Institut für Kommunikations-Technik

MIDAS: Middlebox Discovery and Selection for On-Path Flow Processing

SDN Workshop
Zurich, November 2015

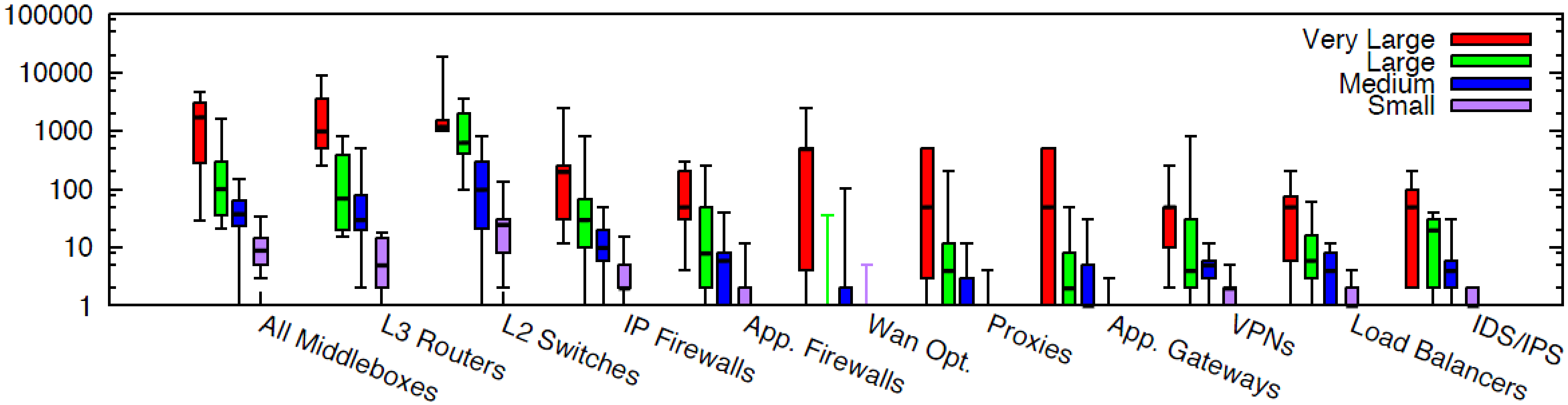
Panagiotis Papadimitriou
Leibniz Universität Hannover

In collaboration with:
Ahmed Abujoda (Leibniz Universität Hannover)

Leibniz Universität Hannover

Introduction

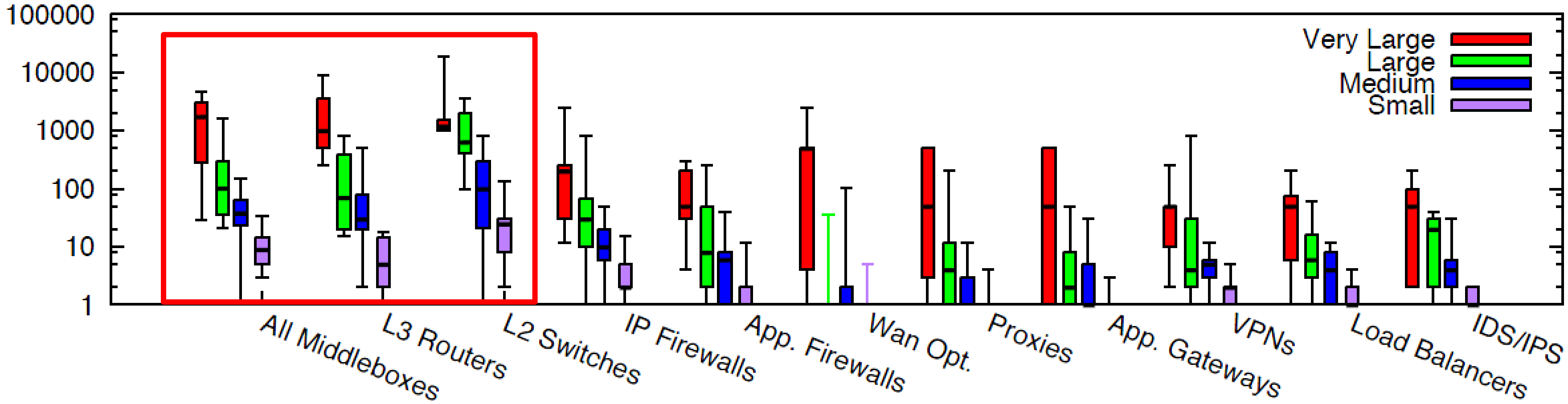
- Proliferation of middleboxes in enterprise networks:
  - packet filtering
  - proxies
  - load balancing
  - redundancy elimination
  - encryption
  - …..



J. Sherry et al., **Making Middleboxes Someone Elses Problem: Network Processing as a Cloud Service**, SIGCOMM 2012

Introduction

- Proliferation of middleboxes in enterprise networks:
  - packet filtering
  - proxies
  - load balancing
  - redundancy elimination
  - encryption
  - …..



J. Sherry et al., **Making Middleboxes Someone Elses Problem: Network Processing as a Cloud Service**, SIGCOMM 2012

Introduction

- Recent trends
  - Packet processing on commodity servers
    - Consolidated SW middleboxes [Flowstream, CoMB, Vyatta]
  - Micro-datacenter deployment by ISPs

- Migration of middleboxes to the network
  - Reduced CAPEX/OPEX for enterprise networks
  - Elastic provisioning
  - Empowering the “middle”

A. Greenhalgh et al, **Flow Processing and the Rise of Commodity Network Hardware**, CCR 2009
V. Sekar et al., **The Design and Implementation of a Consolidated Middlebox Architecture**, NSDI 2012

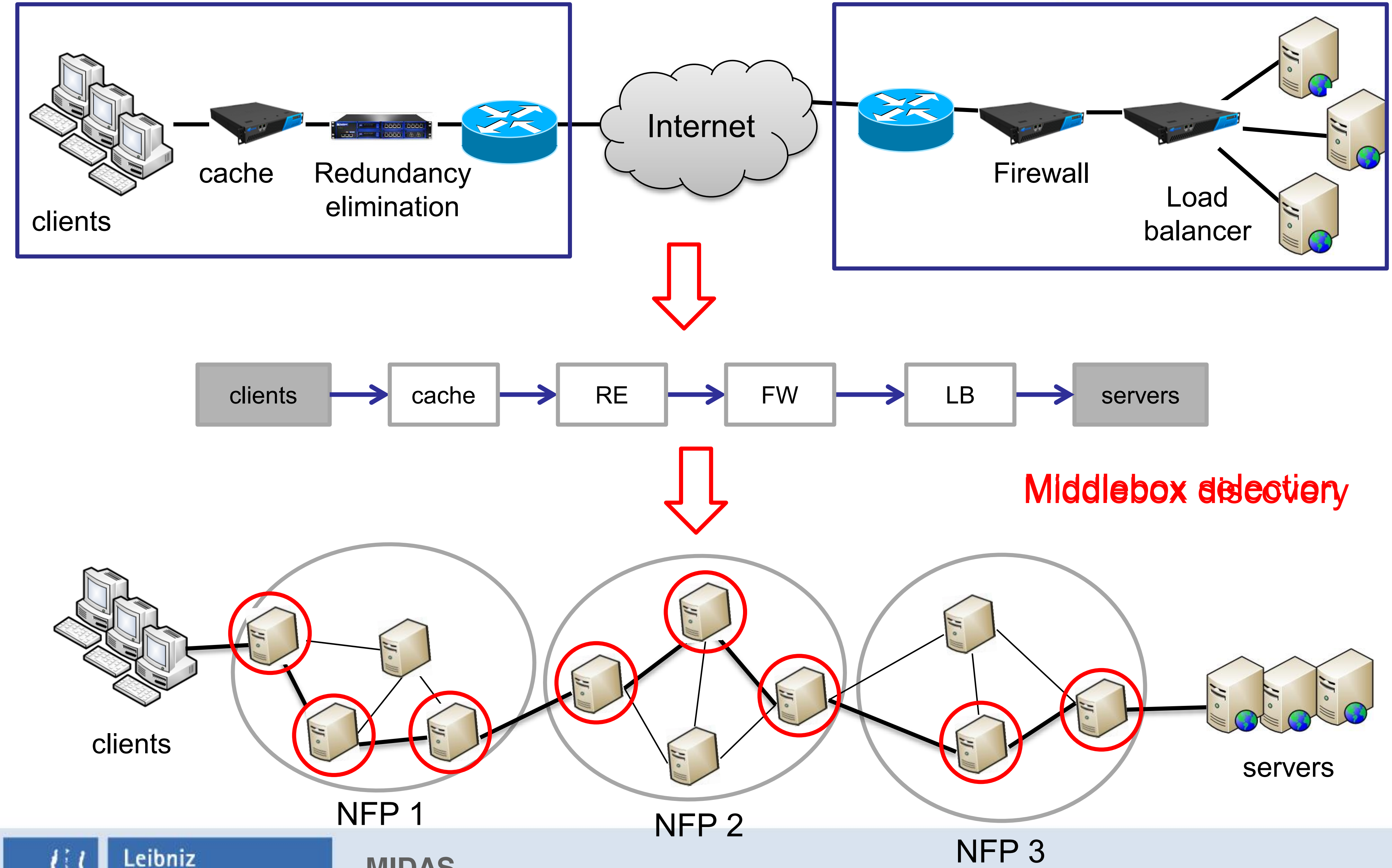
Network Processing

- Processing off the traffic path [**Nestor**, IFIP Networking 2015]
  - ✓ Resource sharing
  - ✘ Need for traffic redirection to datacenters [APLOMB]
    - Latency inflation
  - ✘ Data-center access link overload

- Processing on the traffic path [**MIDAS**, IEEE COMSNETS 2015]
  - ✓ No need for traffic redirection
  - ✘ Robustness

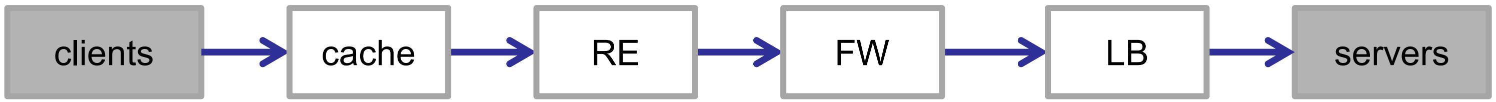
J. Sherry et al., **Making Middleboxes Someone Elses Problem: Network Processing as a Cloud Service**, SIGCOMM 2012

On-Path Flow Processing



Network Processing Requirements

- Performance
  - High packet forwarding rates [RouteBricks, ClickOS]
  - Low processing setup delay
- Load balancing
- Correctness
  - Network functions (NFs) should be deployed in the correct order



M. Dobrescu et al., **RouteBricks: Exploiting Parallelism to Scale Software Routers**, SOSP 2009
J. Martins et al., **ClickOS and the Art of Network Function Virtualization**, NSDI 2014

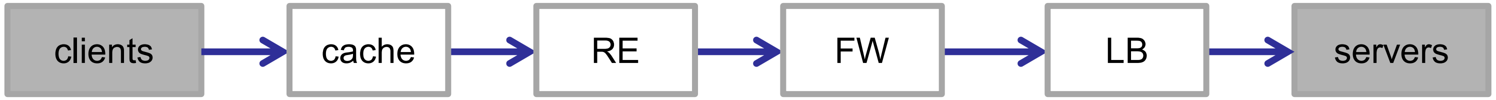
Middlebox Discovery and Selection

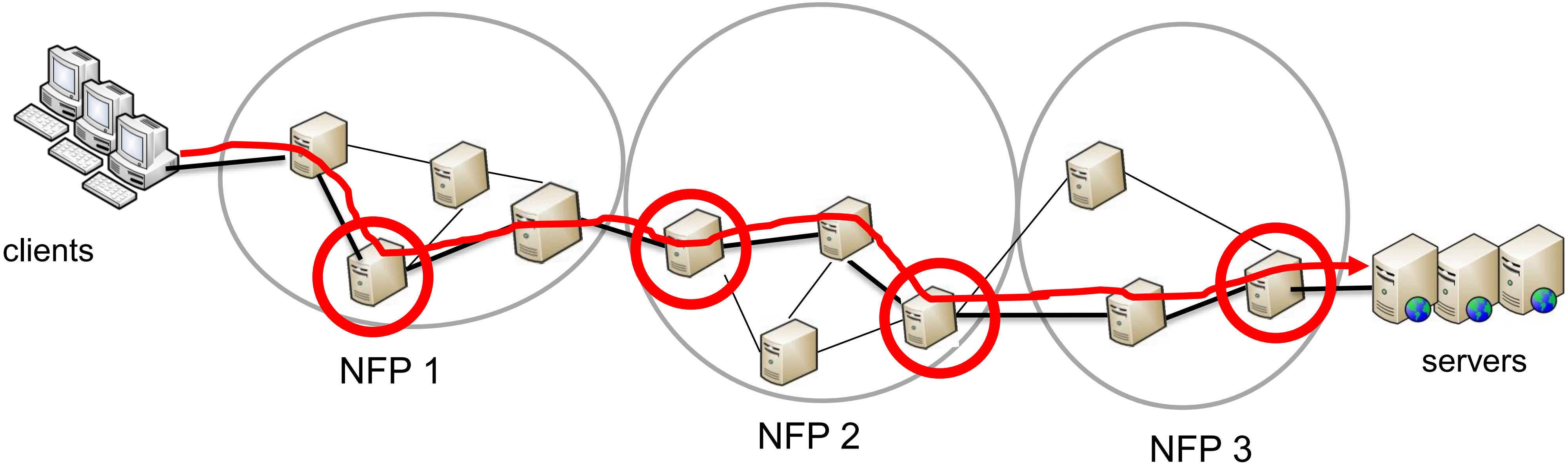
- Middlebox discovery
  - Path discovery and middlebox detection techniques (e.g., traceroute, tracebox) incur high delays
  - Signaling protocols (e.g., SIMCO) are designed for middlebox configuration

- Middlebox selection
  - NF location dependencies require large provider footprint (i.e., multiple NFPs)
    - NFP resource information disclosure policies

On-the-fly Processing Setup

Institut für Kommunikations-Technik



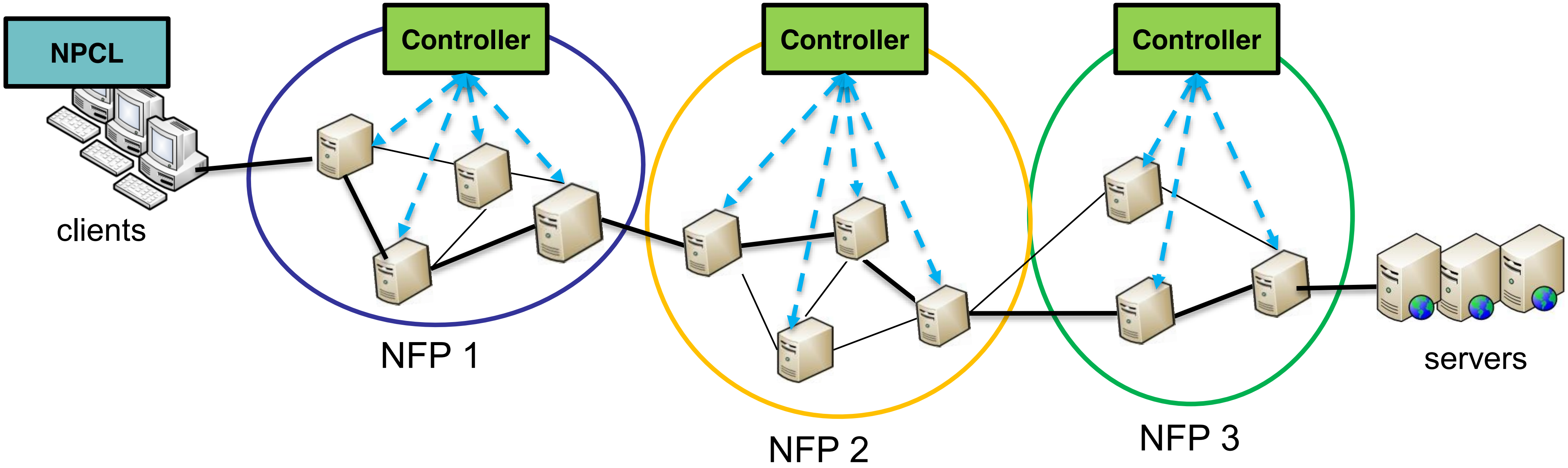


- Middleboxes pick up flows as they arrive
  - ✓ Performance
  - ✘ Trade-off between correctness and load balancing

- Need for processing setup coordination within and across NFPs

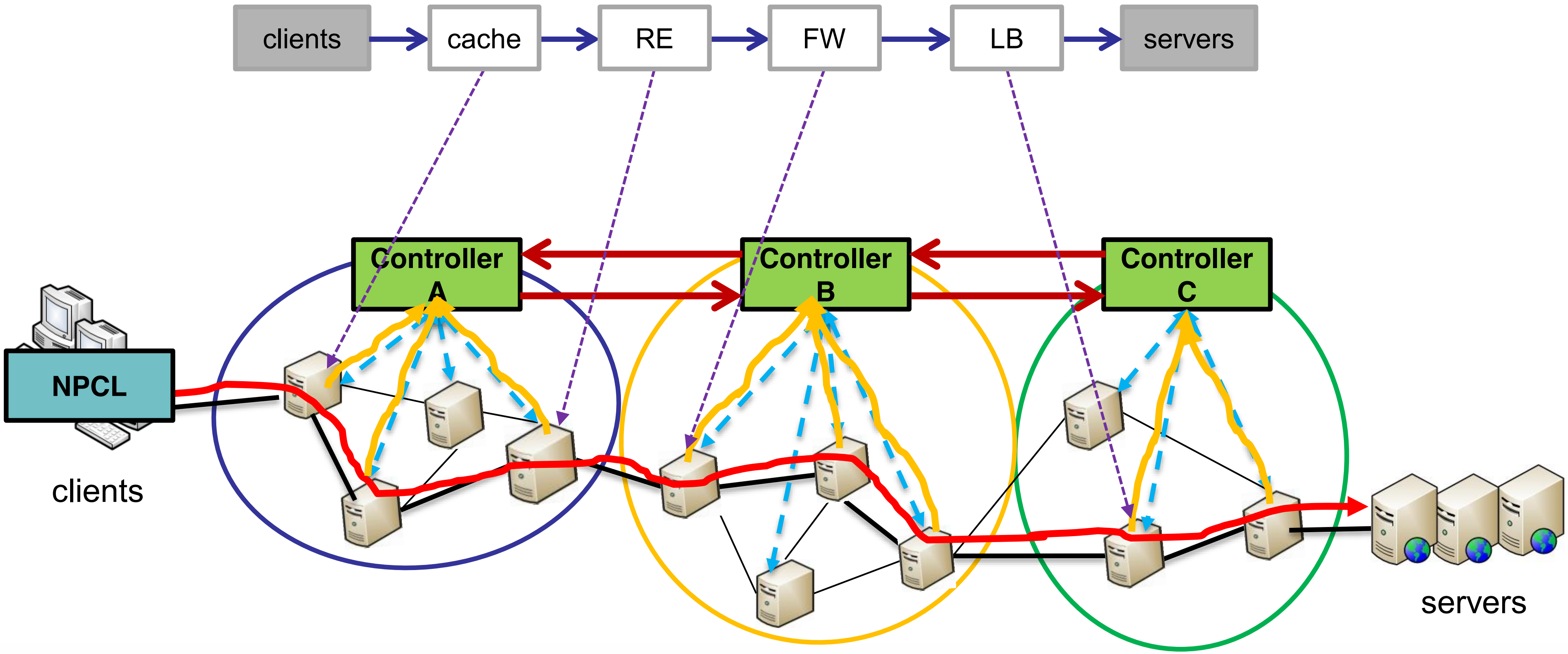
MIDAS Overview

- Main components:
  - Consolidated middlebox (CoMB)
  - Centralized CoMB controller in each NFP
  - Network processing client (NPCL)



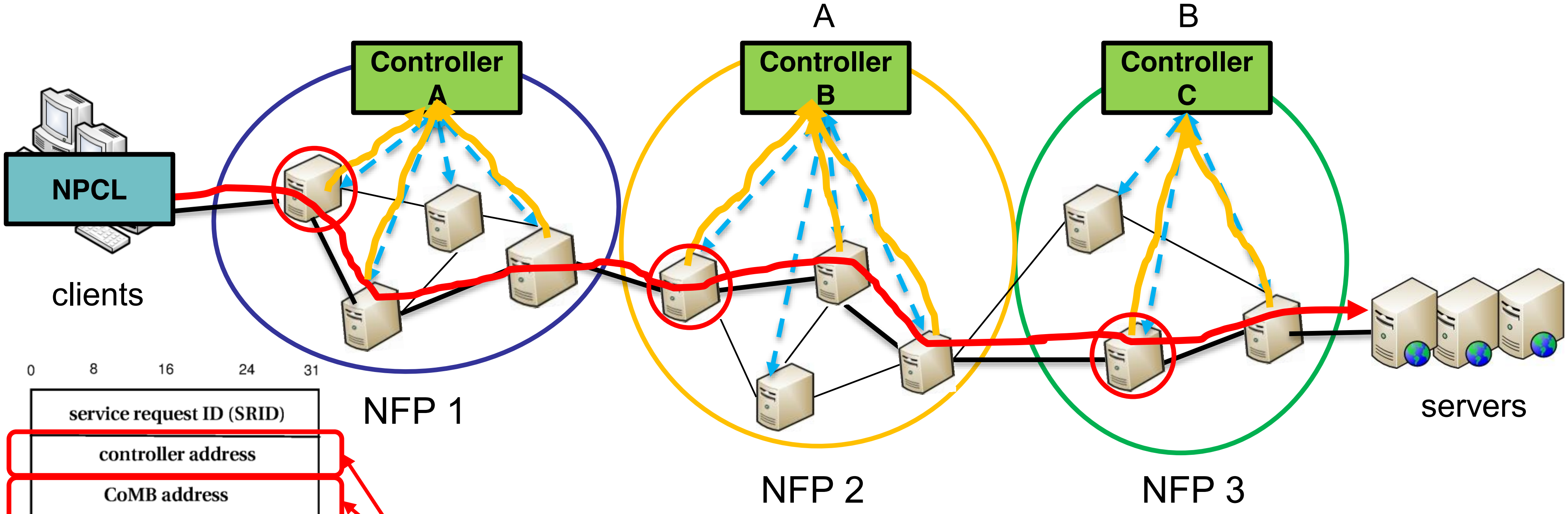
MIDAS Overview

Middlebox Signaling → Controller Notification → Controller Chaining → Middlebox Selection → NF Setup



Middlebox Discovery

Middlebox Signaling



0–31
service request ID (SRID)
controller address
CoMB address
processing type identification
proximity
bandwidth
source IP (flow specification)
destination IP (flow specification)
source port / destination port (flow specification)
protocol (flow specification)

REQUEST message

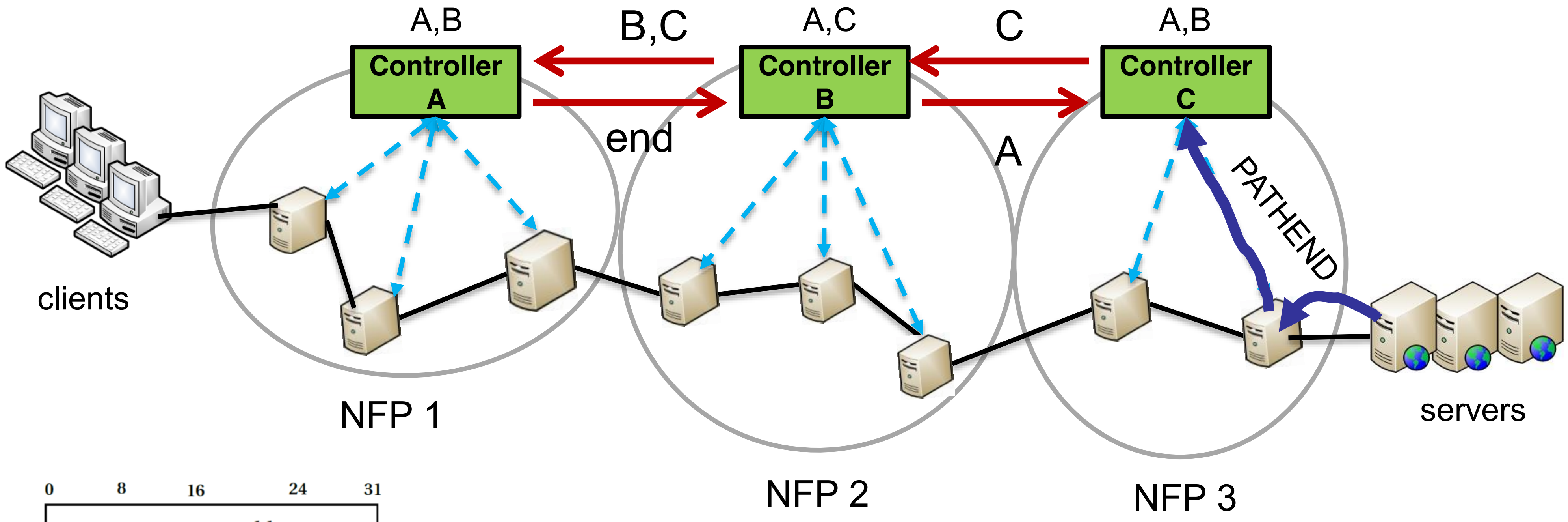
Extracted by each ingress CoMB and inserted into the DISCOVERY message

Each CoMB inserts its IP address

0–31
CoMB ID
CoMB utilization
REQUEST message

DISCOVERY message

Controller Chaining



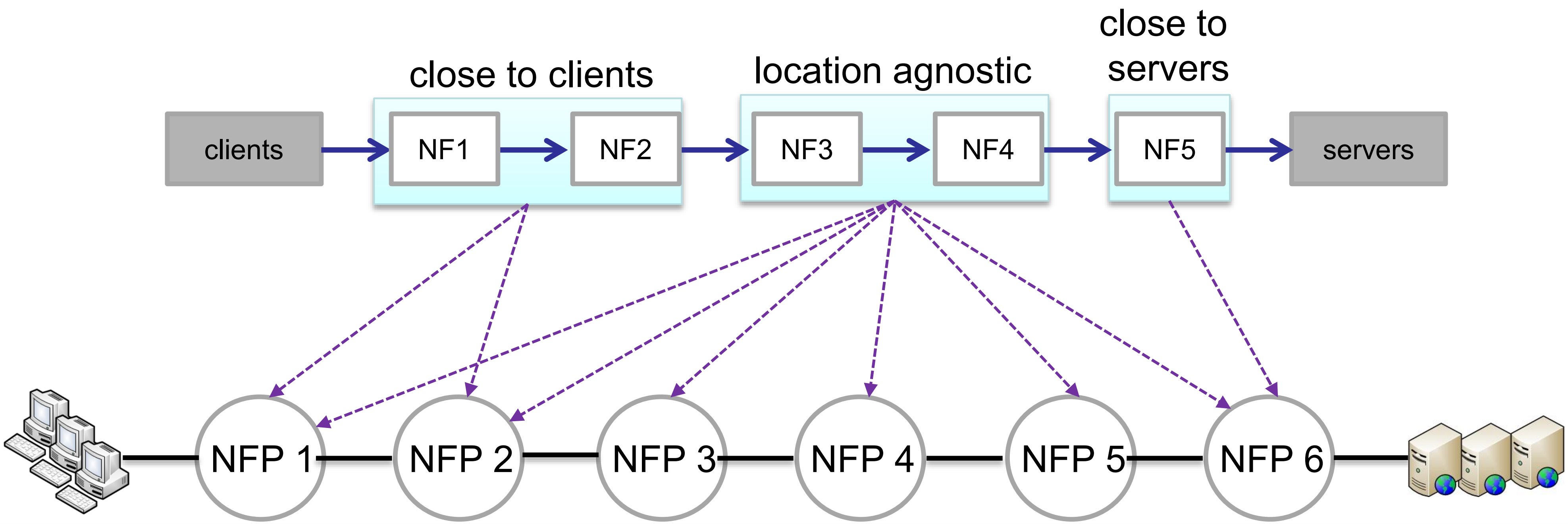
0   8   16   24   31
requester address
service request ID (SRID)
controller address
controller address
⋮
controller address

CONTROLLER message

Middlebox Selection

Service Chain Partitioning

- Objective:
  - Minimize the number of assigned NFPs
- Approach:
  - Service chain partitioning based on NF location dependencies
  - Candidate NFP identification
  - Assignment of chain segments to the candidate NFP with the lowest utilization using Multi-Party Computation (MPC)



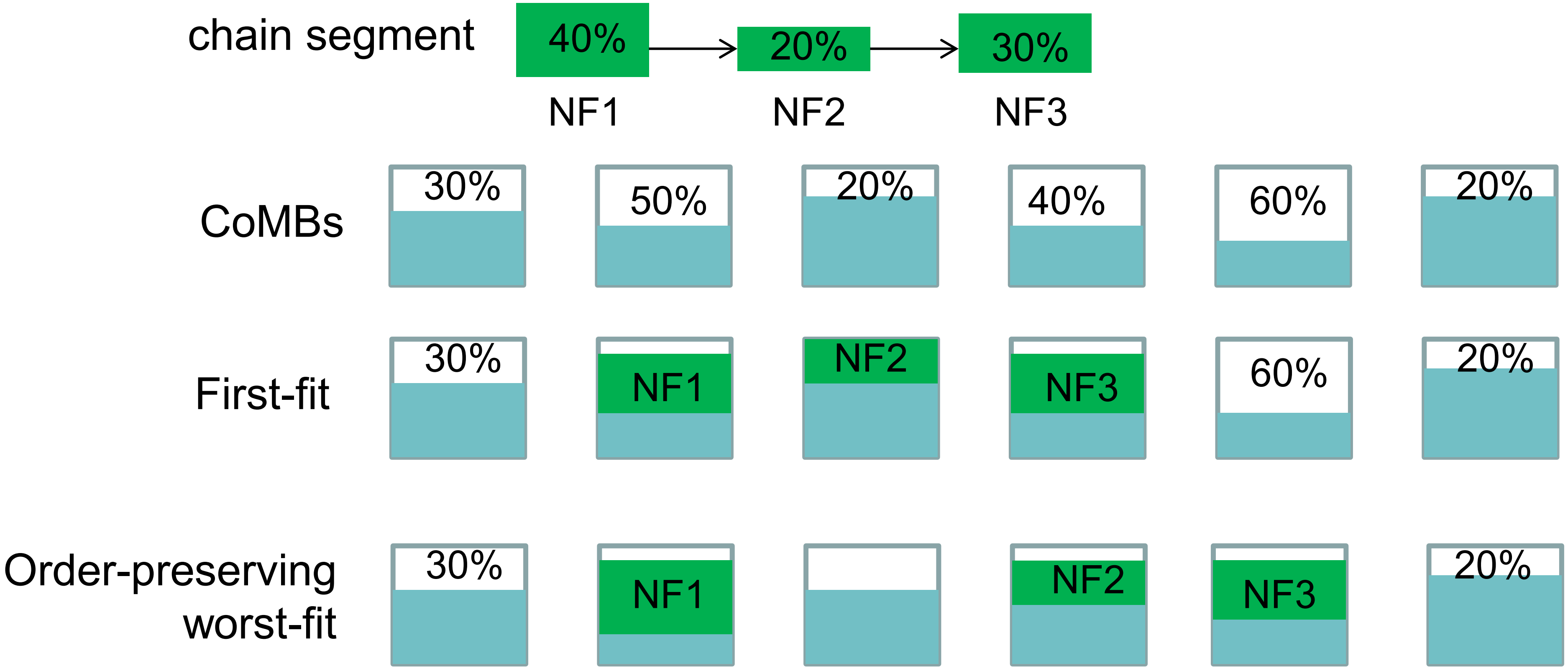
Secure Multi-Party Computation (MPC)

- Cryptographic protocol:
  - Different parties with private inputs to compute a function on their inputs:
    - Input values remain private
    - Result of the computation is correct
    - Cheating parties will not learn information about the honest parties inputs

- Example:
  - Two billionaires want to find who is richer

Intra-Provider Middlebox Selection

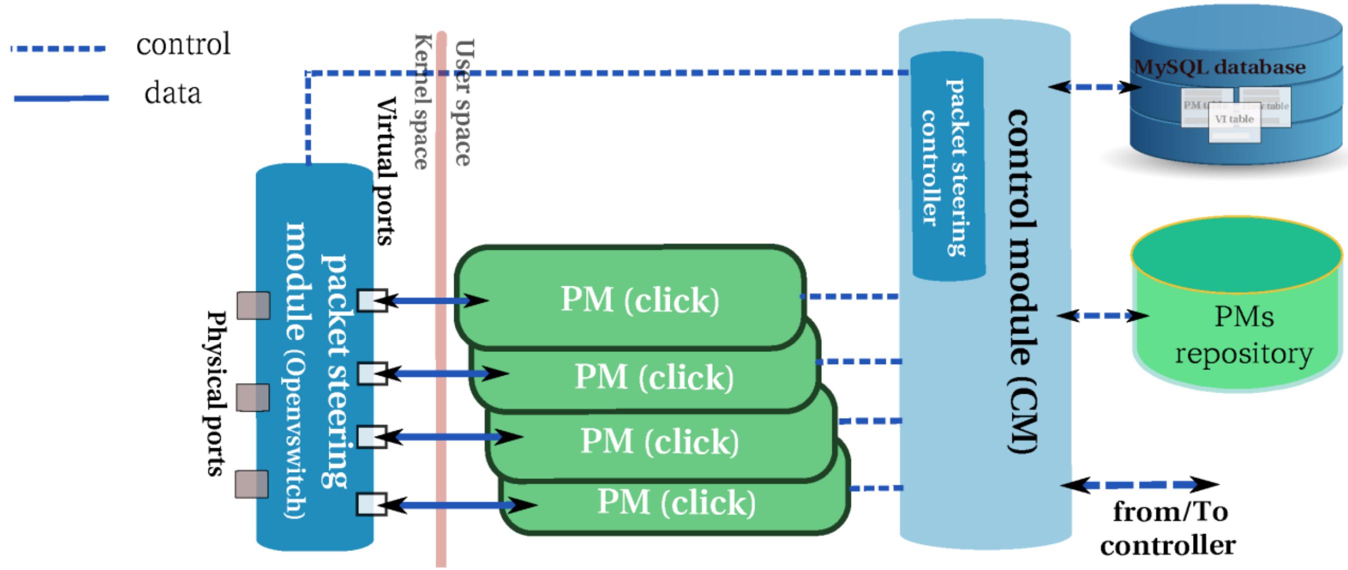
- Objectives:
  - Load balancing
  - Correctness
- Approach:
  - Step 1: First-fit assignment
  - Step 2: Order-preserving worst-fit



Implementation

CoMB Implementation

- Processing module (PM):
  - Implements NFs using Click Modular Router
- Packet steering module:
  - Steers traffic between PMs and physical ports using OpenvSwitch
- Control module:
  - Installs, configures, and terminates PMs
  - API exposed to controller
- Repository:
  - Stores PM configuration templates

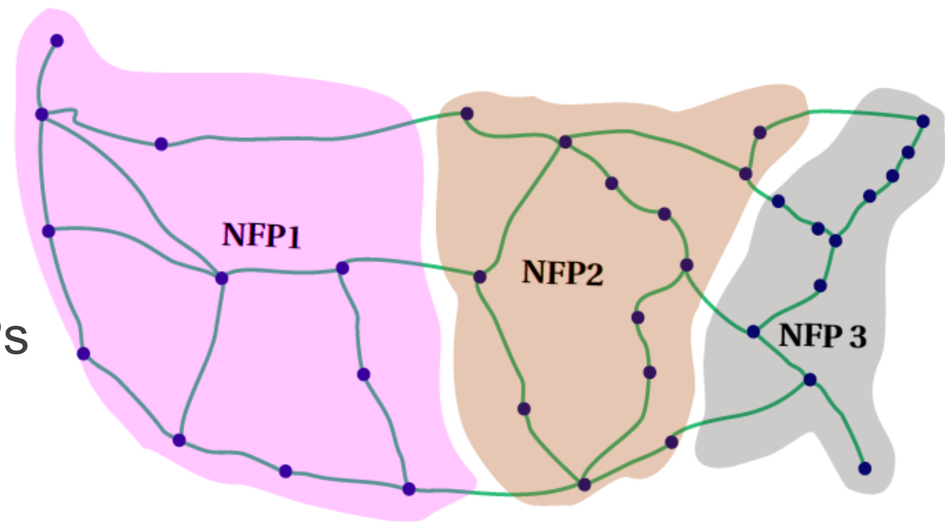


Evaluation

Evaluation Environment

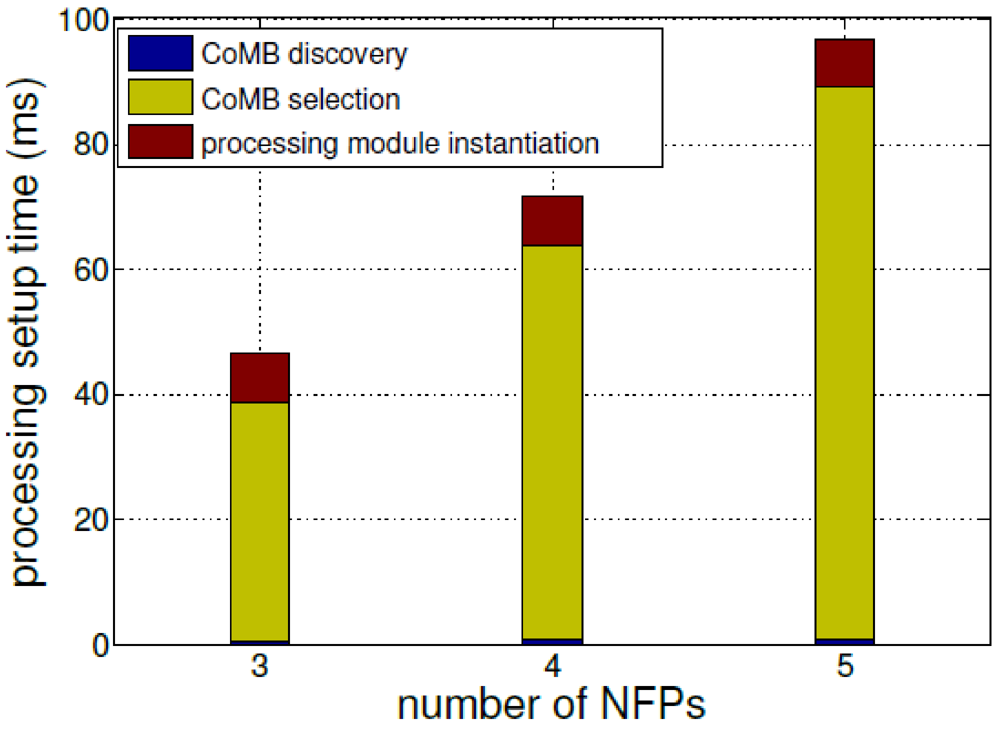
- Experimental evaluation of flow processing setup delay:
  - 22 servers deployed in an Emulab-based testbed (FILAB):
    - quad-core Xeon CPUs @2.27GHz and 6 GB DDR3
  - 2 - 5 NFPs, each with:
    - 1 controller
    - 3 CoMBs (deployed in separate nodes)

- Evaluation of ComB selection efficiency with simulations:
  - Simulator:
    - Flow-level simulator (Python)
  - Simulation setup:
    - Internet-2 topology
    - 34 CoMBs subdivided into 3 NFPs

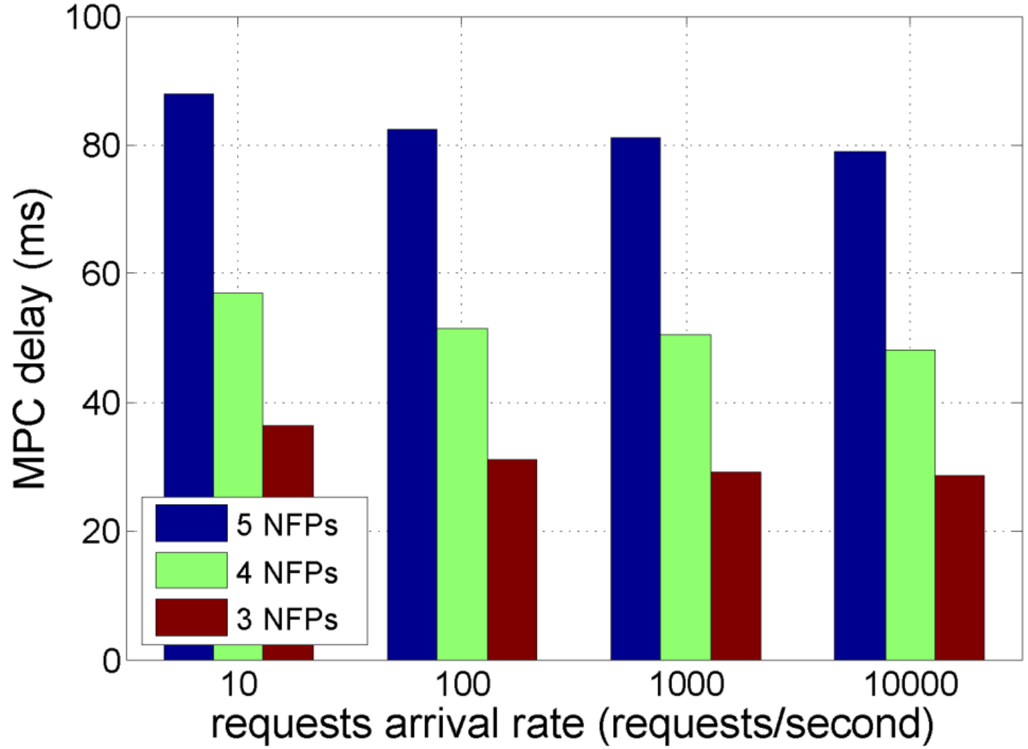


Flow Processing Setup

- MPC dominates flow processing setup delay
- MPC delay < 100 ms for up to 5 NFPs (i.e., average AS-path length)



processing delay per request (10 requests/sec)

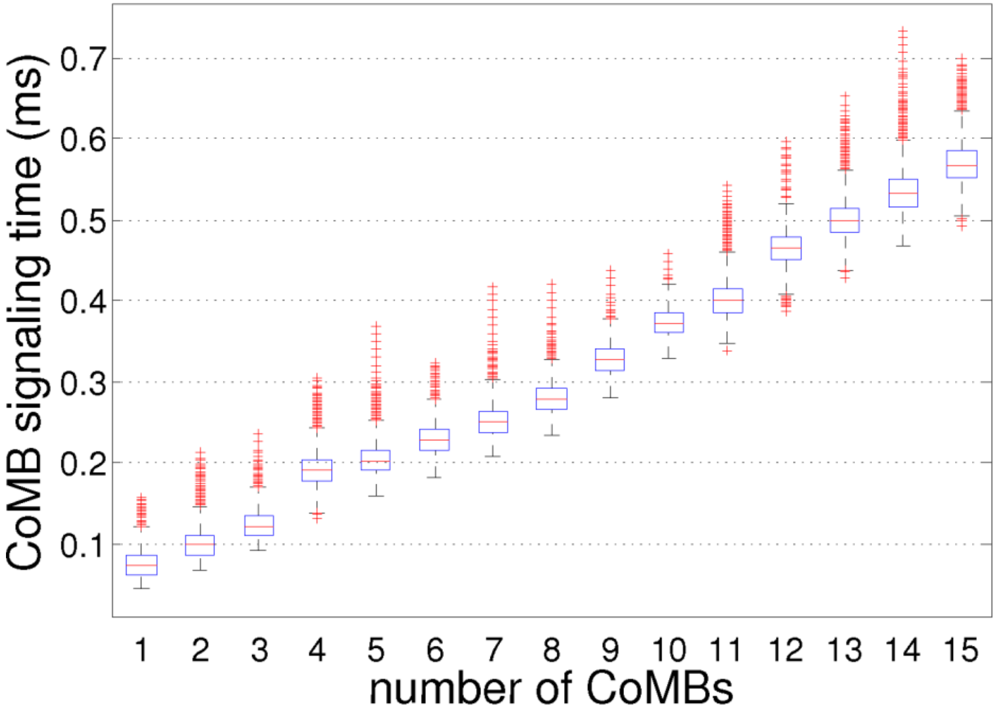


MPC delay per request

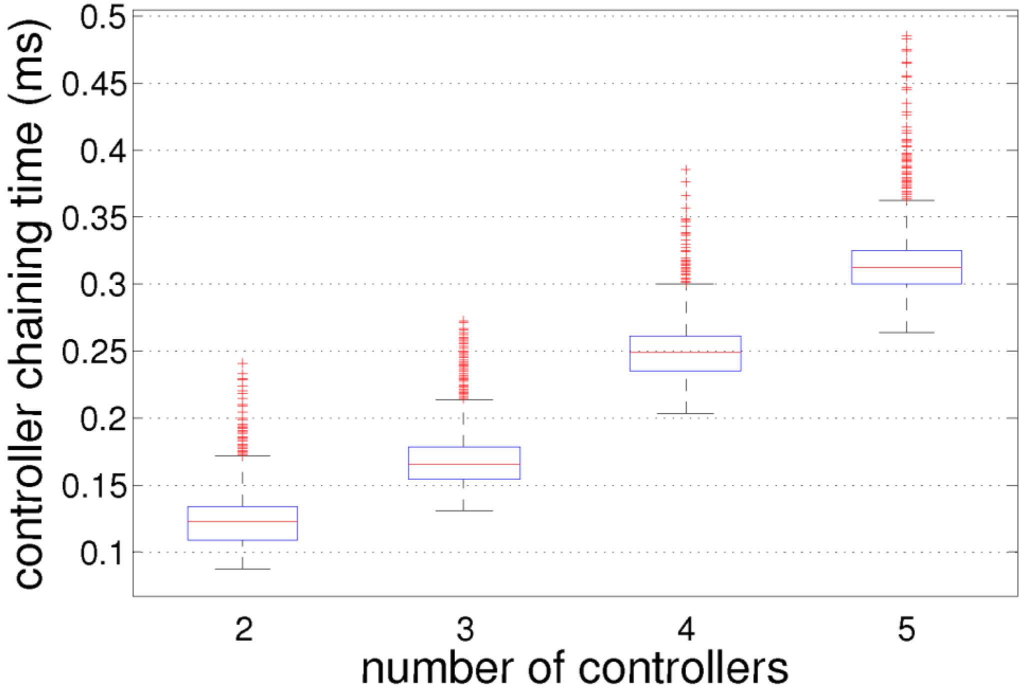
Middlebox Discovery

Institut für Kommunikations-Technik

- Minimal delay with CoMB signaling and controller chaining
- Middlebox discovery scales with the number of CoMBs



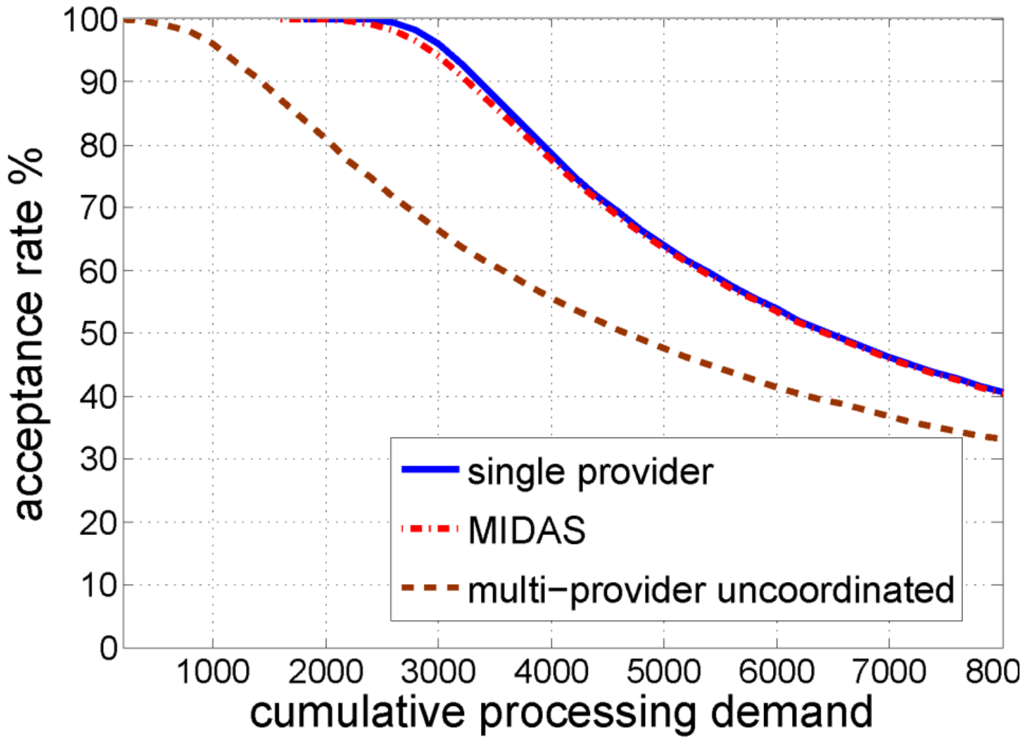
CoMB signaling delay per request

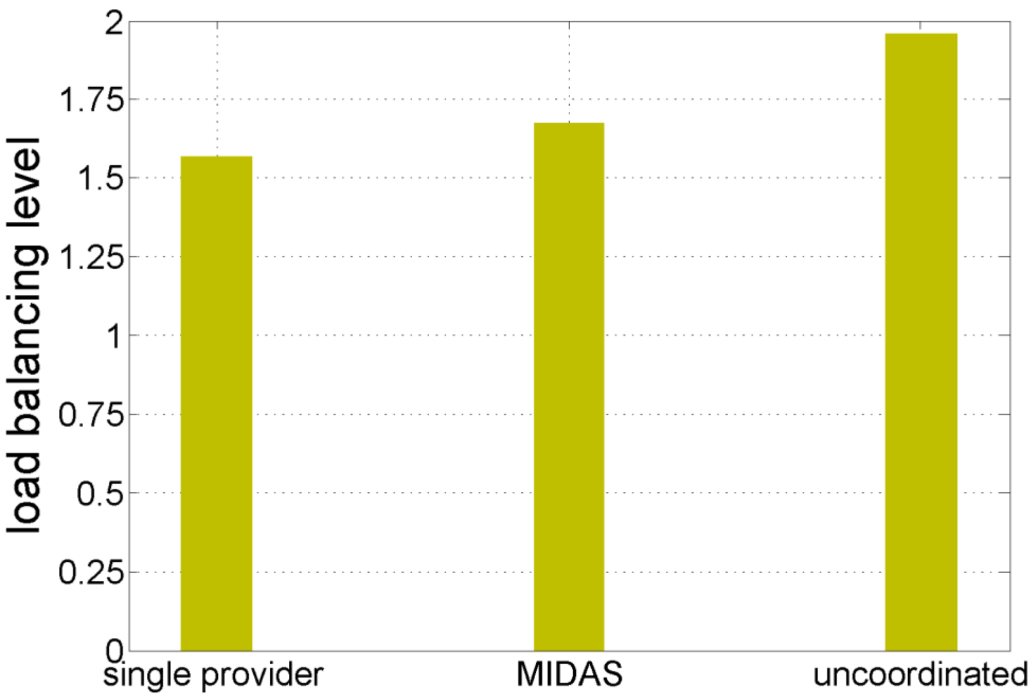


controller chaining delay per request

Simulation Results

- Comparison method:
  - Single provider:
    - All CoMBs managed by a single controller
  - Multi-provider uncoordinated:
    - On-the-fly selection of CoMBs based on the utilization level





Conclusions

Conclusions

- MIDAS enables:
  - Middlebox discovery without prior knowledge of the traffic path
  - Interoperability among NFPs for middlebox selection
  - Rapid and order-preserving network service embedding
    - Feasibility of coordinated on-path processing setup

Thank you!

Panagiotis Papadimitriou

E-mail: panagiotis.papadimitriou@ikt.uni-hannover.de
WWW: http://www.ikt.uni-hannover.de/

A. Abujoda and P. Papadimitriou, **MIDAS: Middlebox Discovery and Selection for On-Path Flow Processing**, IEEE COMSNETS 2015