NFaaS: Named Functions as a Service

Michał Król <m.krol@ucl.ac.uk>, Ioannis Psaras <i.psaras@ucl.ac.uk>

University College London

Outline

Function Execution in ICN

Background

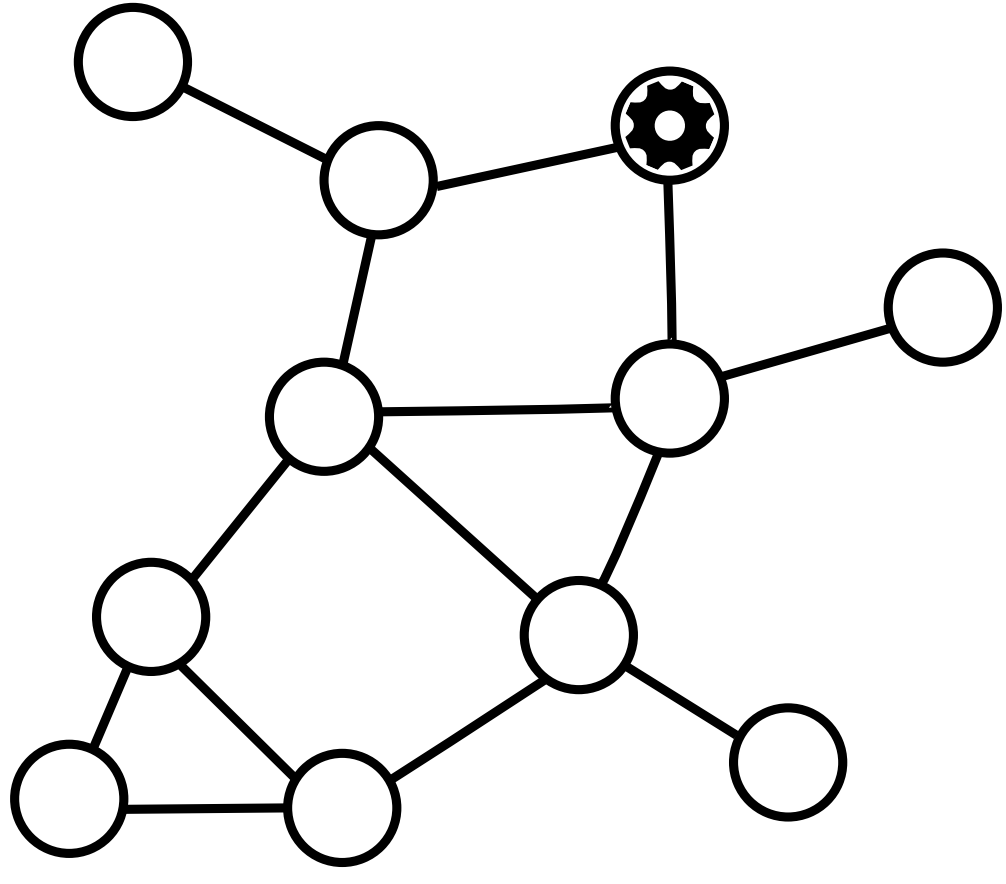
System Overview

Results

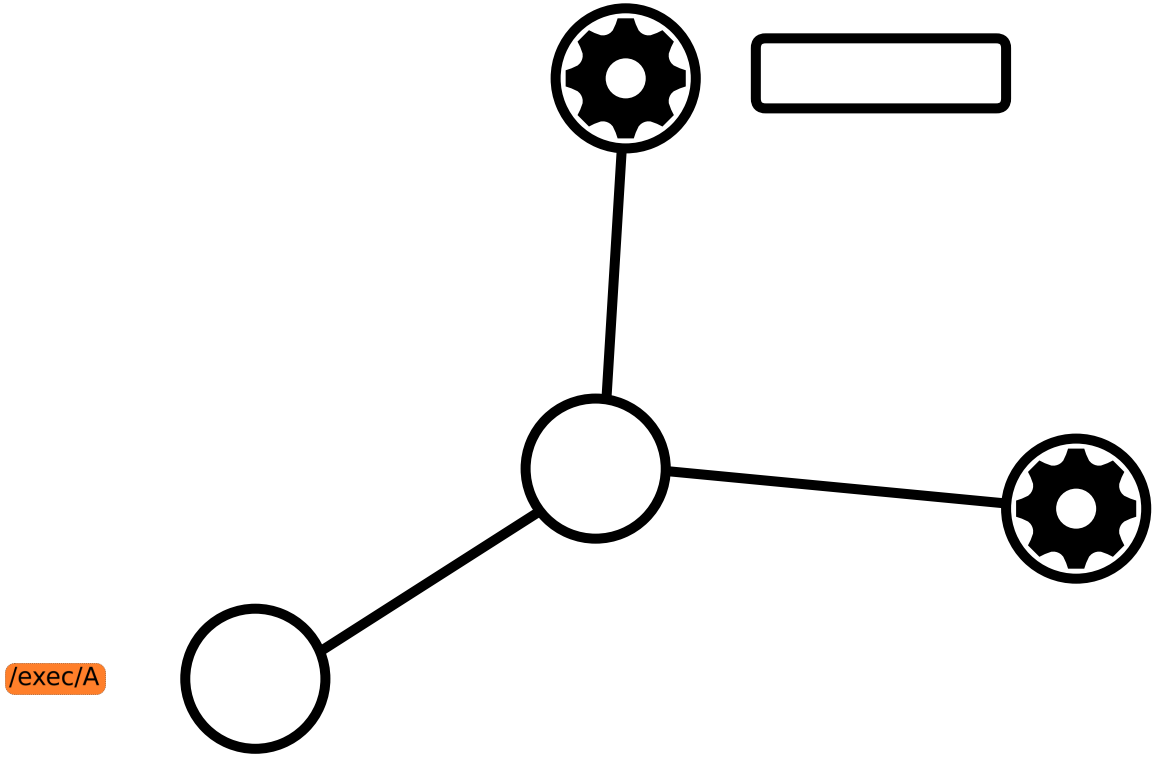
Function Execution in ICN

- ICN is well suited for function execution
- Similar to data retrieval
- Any node can execute a function
- Low power devices request computation
- Many use cases (VR, IoT, etc.)

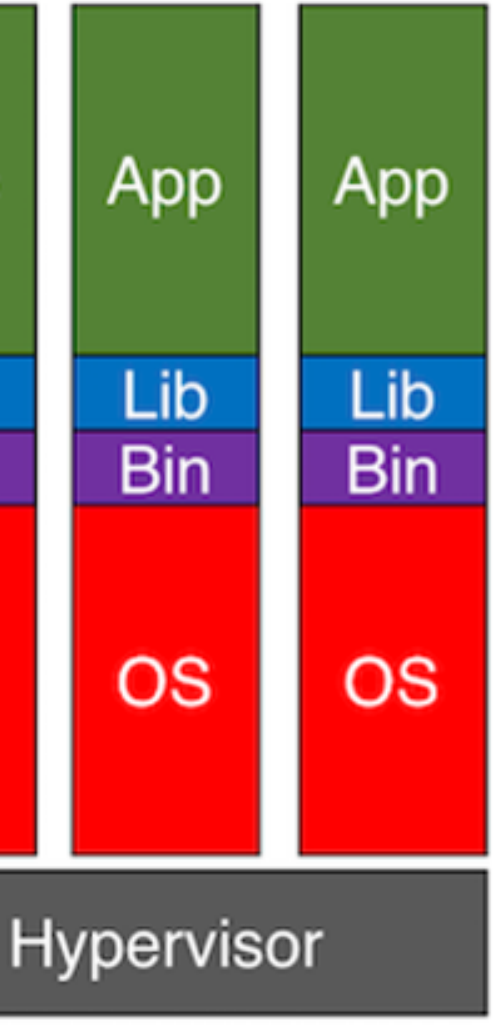
Function Execution



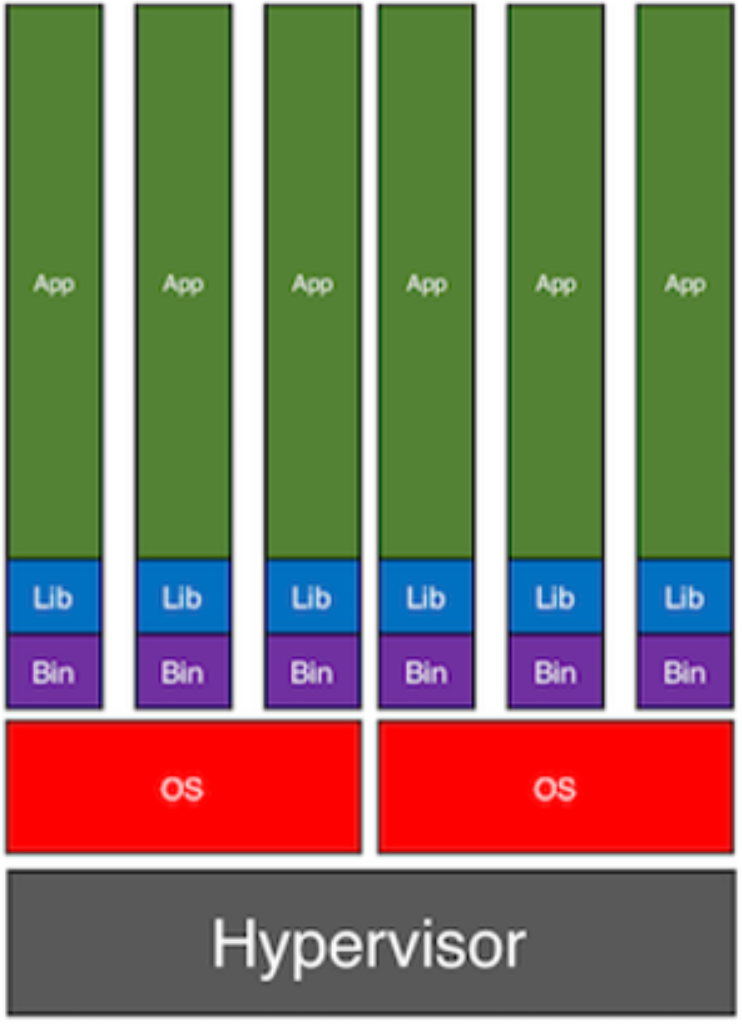
Function Execution



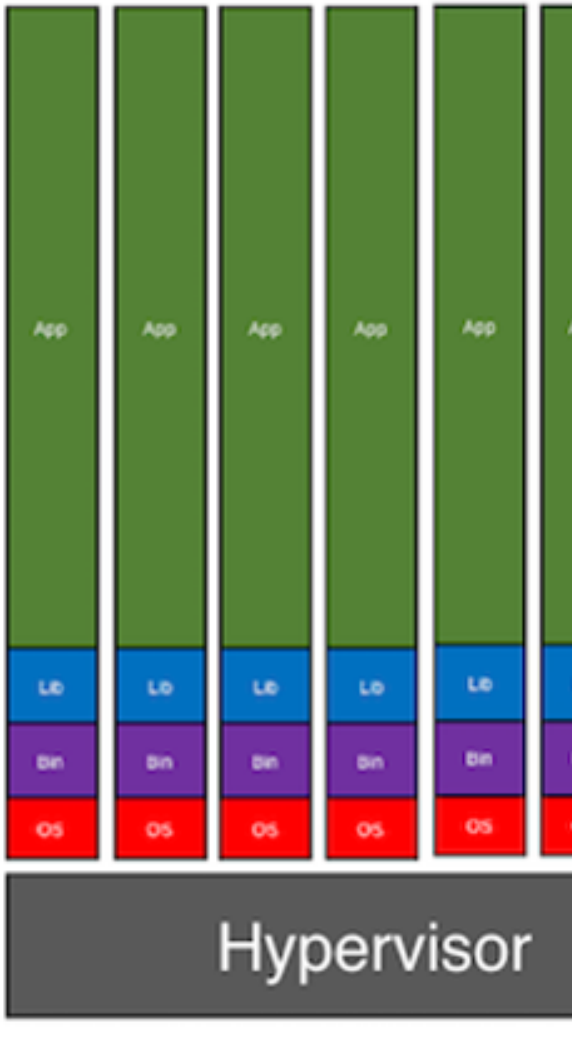
Background



tual Machines



Containers

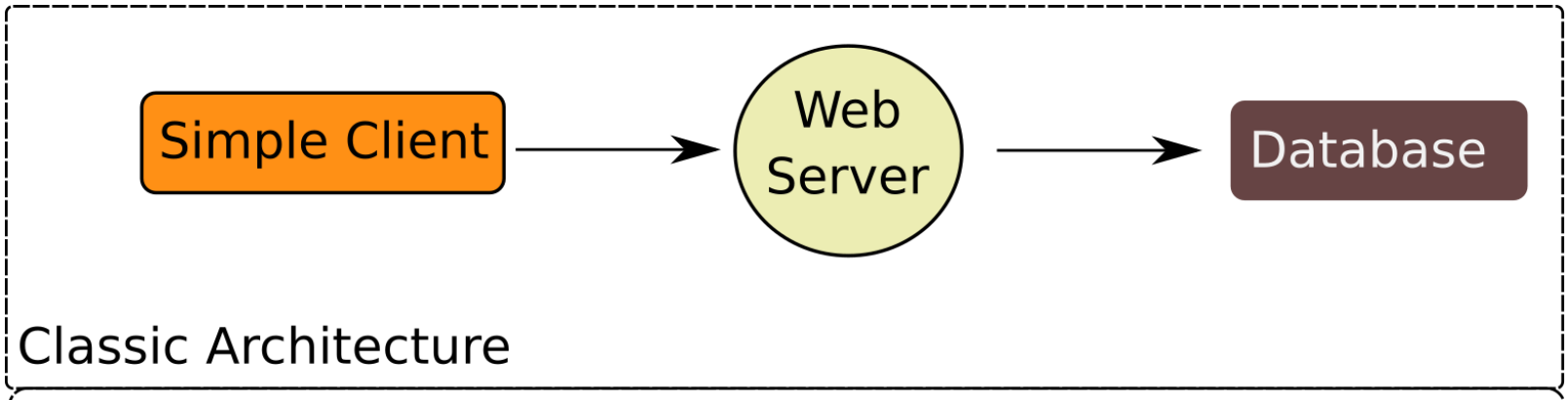


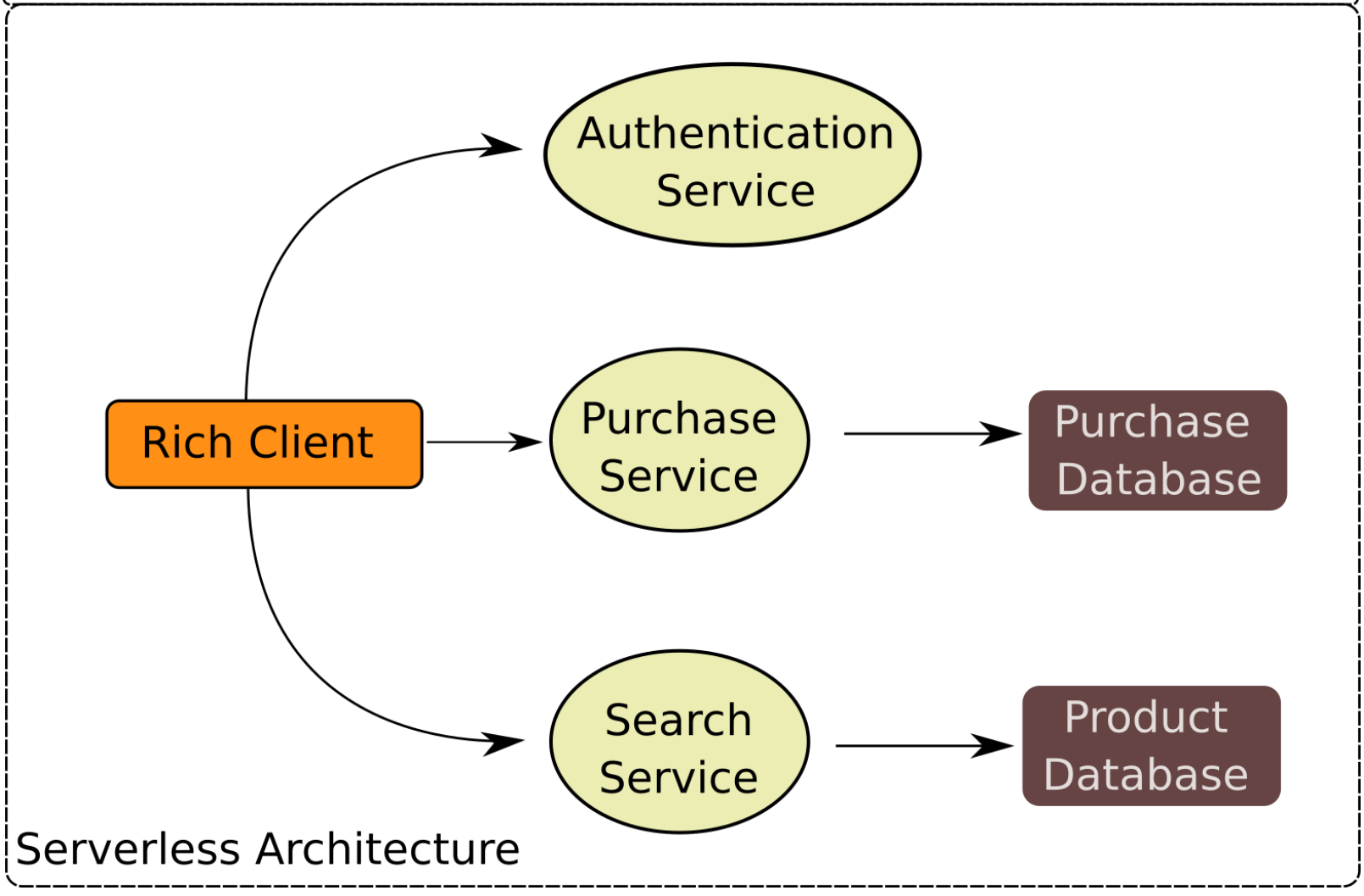
Unikernels

//flic.kr/p/4xv4t

Unikernels

- Immutable image
- Run on a hypervisor or bare metal
- Small attack surface
- Small size
- Highly optimised
- Fast to boot


Simple Client
Web Server
Database
Classic Architecture

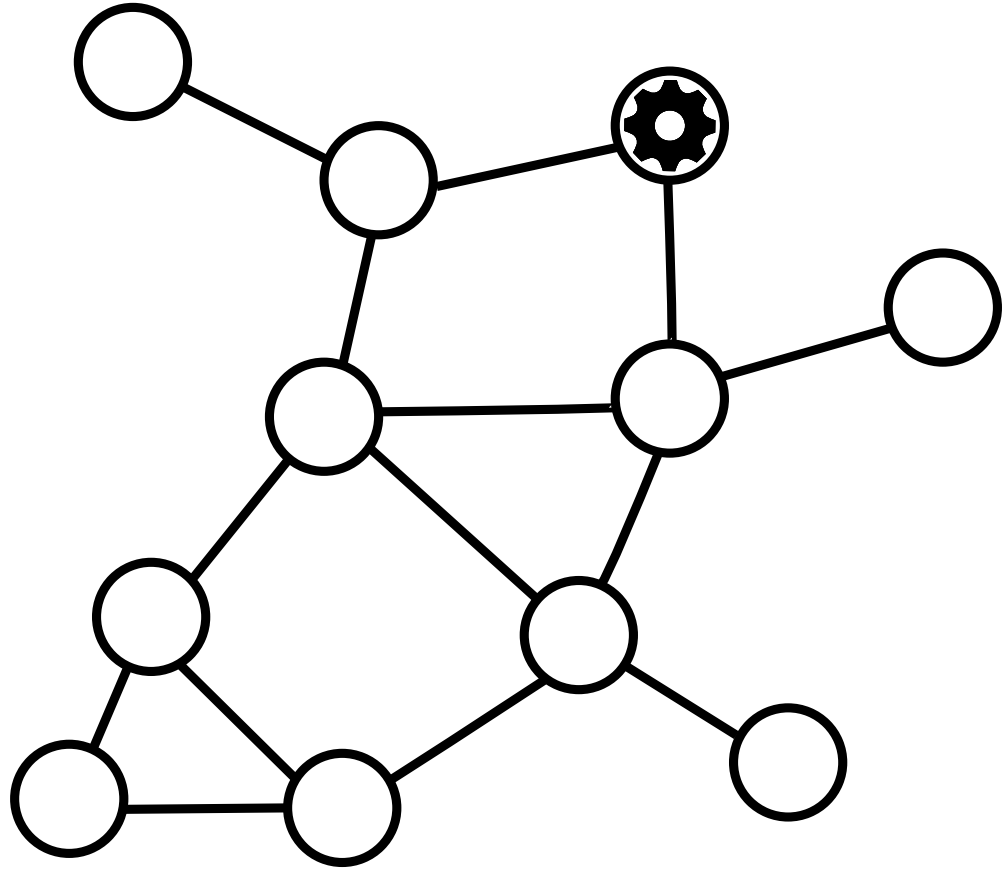

Authentication Service
Rich Client
Purchase Service
Purchase Database
Search Service
Product Database
Serverless Architecture

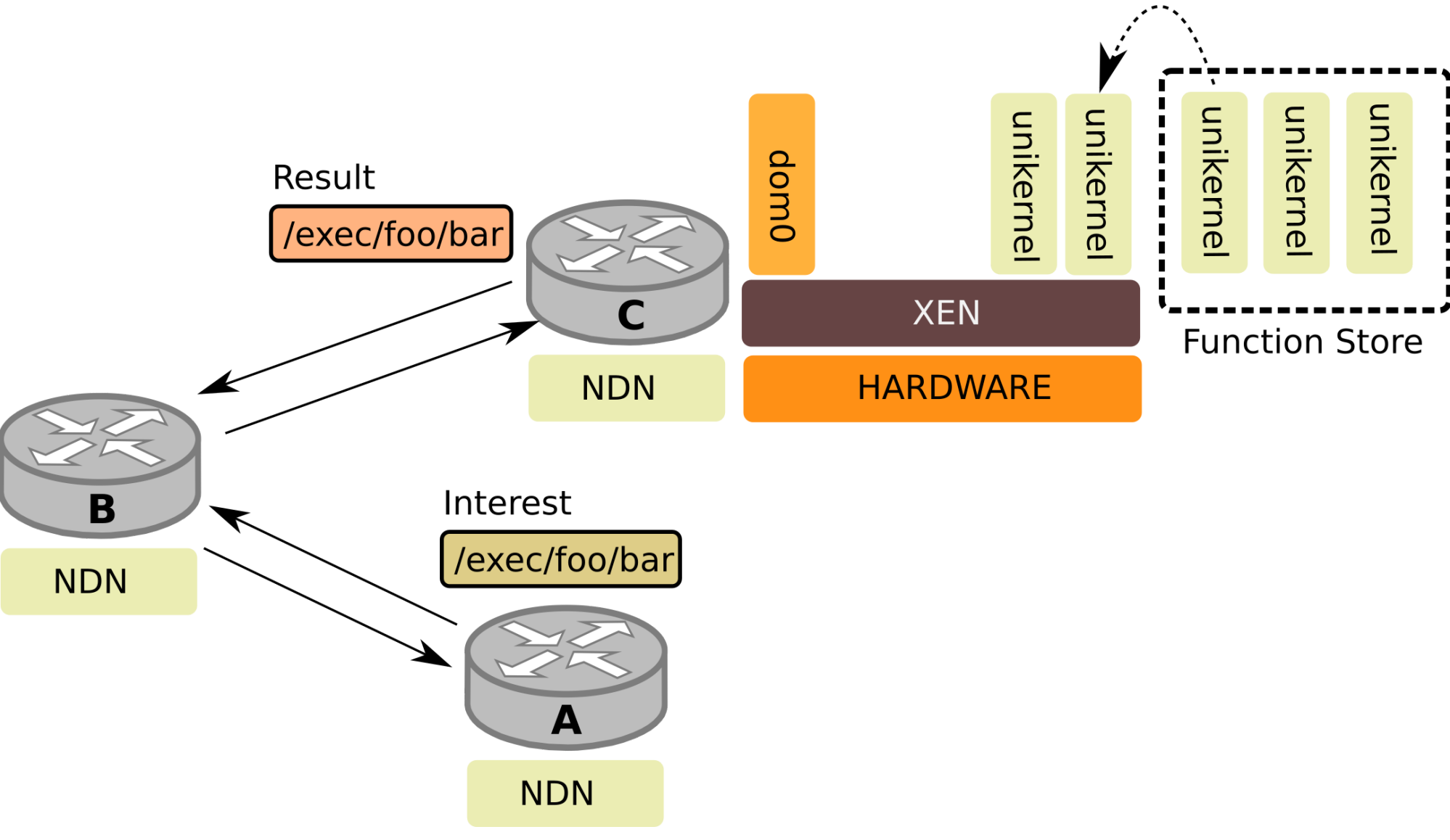
Serverless

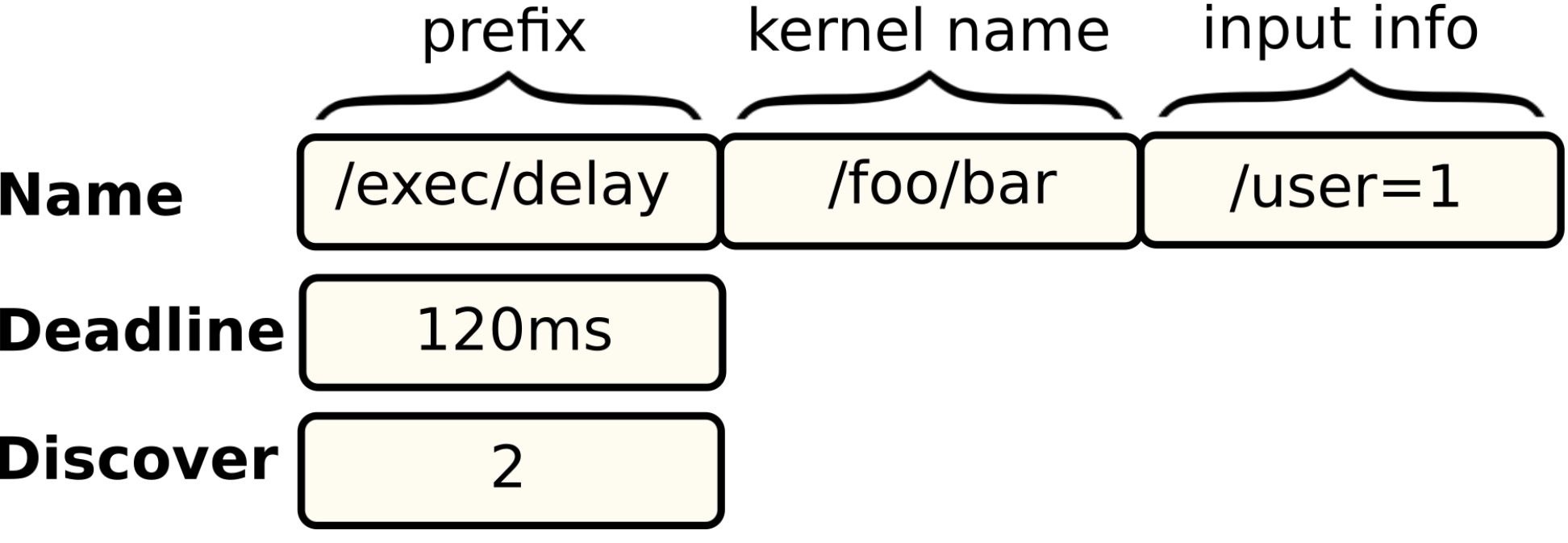
- Service decomposition into small functions
- Application logic on client
- Stateless
- State can be controlled by tokens or external data source
- Easy deployment in the cloud
- Easy "pay as you go"

System Overview

Named Function as a Service




Result
/exec/foo/bar
C
NDN
dom0
unikernel
unikernel
XEN
HARDWARE
unikernel
unikernel
unikernel
Function Store
B
NDN
Interest
/exec/foo/bar
A
NDN


prefix
kernel name
input info
Name
/exec/delay
/foo/bar
/user=1
Deadline
120ms
Discover
2

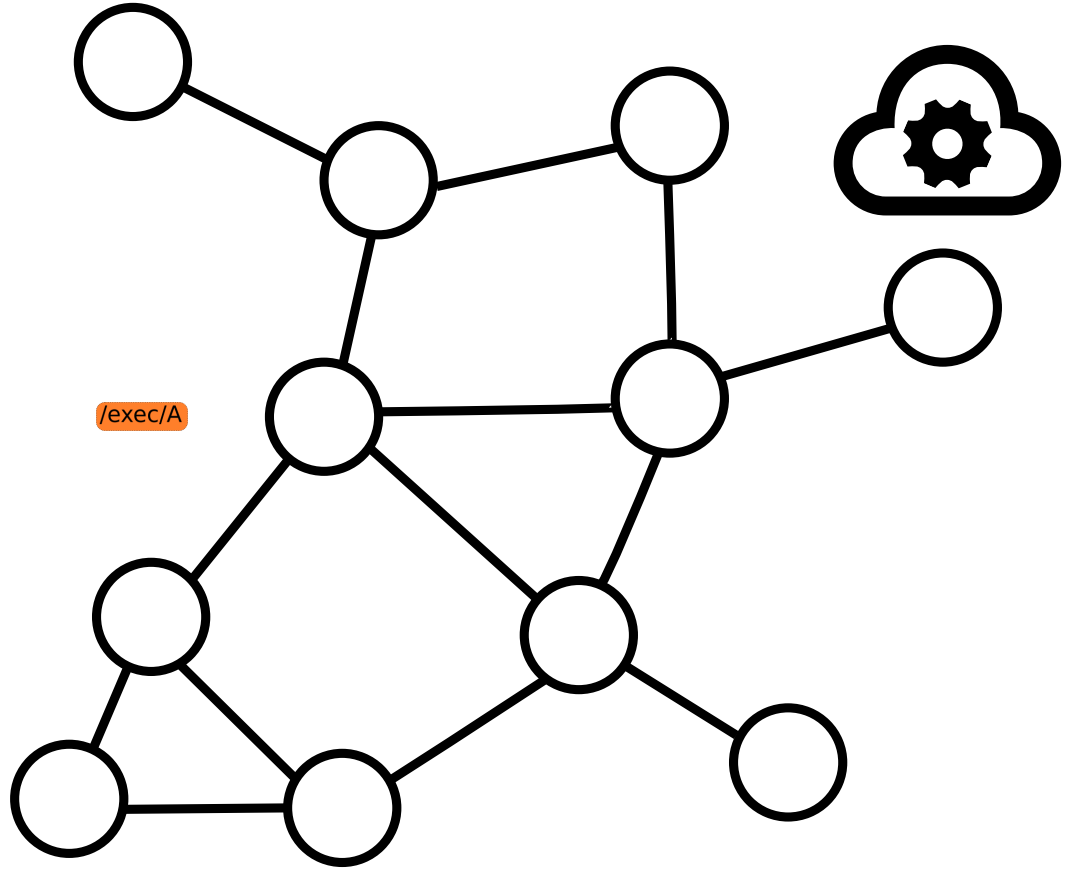
Score Function

- Selects recently the most popular images to download

$$score = \sum_{i=1}^{m} \frac{p_i}{n} * (m - i) + \ldots$$

Function	Popularity		Score
A	2/10	5/10	9
B	5/10	2/10	12
C	1/10	3/10	5
	recent	older	

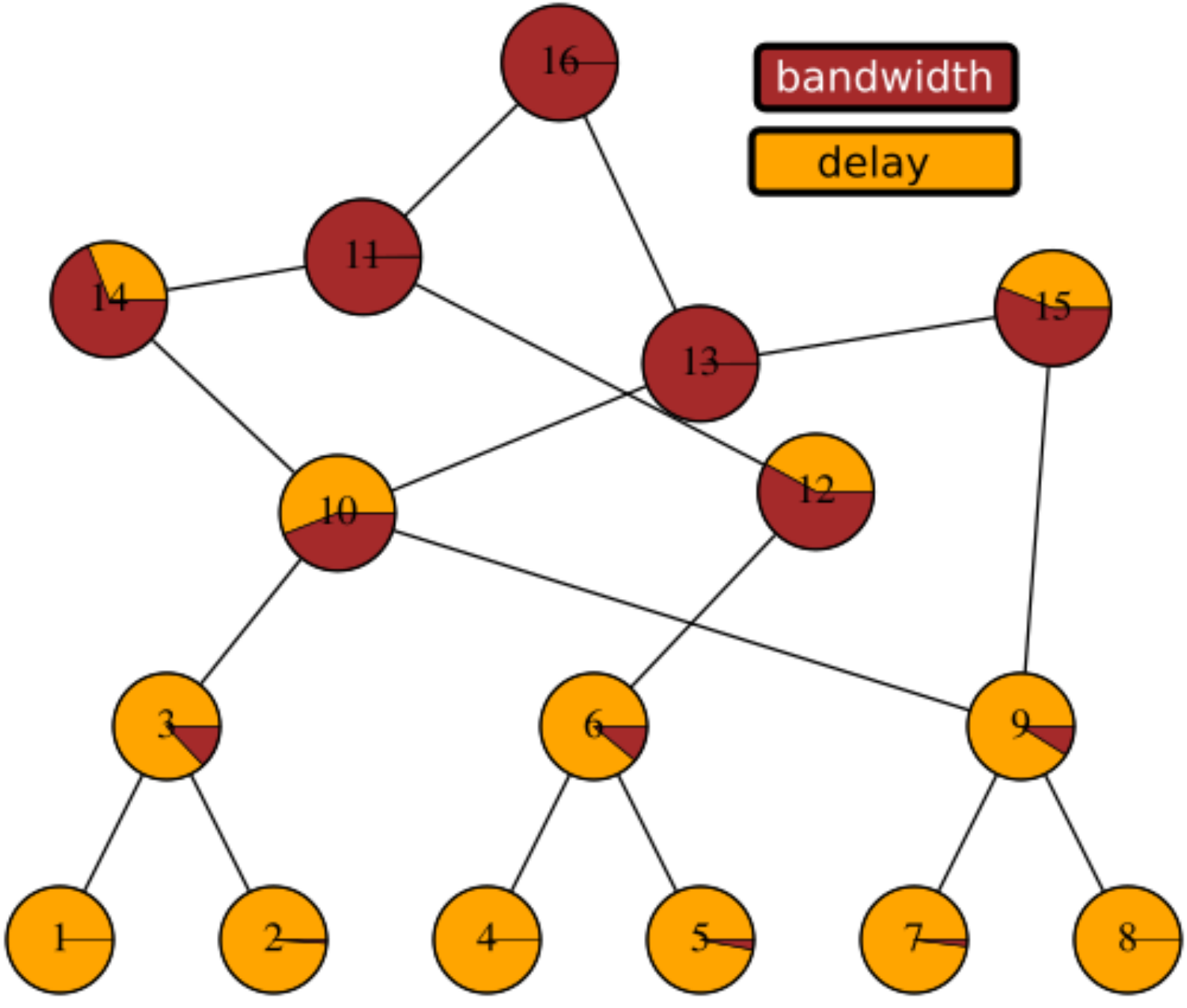
Function Execution

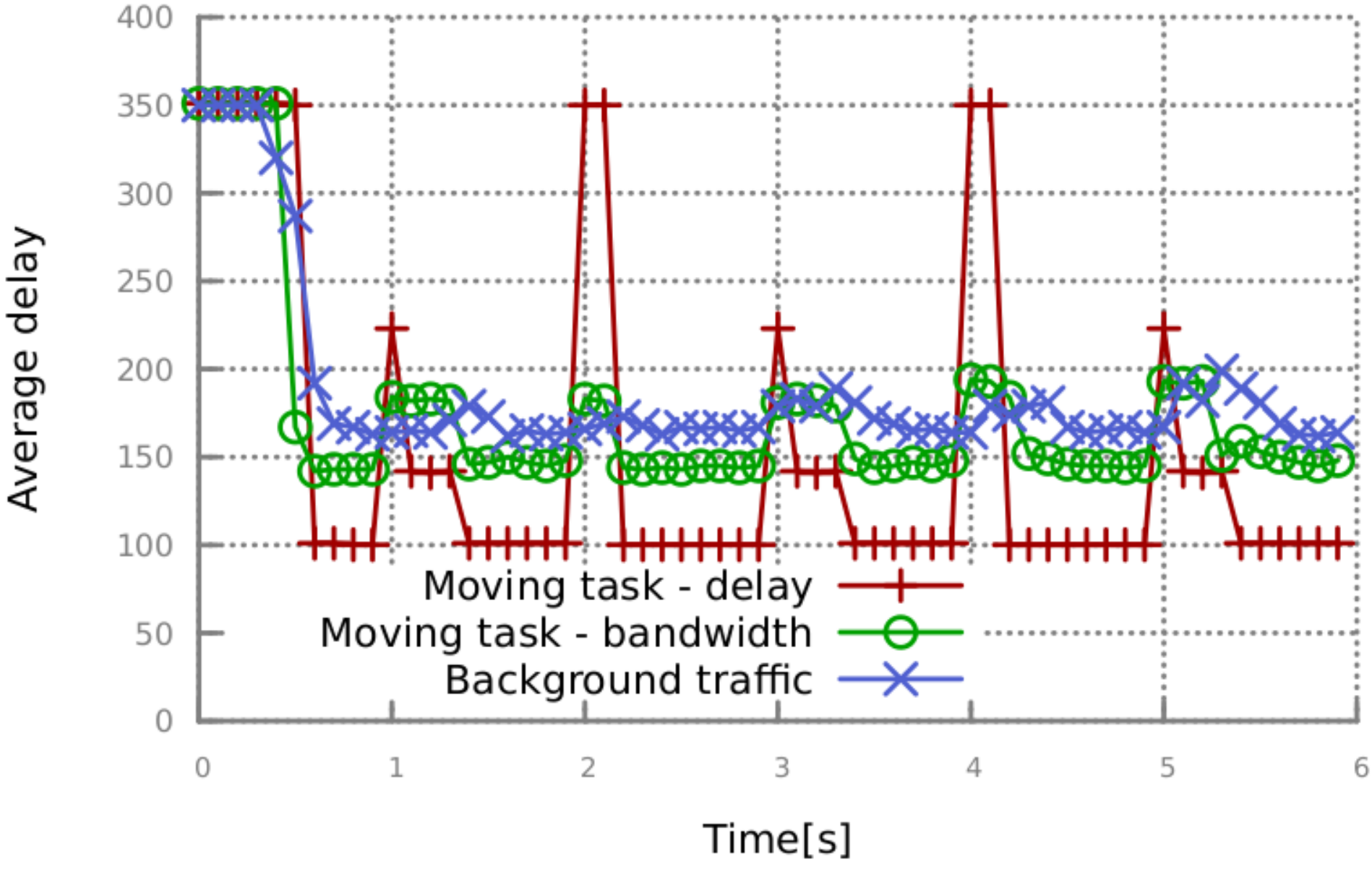


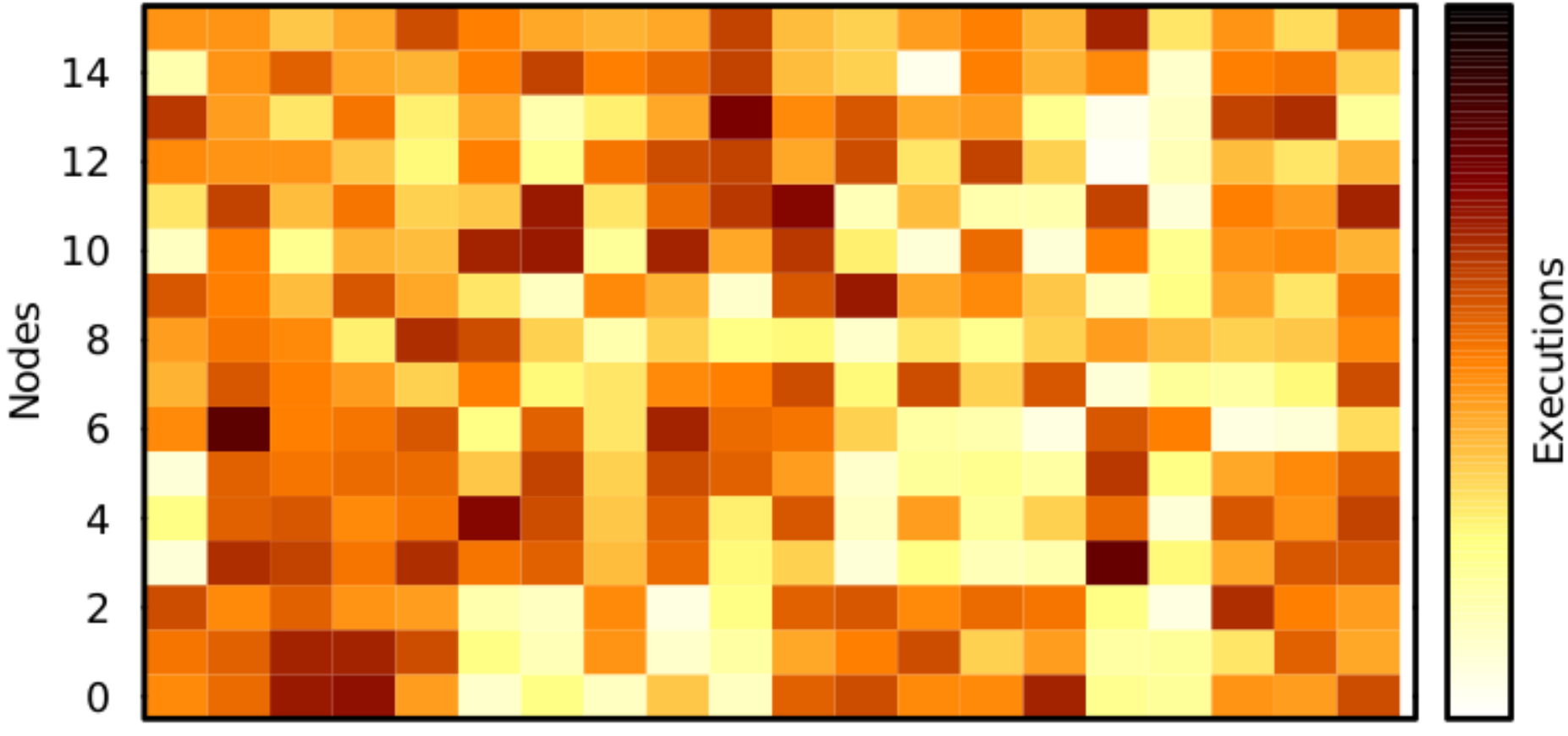
NFaaS

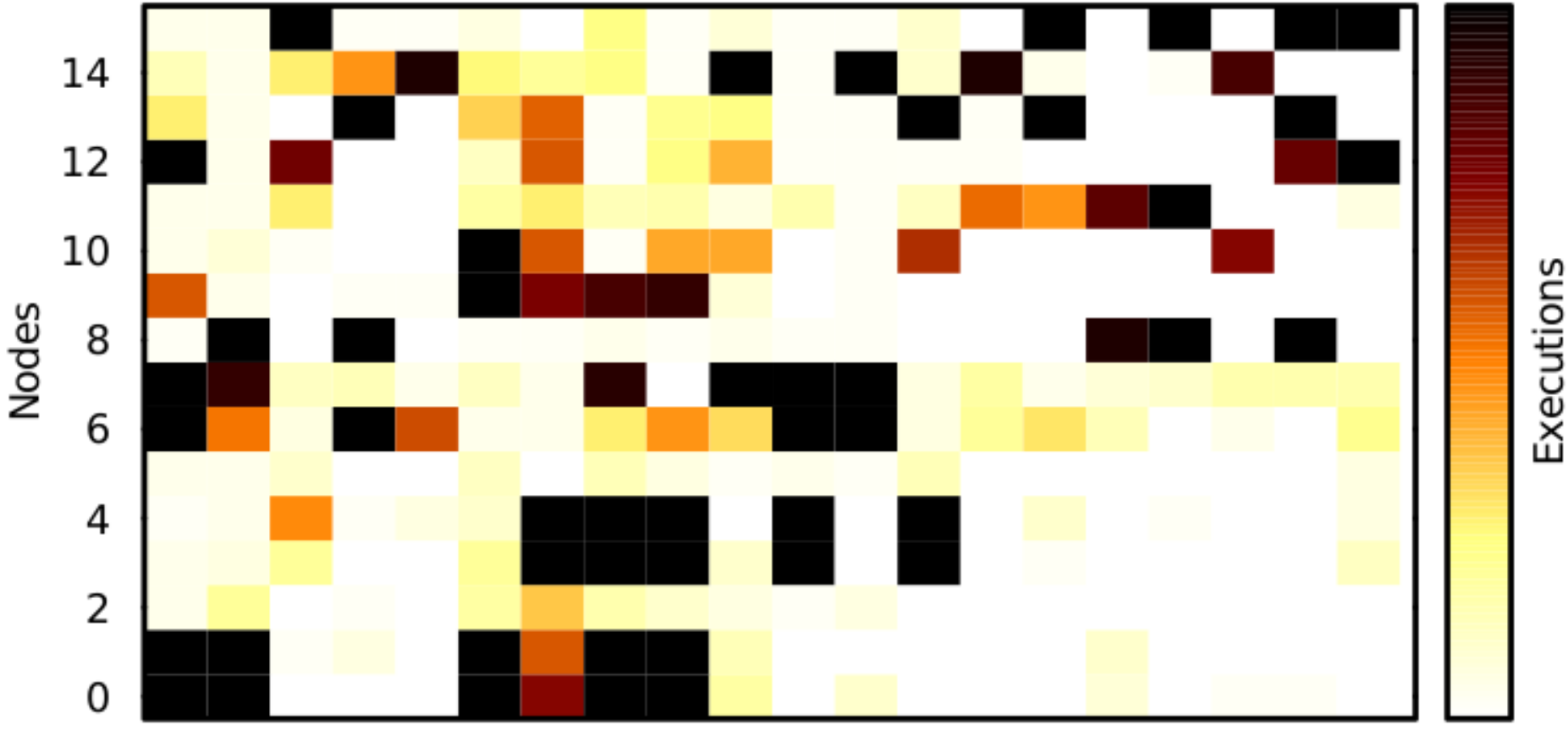
- Completely distributed
- Moving function where they're needed
- Functions as stateless unikernels
- Nodes run popularity contest
- Different forwarding strategies

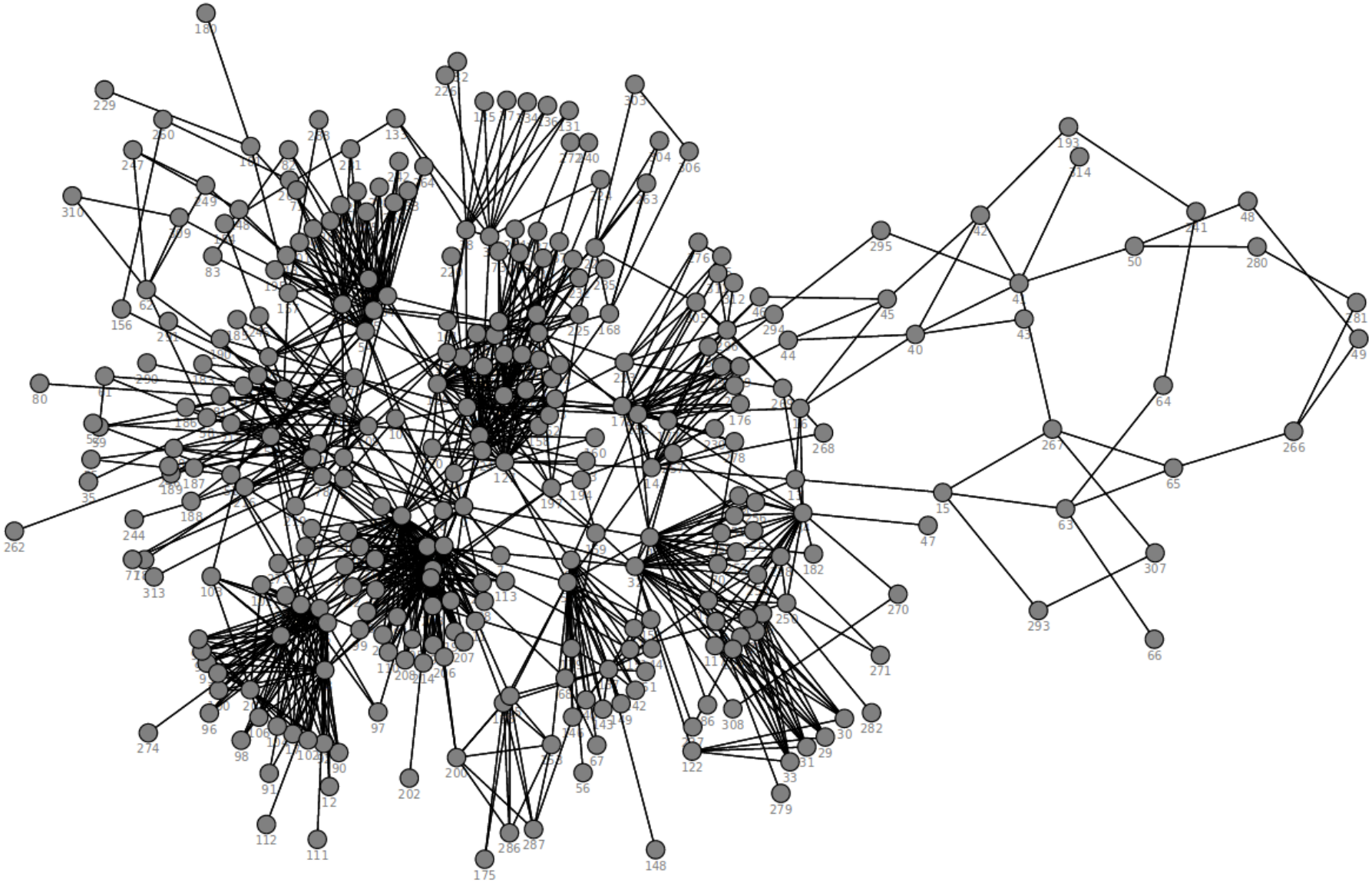
Results

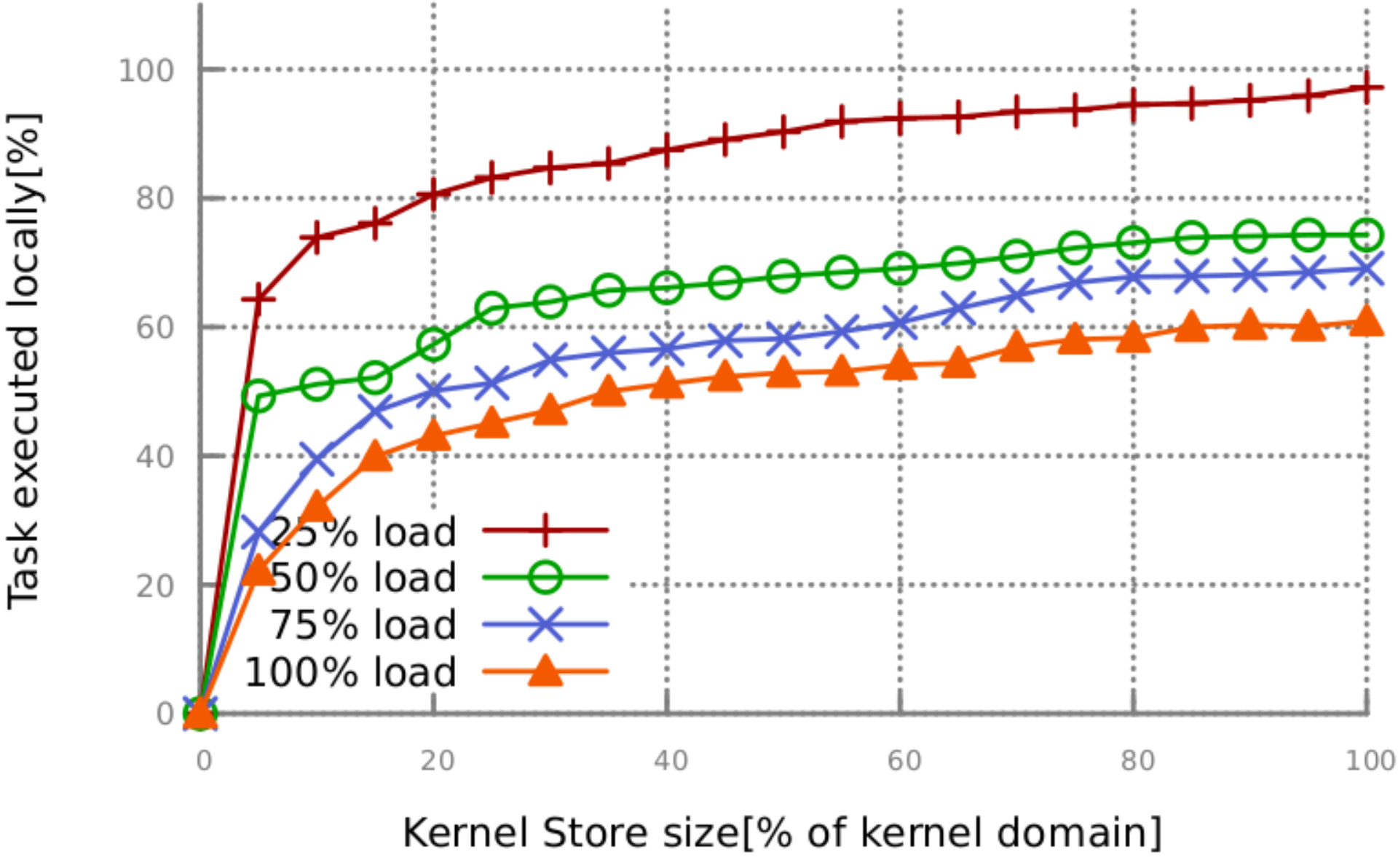

bandwidth
delay
16
11
14
13
15
12
10
3
6
9
1
2
4
5
7
8

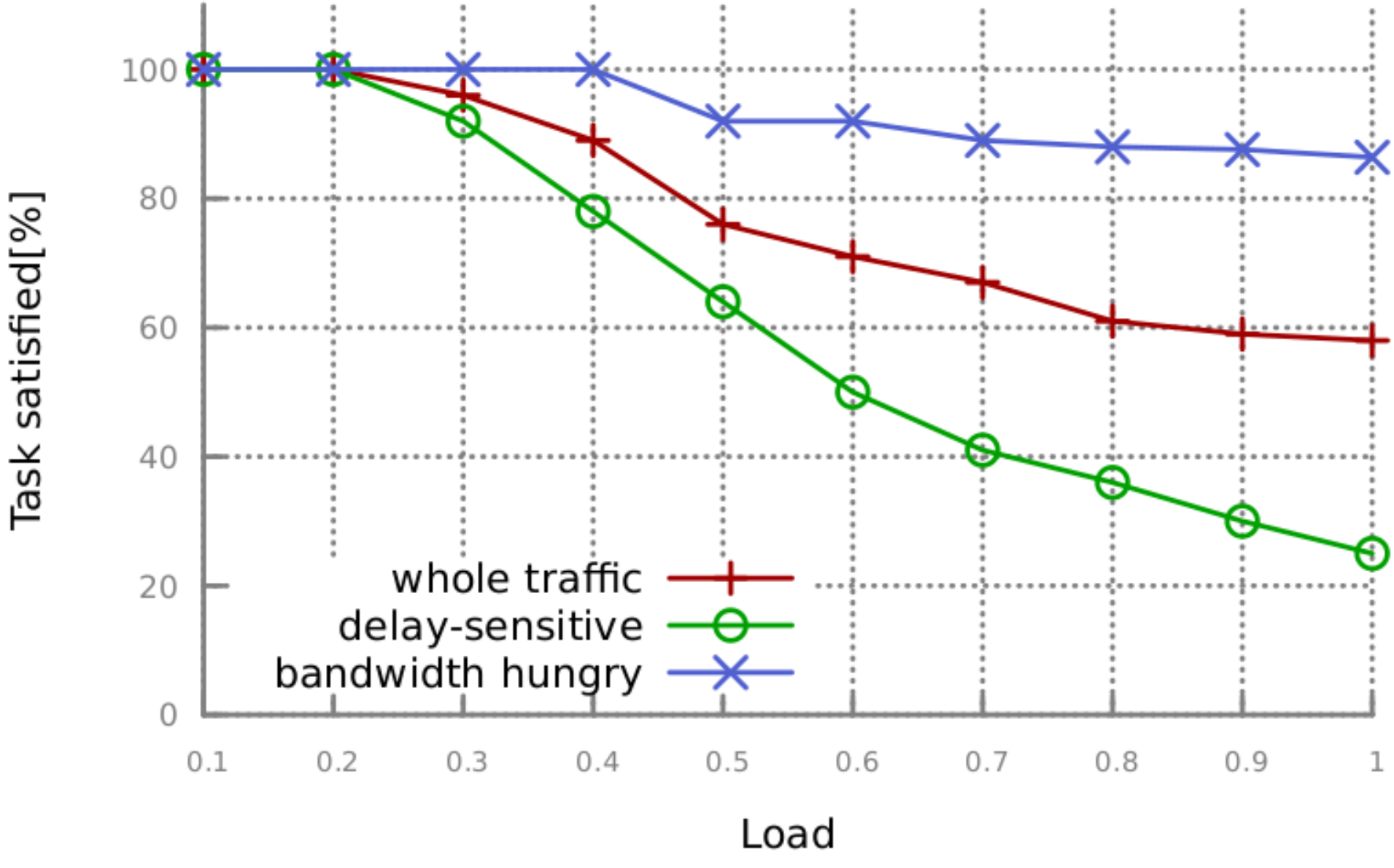

Average delay
400
350
300
250
200
150
100
50
0
0
1
2
3
4
5
6
Time[s]
Moving task - delay
Moving task - bandwidth
Background traffic


Nodes
14
12
10
8
6
4
2
0
Executions


Nodes
14
12
10
8
6
4
2
0
Executions




Task executed locally[%]
100
80
60
40
20
0
0
20
40
60
80
100
Kernel Store size[% of kernel domain]
25% load
50% load
75% load
100% load


Task satisfied[%]
100
80
60
40
20
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Load
whole traffic
delay-sensitive
bandwidth hungry

Prototype

- Rumprun unikernels
- size ~19MB
- 512GB HD = 26947 images
- boot time < 100ms

Conclusion

- System for distributed function execution
- Unmanaged Environment
- Simulations in ns-3
- Correct placement of images
- Nodes quickly addapt to network dynamics
- Lower performance in large topologies

Future Work

- Security
- Score Function
- Queueing mechanisms
- Different types of tasks
- Payments and incentives
- ICNRG Discussion - “Adapting ICN to Function Execution for Edge Computing”

Thank you