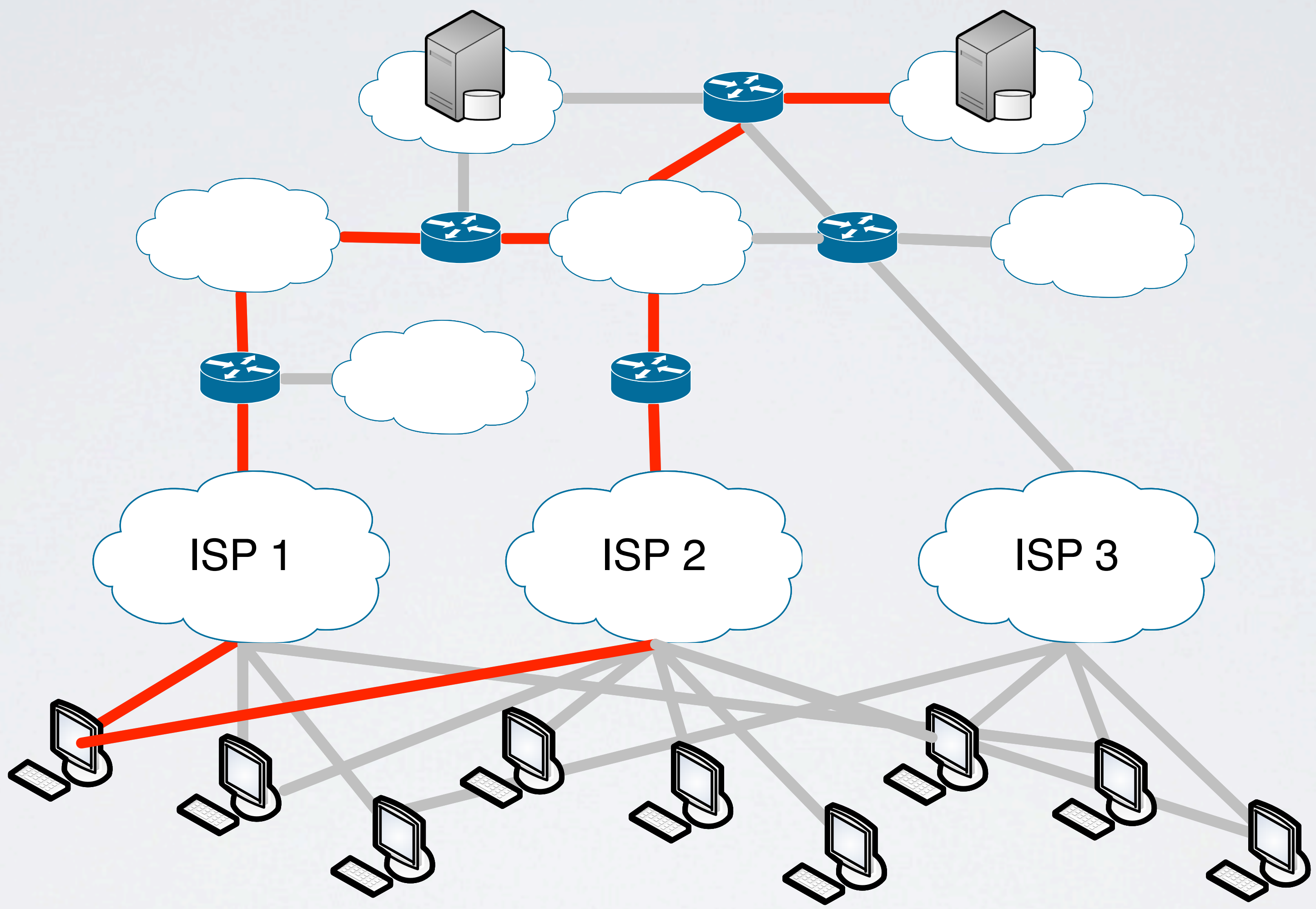
Multi-Homing



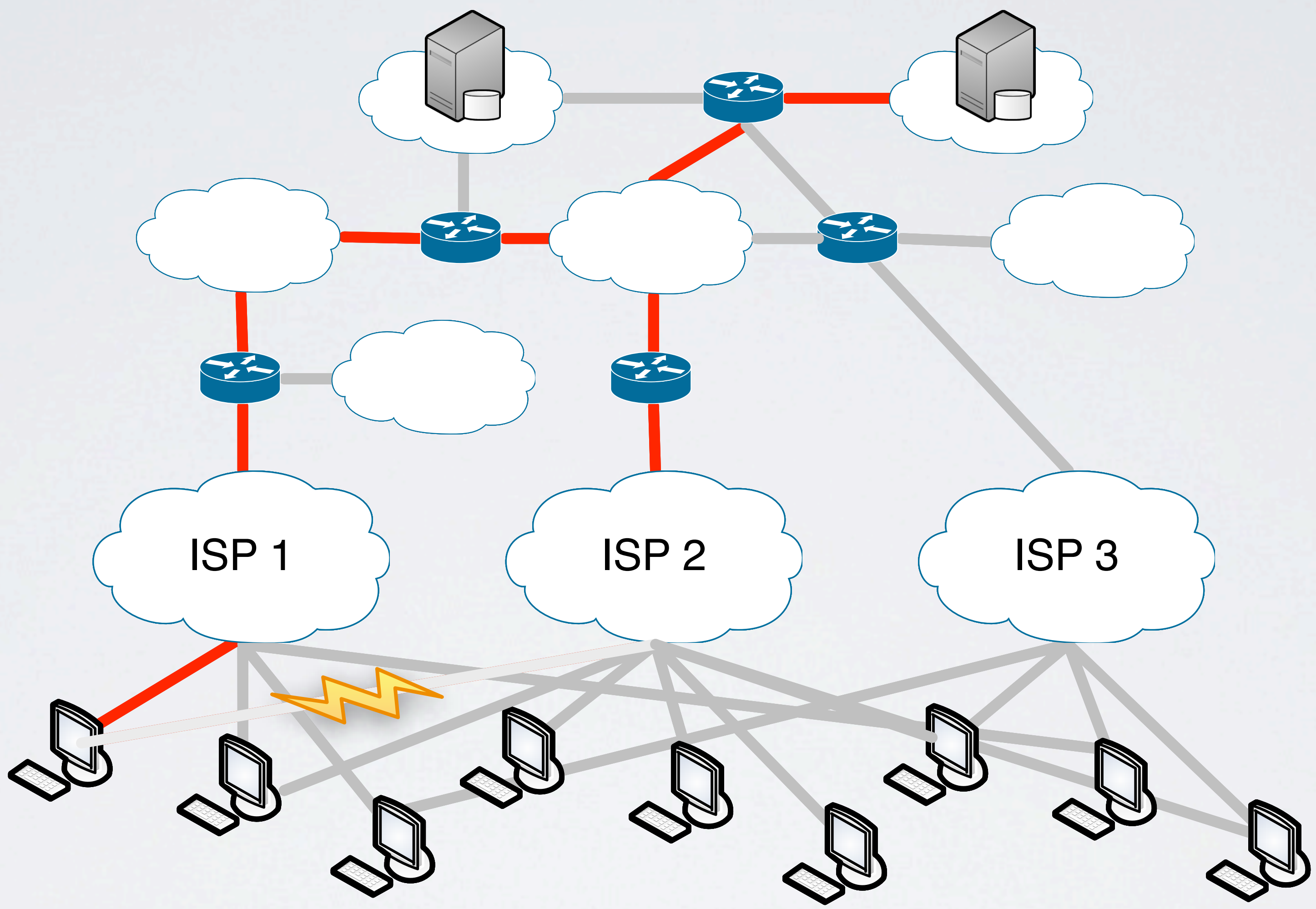
Multi-Homing



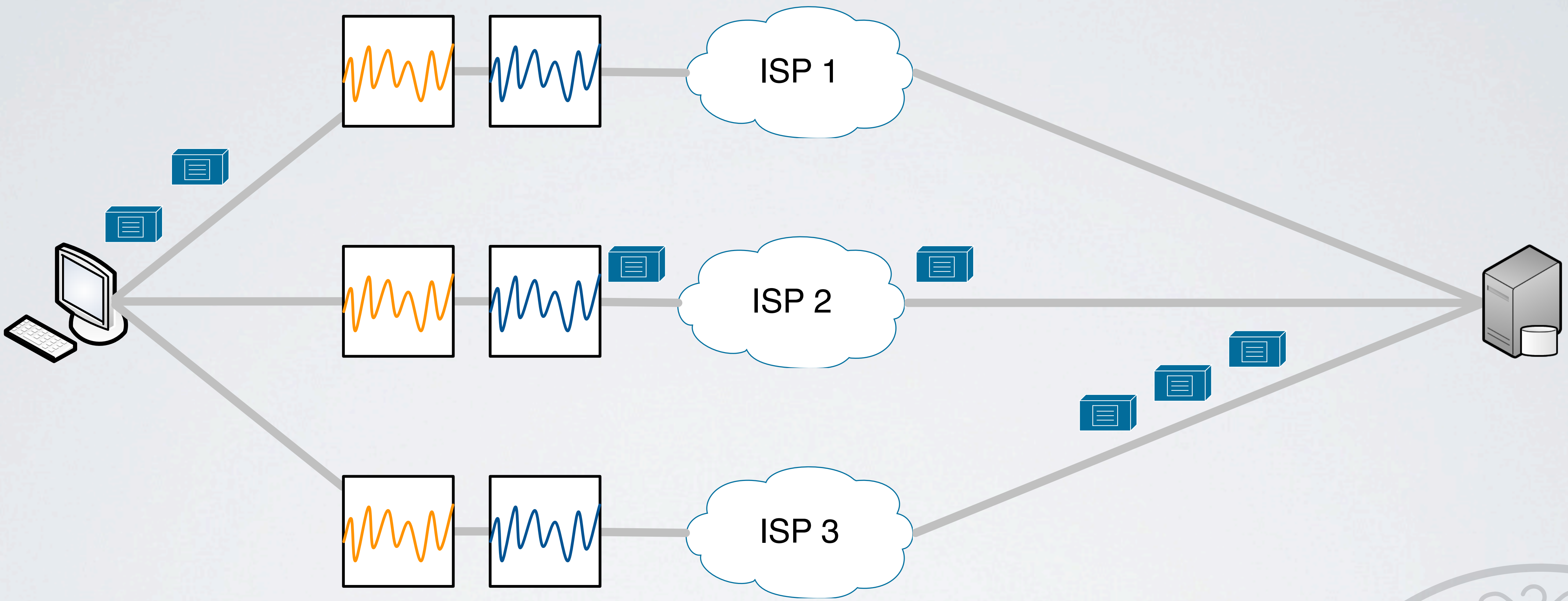
Multi-Homing



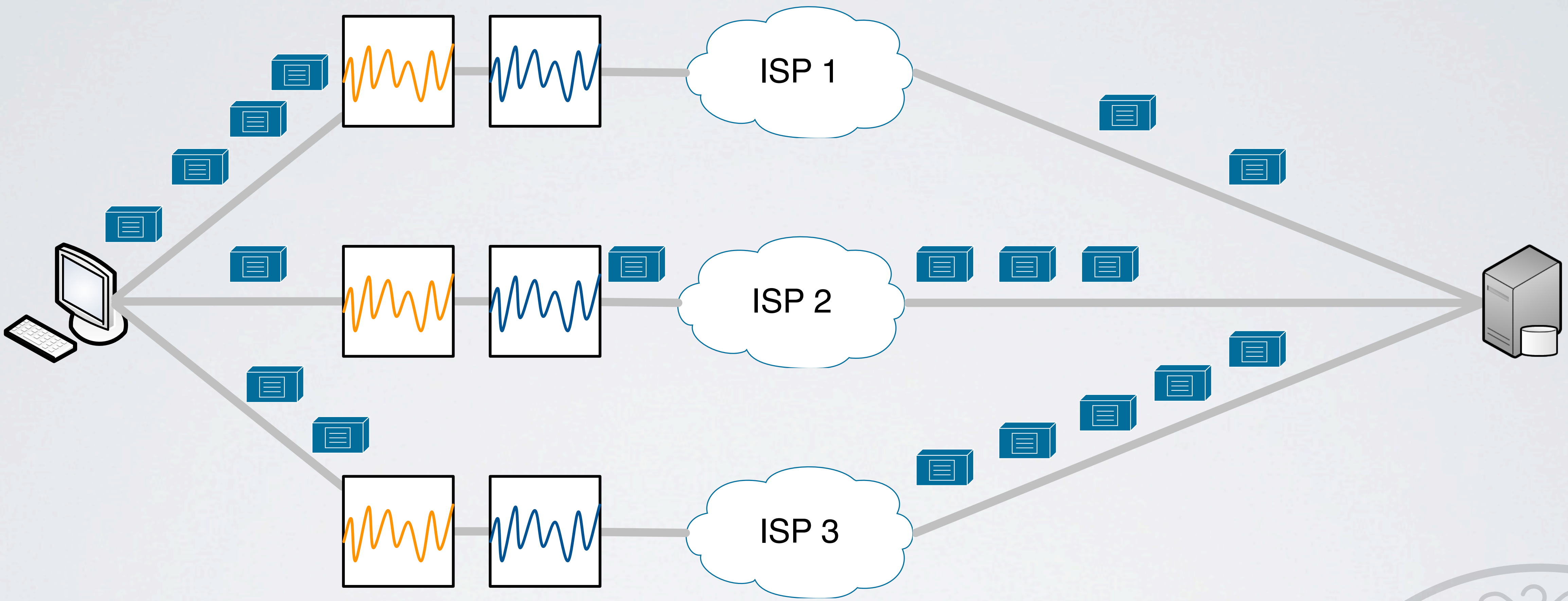
Multi-Homing



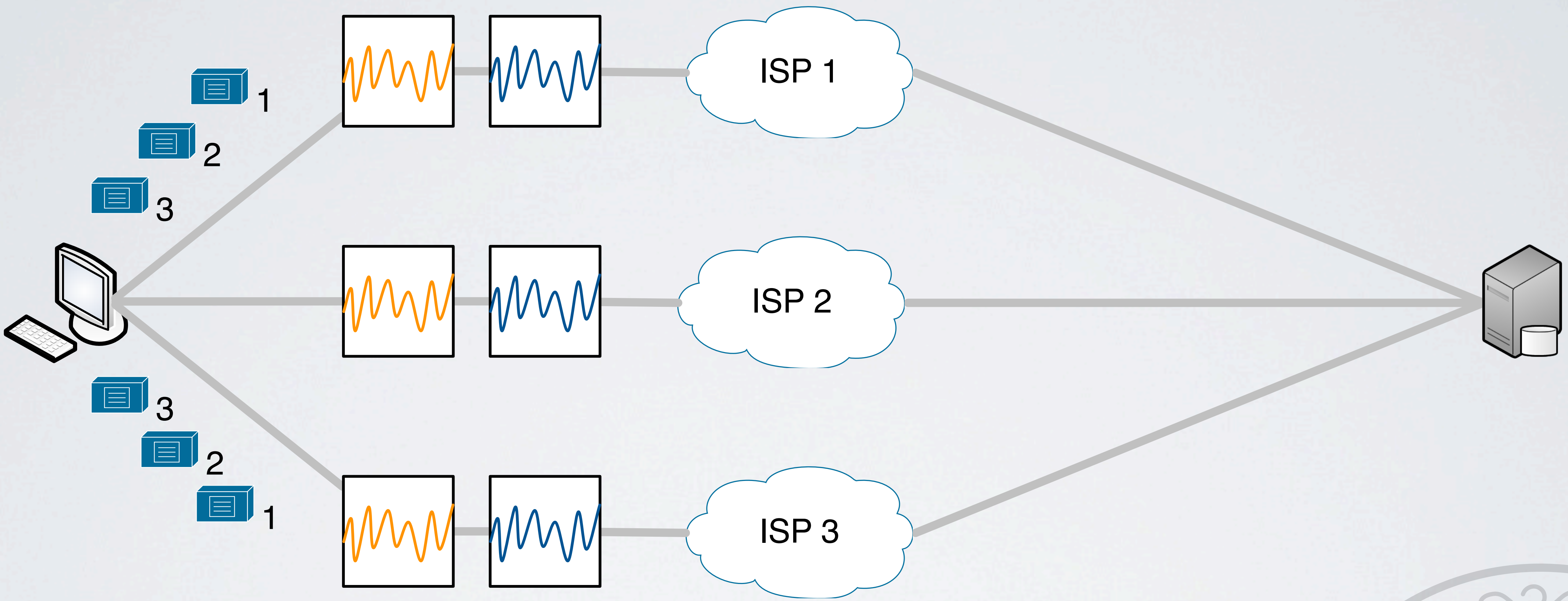
Resource-Selection



Concurrent Multipath Transmission



Speculative Execution



Latency-based Path Selection

path latencies get collected in sliding window matrix:

$$\overrightarrow{l_r} = (l_{r,i}, l_{r,i+1}, ..., l_{r,n}) \qquad L_r = \begin{pmatrix} \overrightarrow{l_{r-\omega}} \\ \vdots \\ \overrightarrow{l_{r-1}} \\ \overrightarrow{l_r} \end{pmatrix}$$

selection function s(L, μ) returns "best" μ path indices according to some objective (here min! latency):

$$s(L, \mu) = (p_0, p_1, ..., p_\mu), \quad \mu \leq n$$

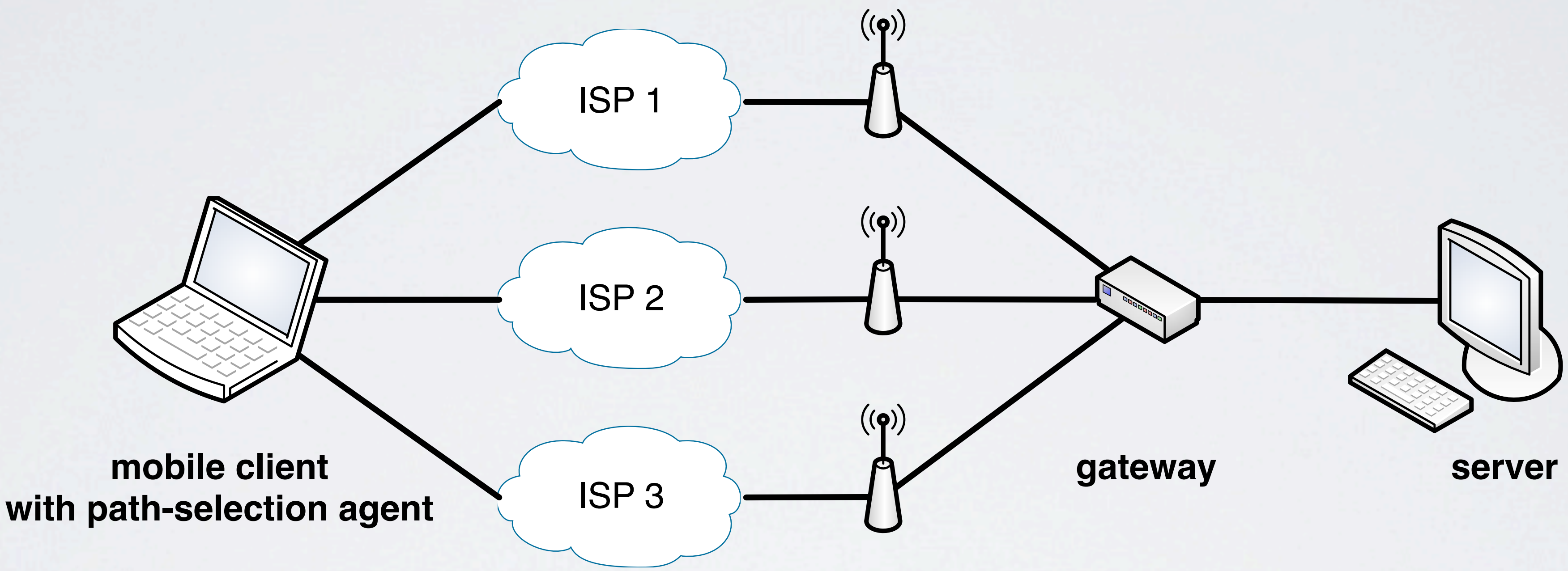
path decision was correct if:

$$\min(\overrightarrow{l_{r+1}}) = l_{r+1, s_r(L,\mu)}$$

Requirements

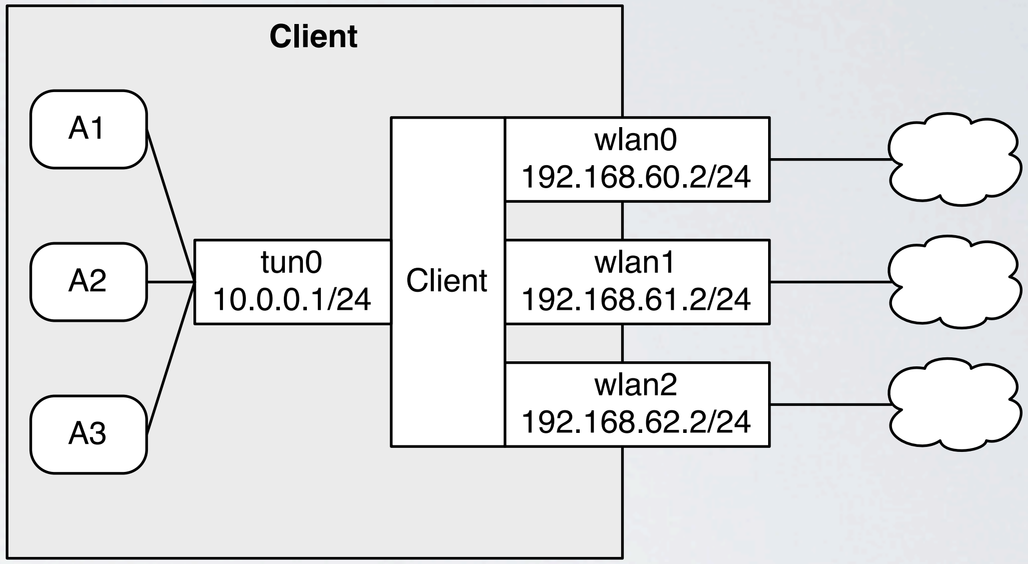
- improve data-transmission between a mobile client and arbitrary servers
- transparently usable by any application
- executable in user-space

System Architecture



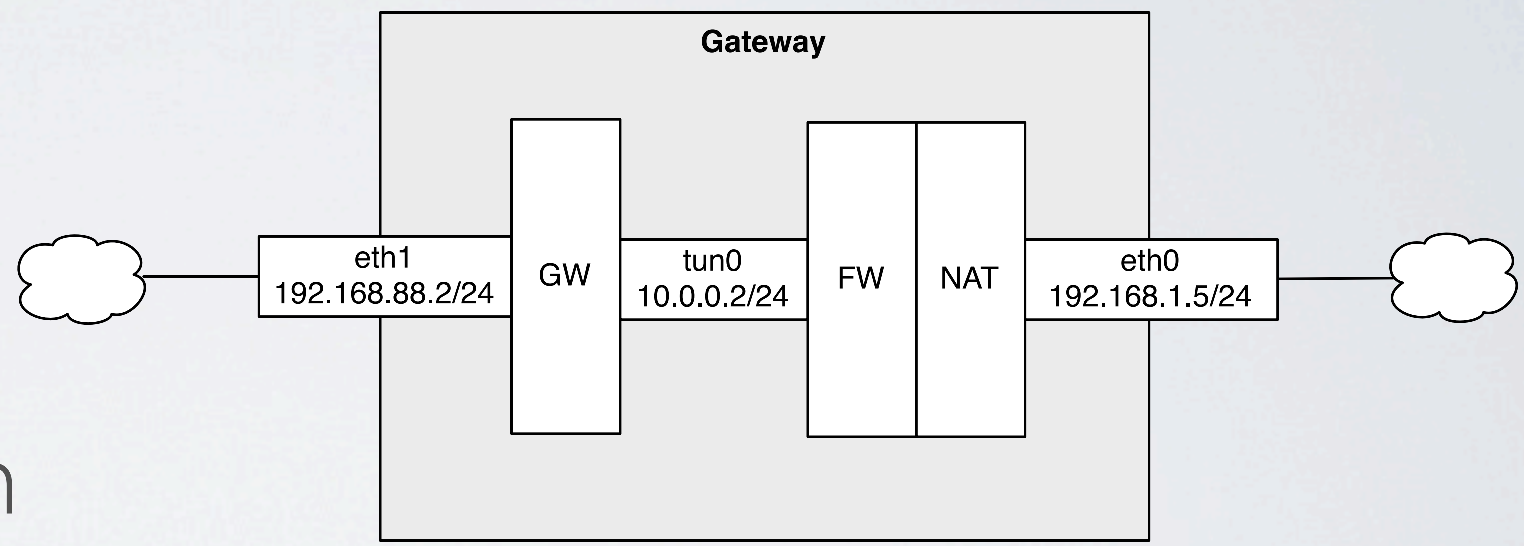
System Architecture: Client

- controls active probing
- performs path-selection
- reads data from tunnel-device
- encapsulates data and sends over selected path

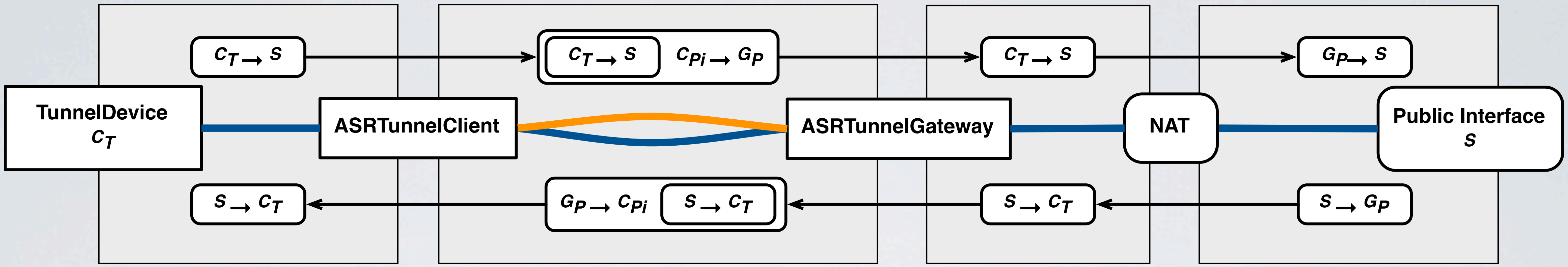


System Architecture: Gateway

- reads data from public interface
- removes custom header
- writes data to tunnel device for forwarding in kernel and NAT
- (also does active probing on return path)



System Architecture: Packet Encapsulation

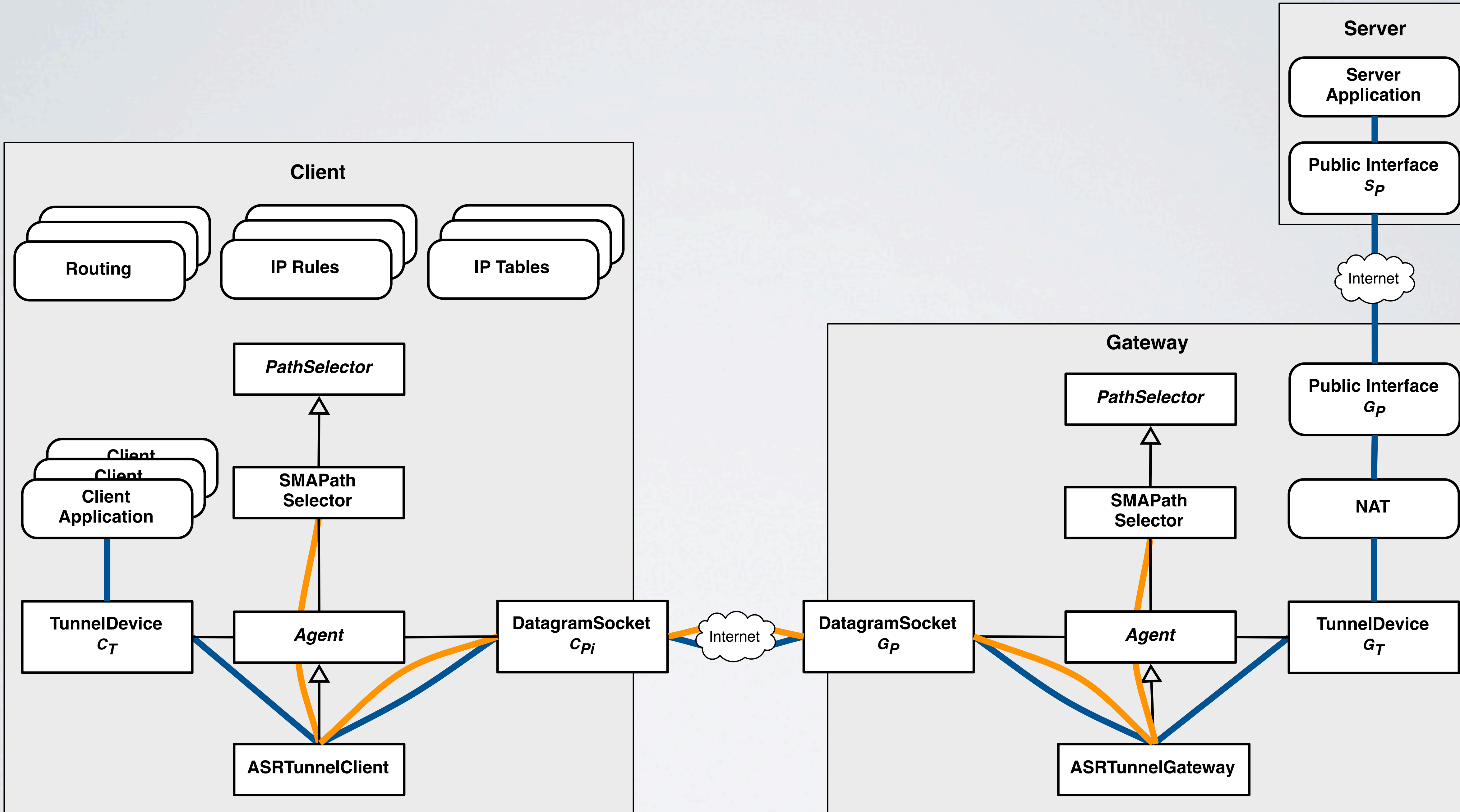


- IP-layer packets get encapsulated as UDP payload for transmission over tunnel

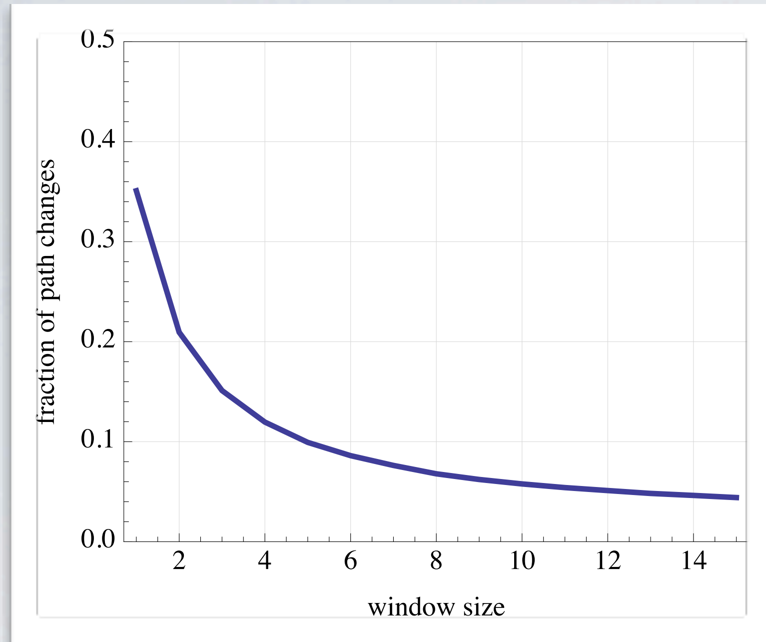
- custom 12 Byte packet header:

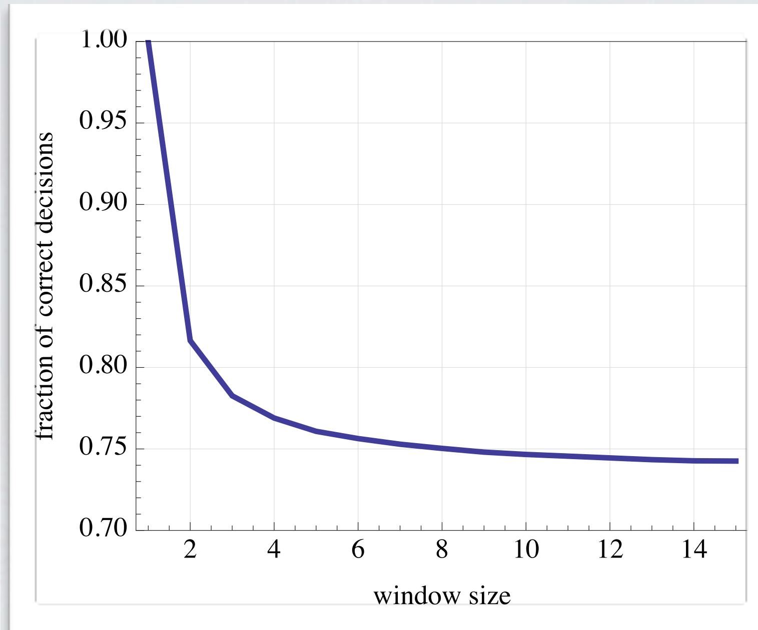
```
uint8_t _flags;
uint8_t _client_id;
uint16_t _pl_length;
uint32_t _seqn;
uint32_t _client_addr;
```

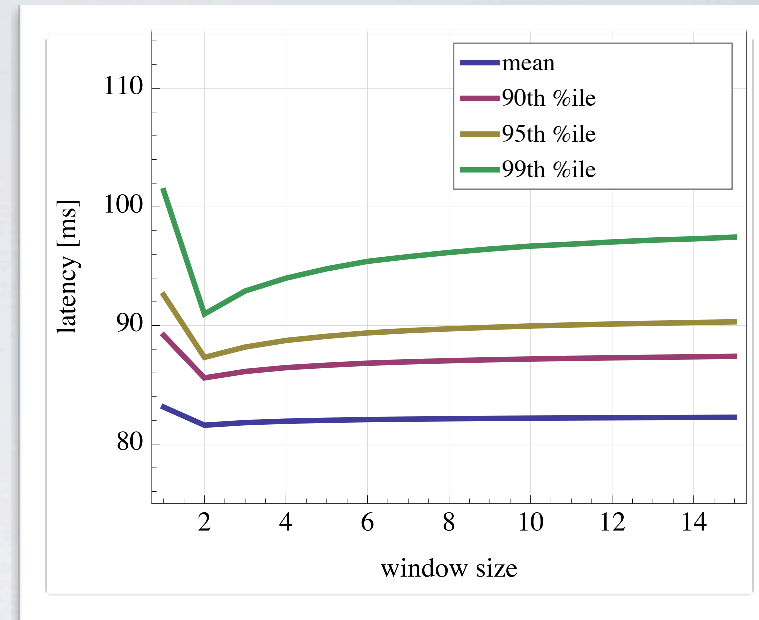
System Architecture: Modules



Results: Path Selection I



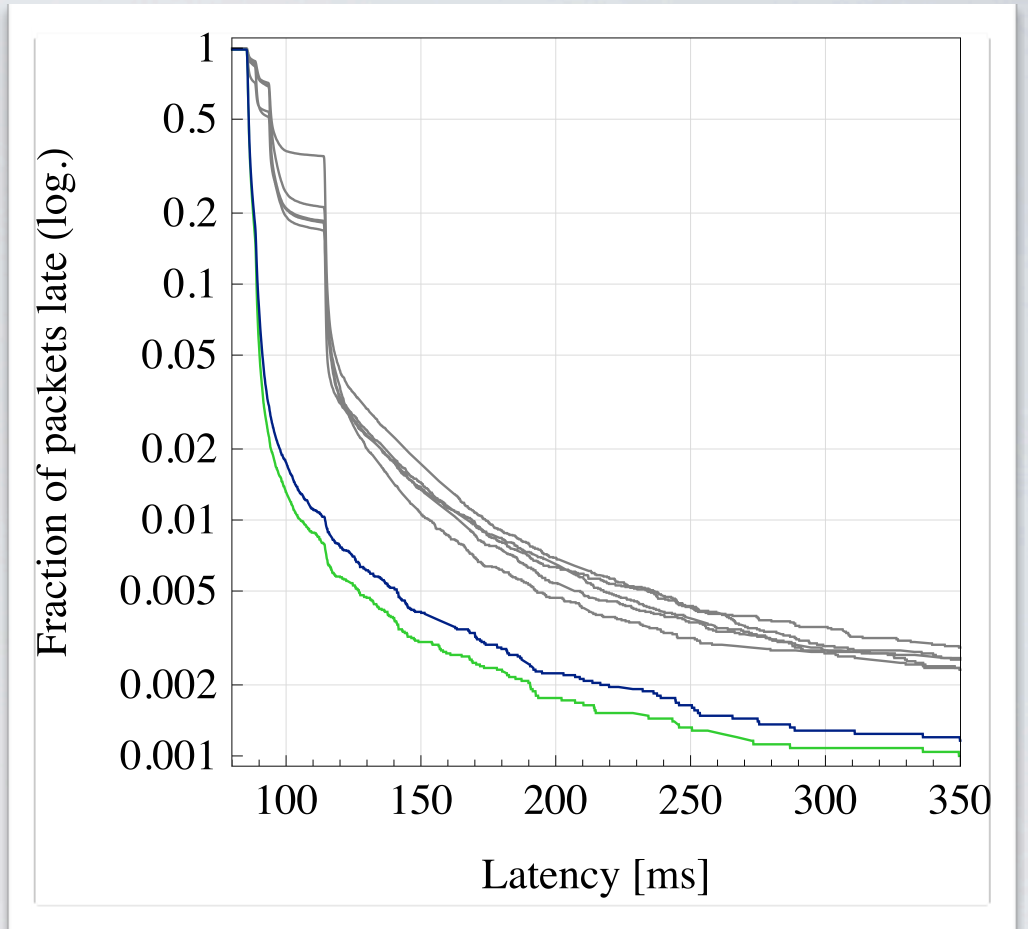


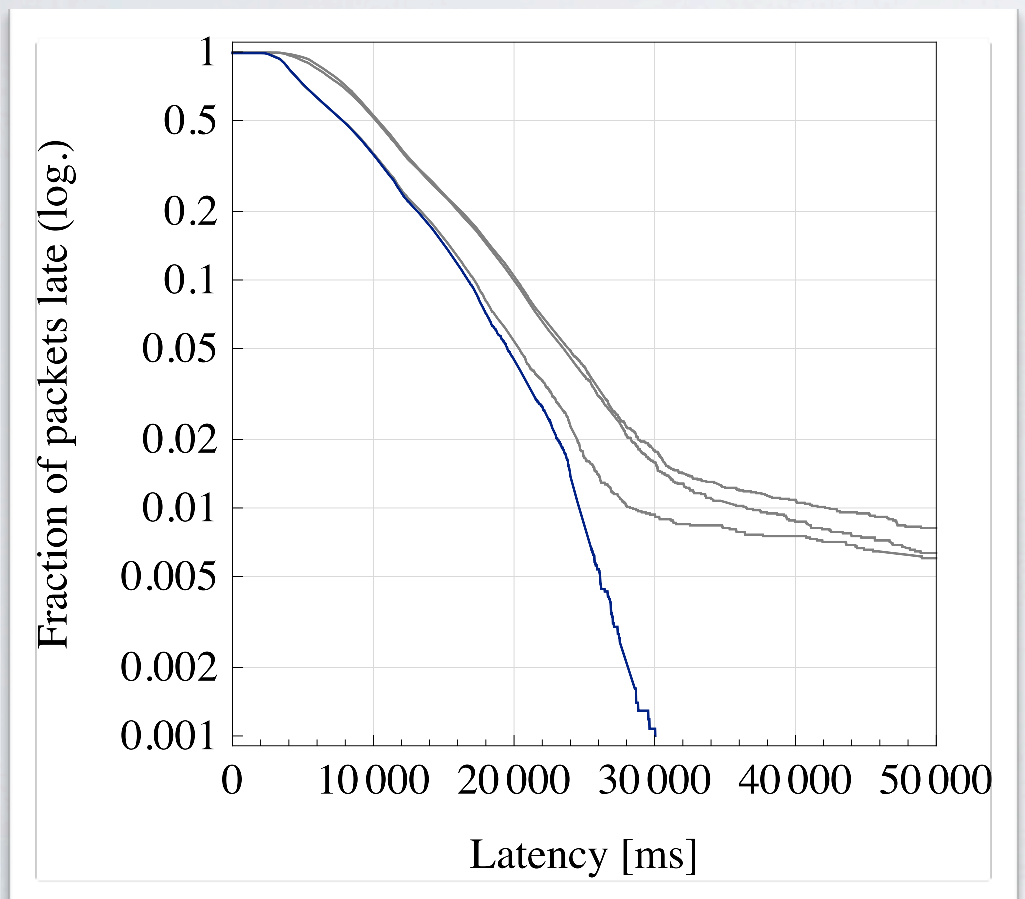


- post-hoc analysis of SMA, TPMA, and EWMA algorithms on measured latency data (PlanetLab) and sampled values from Pareto-Type I and Gaussian
- for window-based algorithms a window size between 4-6 seems reasonable

Results: Path Selection 2

- SMA5 path-selection from data collected from a PL tripartide overlay graph
- SMA surprisingly good, close to optimal (in retrospect)
- TPMA, EWMA slightly inferior

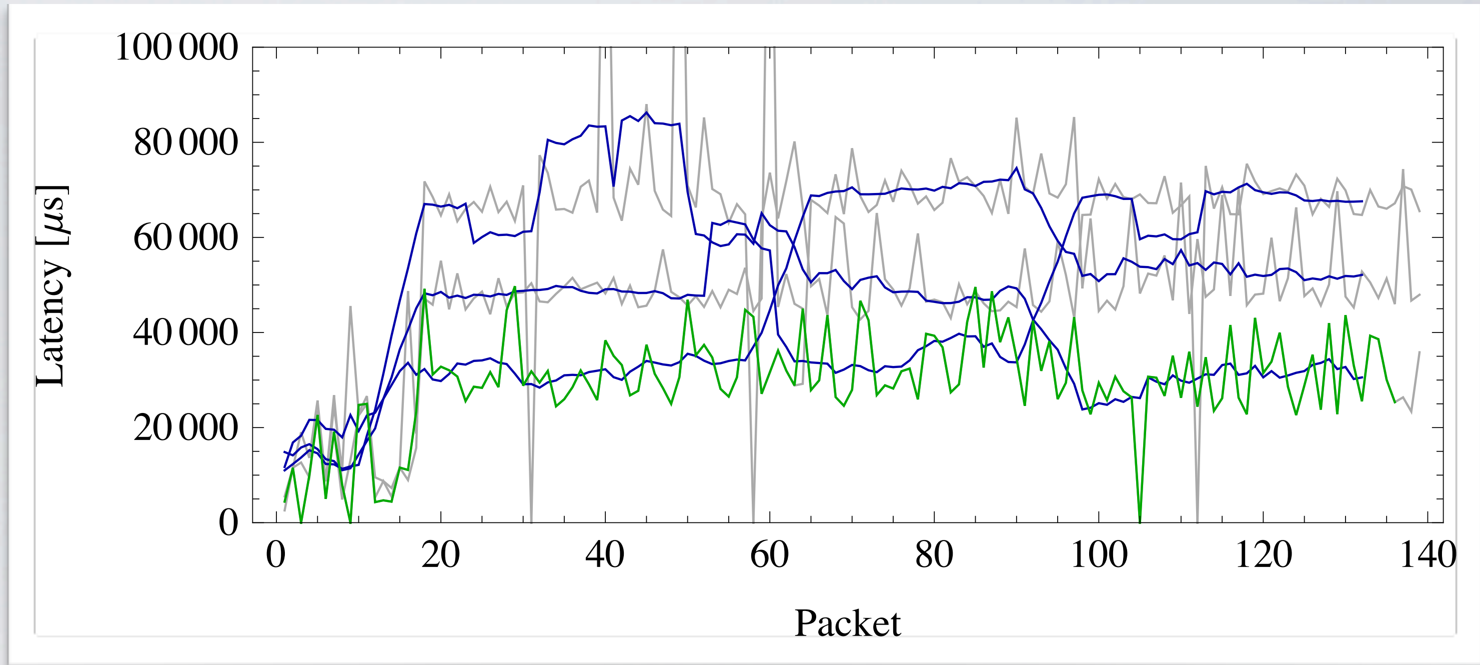


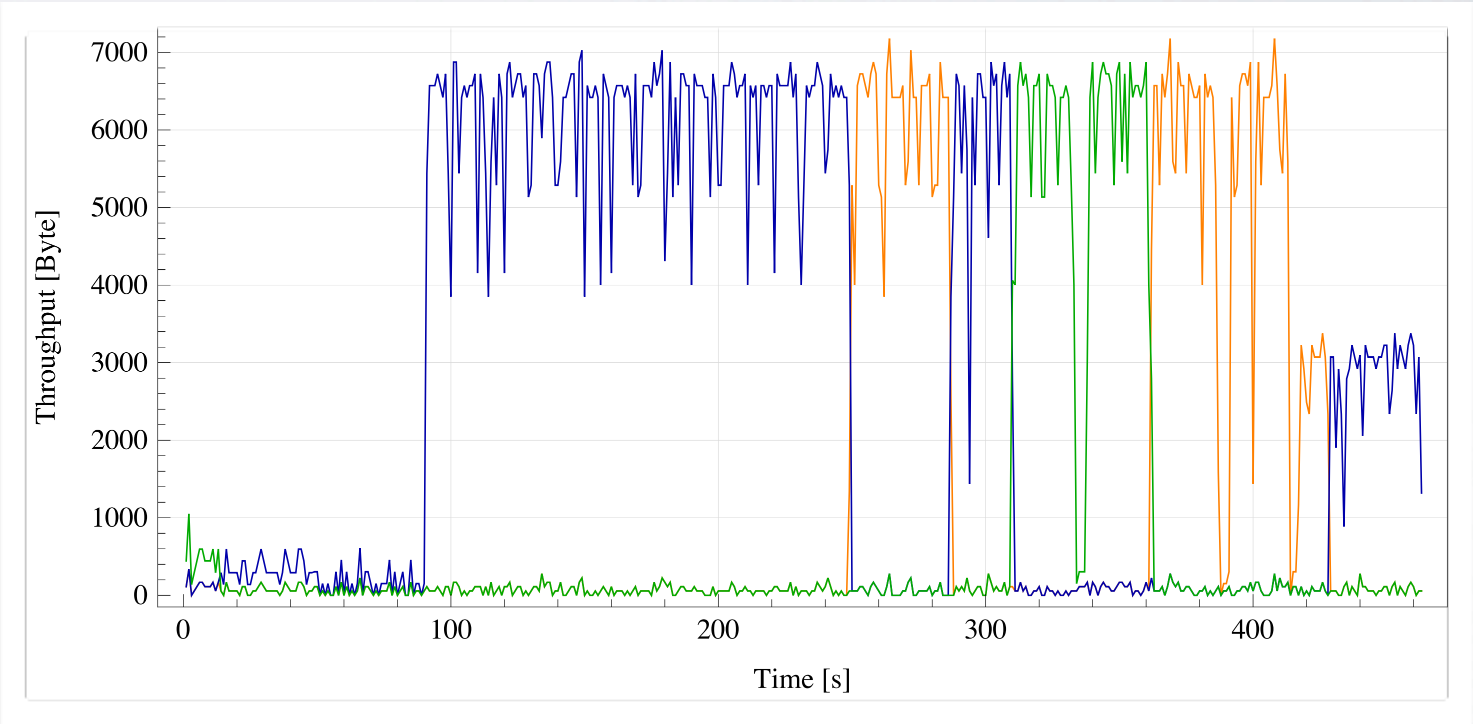


- SMA5 using data-flow (64-128kbit/s) on wireless testbed
- latencies of probe, as well as data traffic (1 per second, exponential)

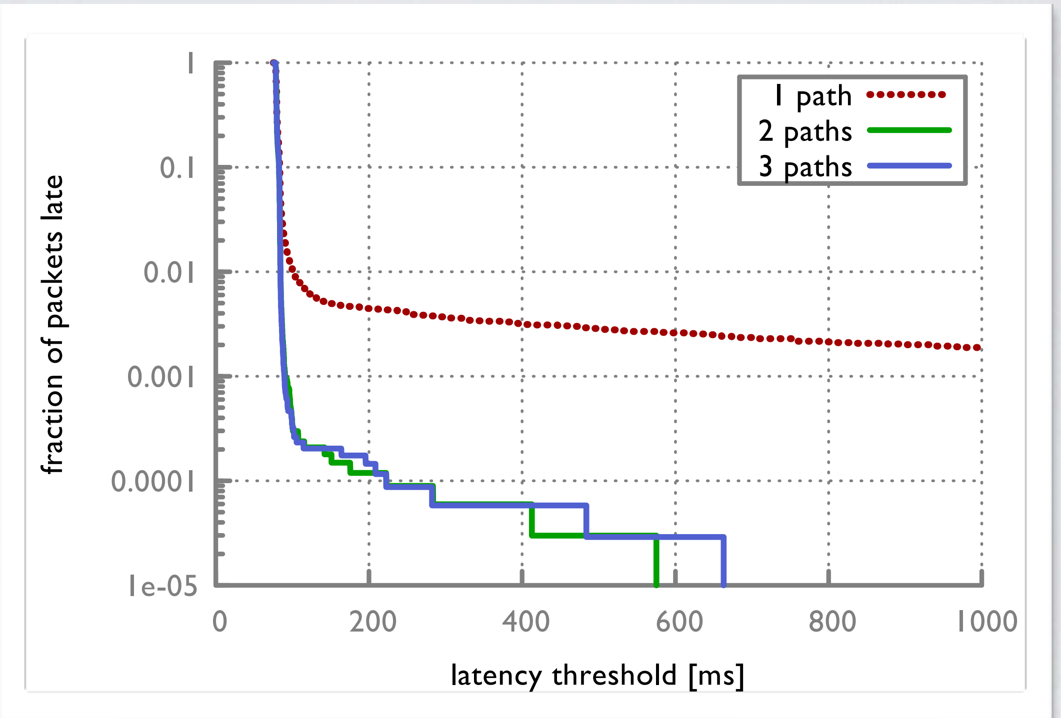
Results: Fast Adaption and Handover

- measured latency data modified using netem
- satisfying handover-behavior
- little occurrence of packet losses while switching paths
- continuous high data throughput





Results: Packet Replication



- redundant multipath transmission (i.e. packet duplication) may significantly reduce the latency tail
- virtually no packet loss in case of path failure
- general technique for system designers when secondary resources would be idling normally [Vulimiri et. al. - Hotnets '12]

Monday, February 11, 13

Future Directions

- detailed Investigation of TCP behavior
  - TCP connections sometimes get stuck when paths switch frequently
- IP Fragmentation
  - iptables NAT functionality does not support fragmented IP packets
- more detailed parameter study for path-selection
  - other algorithms (e.g. multi-armed bandit)
- experimentation with different probing intervals
- consideration of further metrics (esp. bandwidth)

Conclusions

- transparently usable framework designed to enhance mobile data connectivity
- technically feasible and performant
- modular design allowing system designers to attach custom logic and state
- investigation of reasonable approximations for the Internet path-selection problem
- focus on special needs and applicability in the health-care field (esp. preclinical medical care)