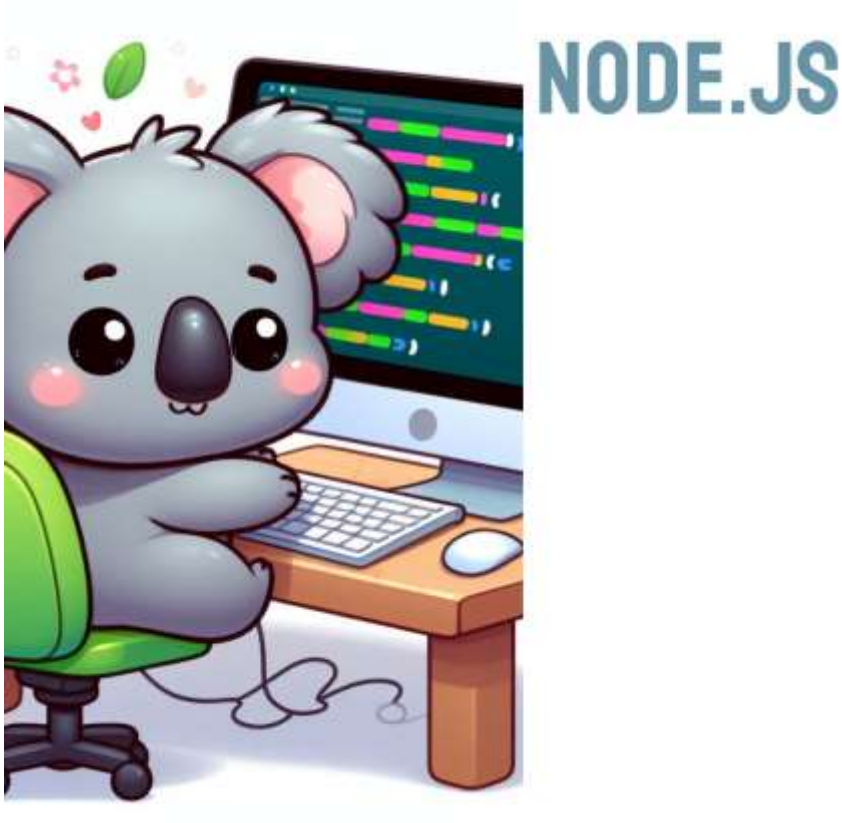



Fast Userspace Networking for the Rest of Us

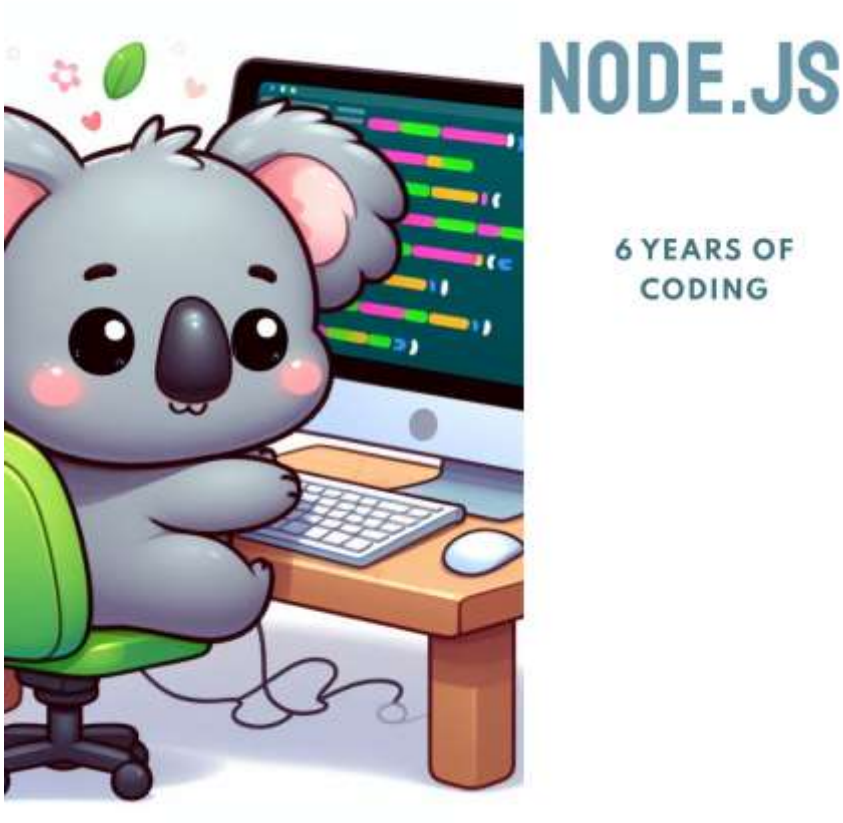
Alireza Sanaee (Huawei, QMUL, Uni of Cambridge)
Gianni Antichi, Farbod Shahinfar(QMUL, Polimi)
Vahab Jabrayilov, Kostis Kaffess (Columbia University)
Anuj Kalia (OpenAI)
Ilias Marinos (NVIDIA)

Meet our Koala!

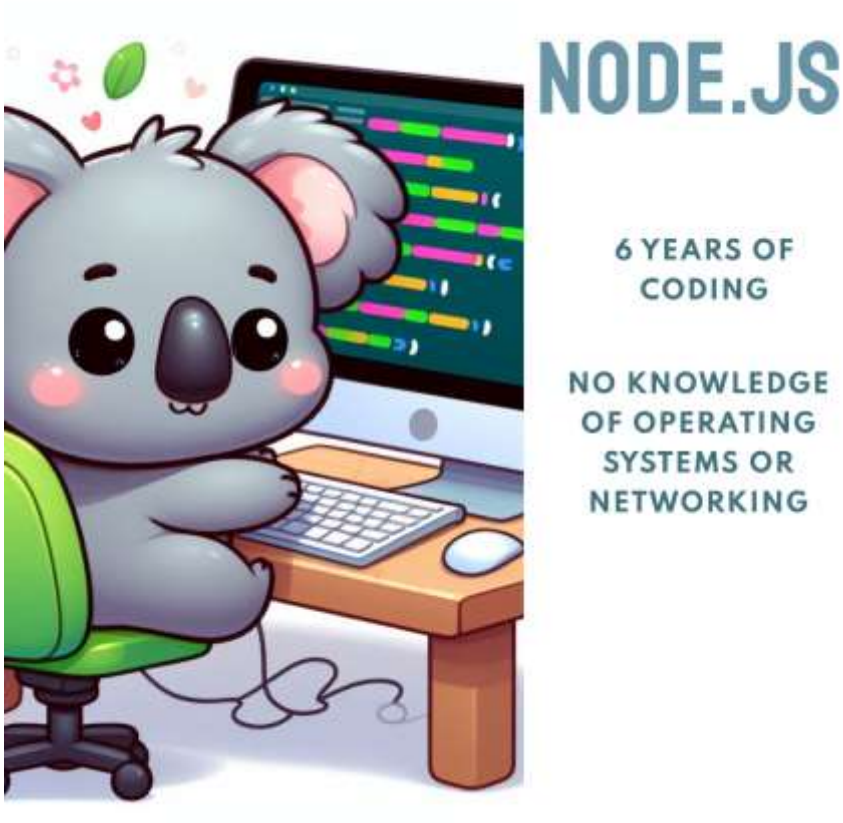
Meet our Koala!



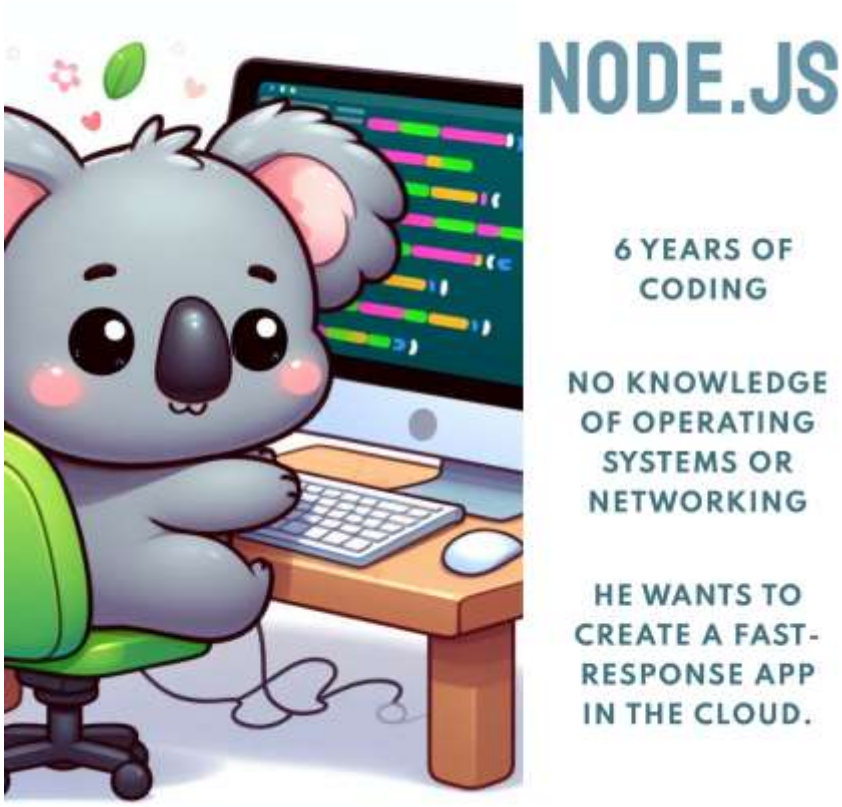
Meet our Koala!



Meet our Koala!



Meet our Koala!



What does he do?

Rents a VM!

What does he do?

Rents a VM!

Deploys his application

What does he do?

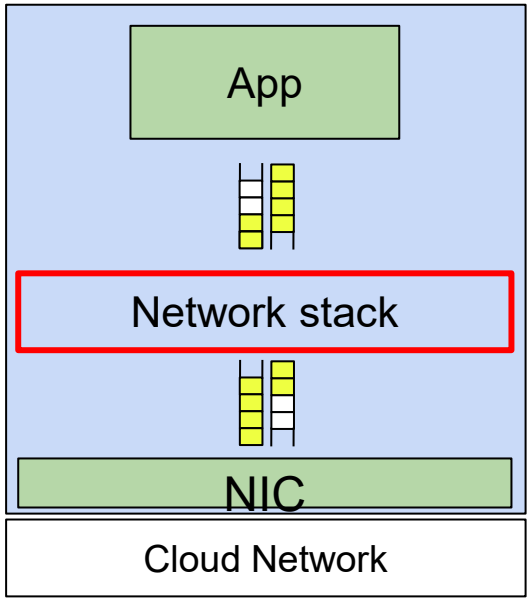
Rents a VM!

Deploys his application

And it is slow!

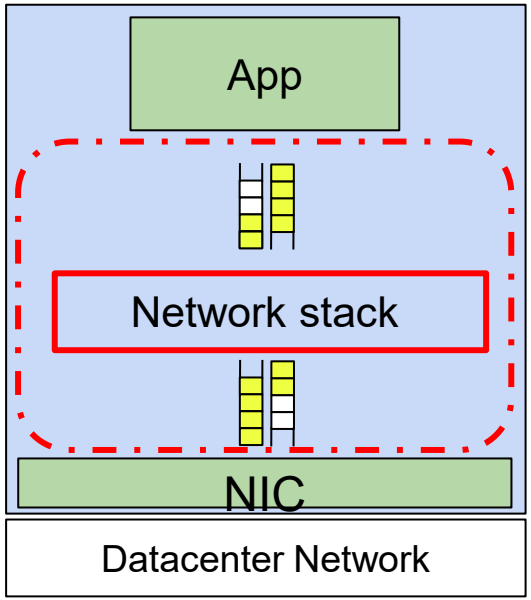
What a (**slow**) networking stack looks like:

What a (**slow**) networking stack looks like:



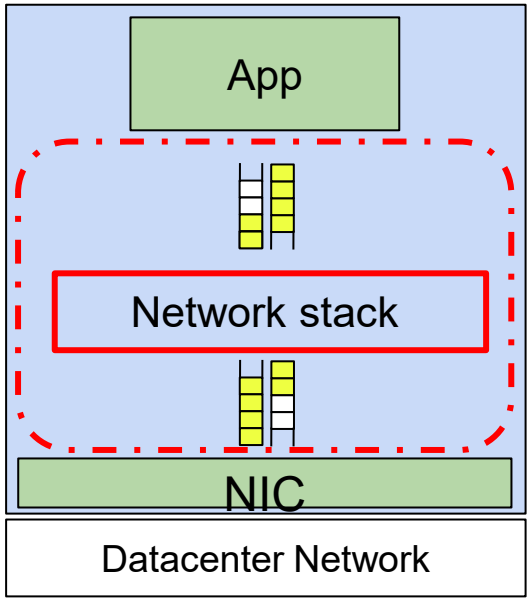
What a (**slow**) networking stack looks like:

Operating System



What a (**slow**) networking stack looks like:

Operating System



Usually networking through OS is slower!

There are many existing fast networking stacks!

IX: A Protec
High T

Adam Belay[1] Gec
Chri

Zyg(
Mic

Shenango: A

Datacenter RPCs can be General and Fast

Anuj Kalia, *Carnegie Mellon University;* Michael Kaminsky, *Intel Labs;*
David Anderson, *Carnegie Mellon University*

When Idling is Ideal: Optimizing Tail-Latency for Heavy-Tailed Datacenter Workloads with Perséphone

Henri Maxime Demoulin, University of Pennsylvania, USA
Joshua Fried, MIT CSAIL, USA
Isaac Pedisich, Grammatech*, USA
Marios Kogias, Microsoft Research, United Kingdom
Boon Thau Loo, University of Pennsylvania, USA
Linh Thi Xuan Phan, University of Pennsylvania, USA
Irene Zhang, Microsoft Research, USA

nd-scale Tail Latency

Tigar Humphries[1]

nter Workloads

Amy Ousterhout, Joshua Fried, Jonathan Behrens, Adam Belay, Hari Balakrishnan
MIT CSAIL

There are many existing fast networking stacks!

IX: A Protec
High T

Adam Belay[1] Gec
Chri

Zyg(
Mic

Shenango: A

Datacenter RPCs can be General and Fast

Anuj Kalia, *Carnegie Mellon University;* Michael Kaminsky, *Intel Labs;*
David Anderson, *Carnegie Mellon University*

When Idling is Ideal: Optimizing Tail-Latency for Heavy-Tailed Datacenter Workloads with Perséphone

Henri Maxime Demoulin, University of Pennsylvania, USA
Joshua Fried, MIT CSAIL, USA
Isaac Pedisich, Grammatech*, USA
Marios Kogias, Microsoft Research, United Kingdom
Boon Thau Loo, University of Pennsylvania, USA
Linh Thi Xuan Phan, University of Pennsylvania, USA
Irene Zhang, Microsoft Research, USA

nd-scale Tail Latency

Tigar Humphries[1]

nter Workloads

Amy Ousterhout, Joshua Fried, Jonathan Behrens, Adam Belay, Hari Balakrishnan
MIT CSAIL

NIC fancy features + kernel bypass


Fast
network
stacks!

What does he do?

What does he do?

Rents a VM!

What does he do?

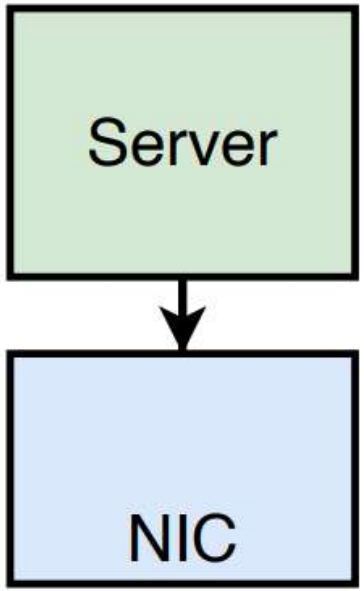
Rents a VM!

Not able to run those fast networking stacks on cloud VMs!

What is going on?

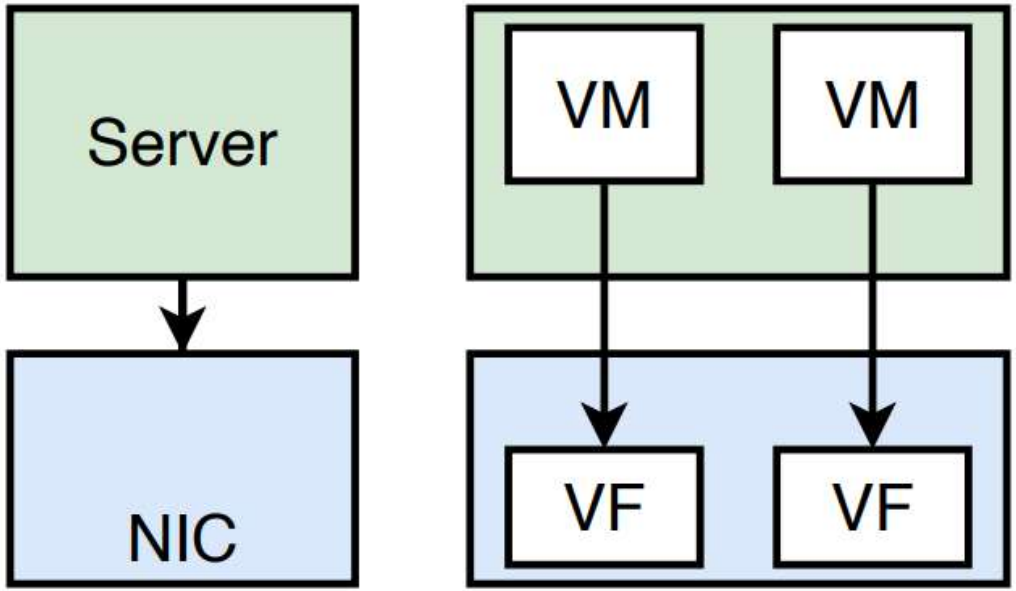
To understand **Why those systems do not work,** let’s compare Cloud VM and Bare metal networking

To understand **what's missing** let's compare Cloud VM and Bare metal networking



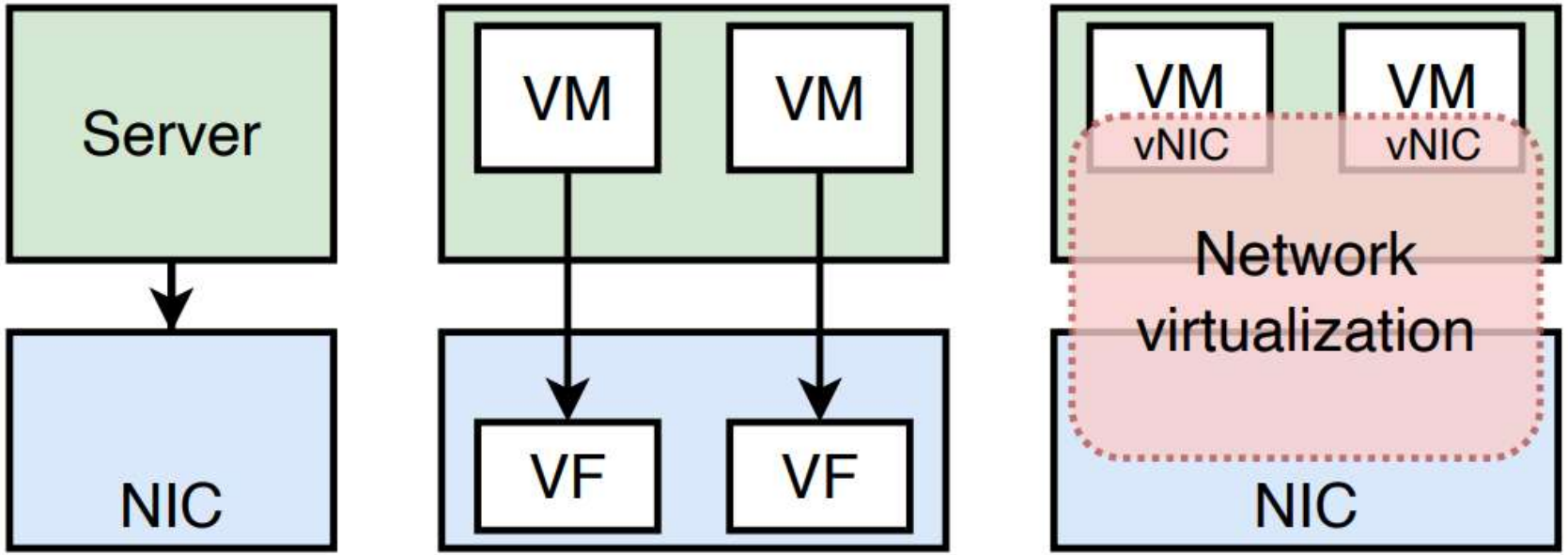
(a) Bare-metal NIC

To understand **what's missing** let's compare Cloud VM and Bare metal networking



(a) Bare-metal NIC (b) Virtual functions

To understand **what's missing** let's compare Cloud VM and Bare metal networking



(a) Bare-metal NIC (b) Virtual functions (c) Cloud vNICs

What's exactly missing in vNICs?

NIC feature	Year of introduction	Systems
Flow steering / RSS reconfiguration	ConnectX3 (2011)	eRPC, Snap, Shinjuku, IX, TAS, RSS++, mTCP
Deep RX queues Multi-packet RQs	ConnectX4 (2014)	eRPC, Virtuoso, Junction
TX DMA from app memory Remote DMA	ConnectX3 (2011)	Cornflakes, eRPC, SocksDirect, mRPC
Poll Event Queue	ConnectX4 (2014)	Junction

What's exactly missing in vNICs?

NIC feature	Year of introduction	Systems
Flow steering / RSS reconfiguration	ConnectX3 (2011)	eRPC, Snap, Shinjuku, IX, TAS, RSS++, mTCP
Deep RX queues Multi-packet RQs	ConnectX4 (2014)	eRPC, Virtuoso, Junction
TX DMA from app memory Remote DMA	ConnectX3 (2011)	Cornflakes, eRPC, SocksDirect, mRPC
Poll Event Queue	ConnectX4 (2014)	Junction

Because of those net virtualization constraints we cannot use these fast systems

Can we design a new fast userspace network stack that does not use any fancy/highend NIC features?

What are the new requirements for user space network stacks?

What are the new requirements for user space network stacks?

NIC agnostic

What are the new requirements for user space network stacks?

NIC agnostic

High level language binding

High level components and design choices!

We should move data from NIC to userspace quickly

High level components and design choices!

We should move data from NIC to userspace quickly

DPDK

AF_XDP

High level components and design choices!

We should move data from NIC to userspace quickly

DPDK

AF_XDP

High level components and design choices!

We need a transport that can relay messages to applications

High level components and design choices!

We need a transport that can relay messages to applications

Sidecar

LibOS

High level components and design choices!

We need a transport that can relay messages to applications

Sidecar

LibOS

High level components and design choices!

Connection state management in multi-core scenarios

High level components and design choices!

Connection state management in multi-core scenarios

Shared Nothing

Shared

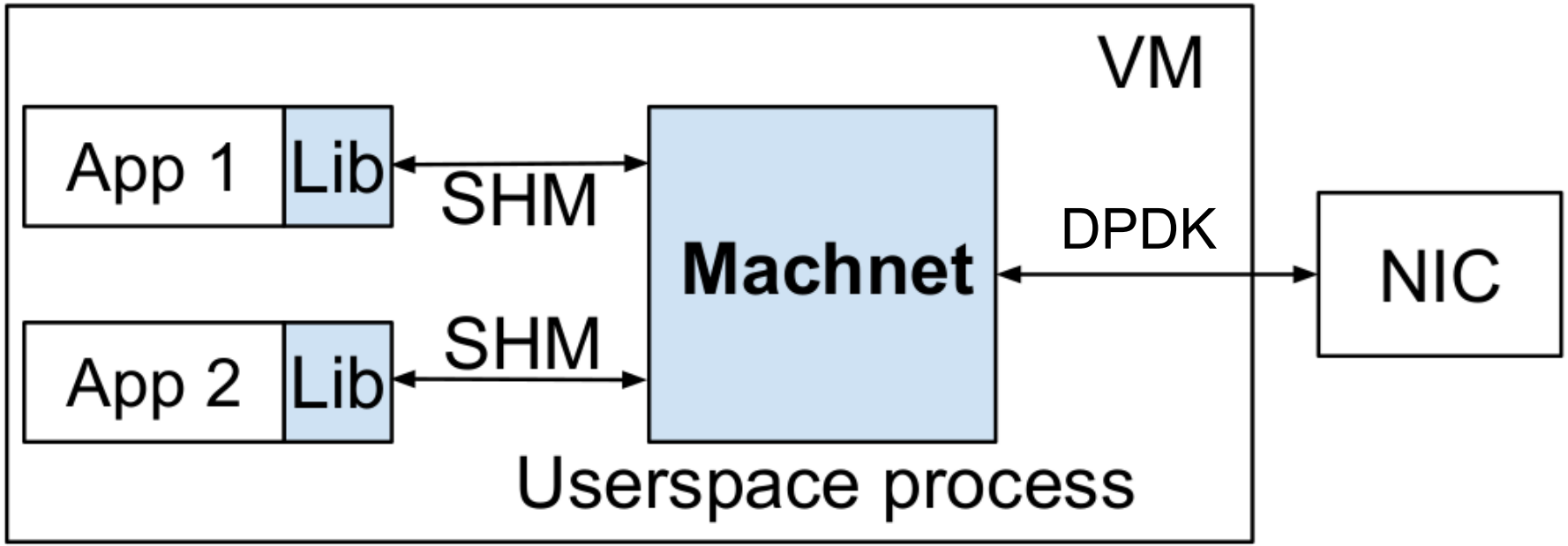
High level components and design choices!

Connection state management in multi-core scenarios

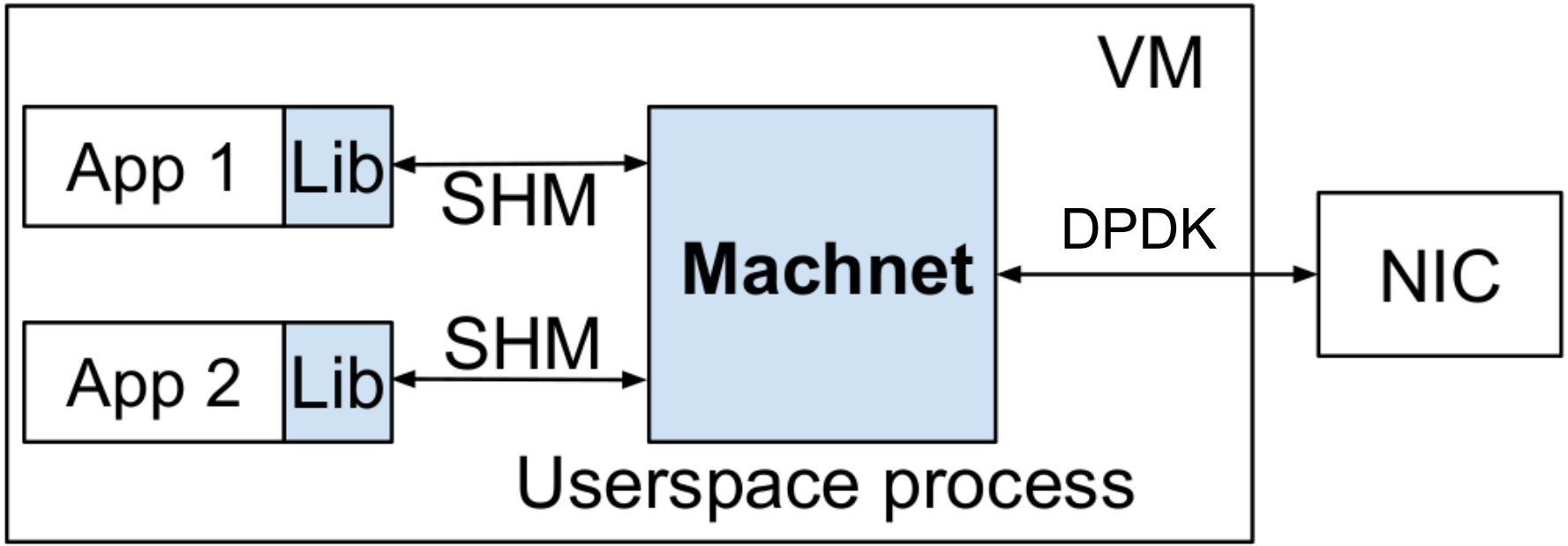
Shared Nothing

~~***Shared***~~

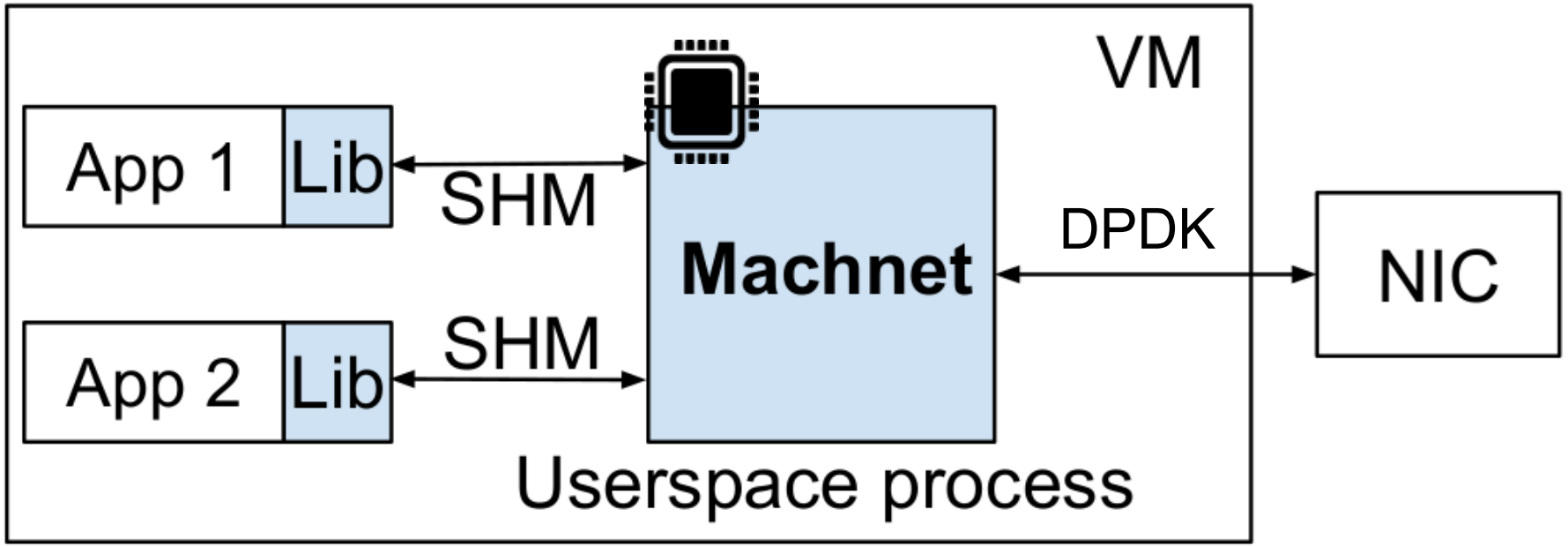
Machnet Design



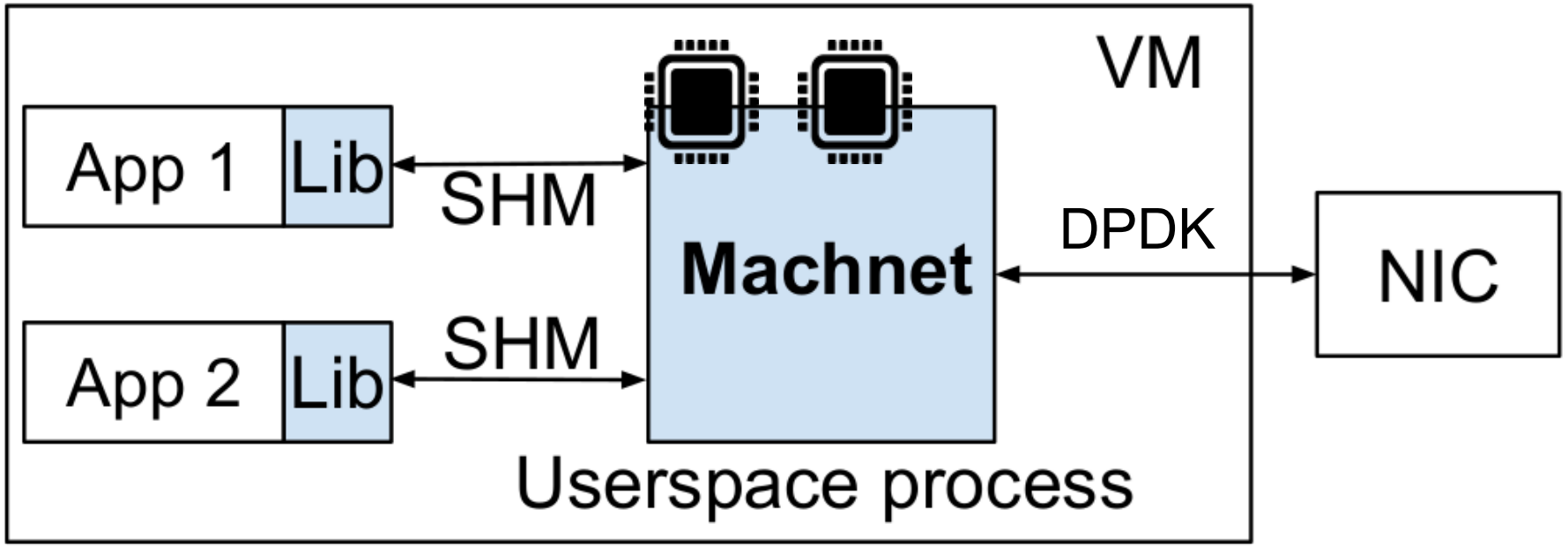
Machnet should support multiple CPU cores for higher throughput



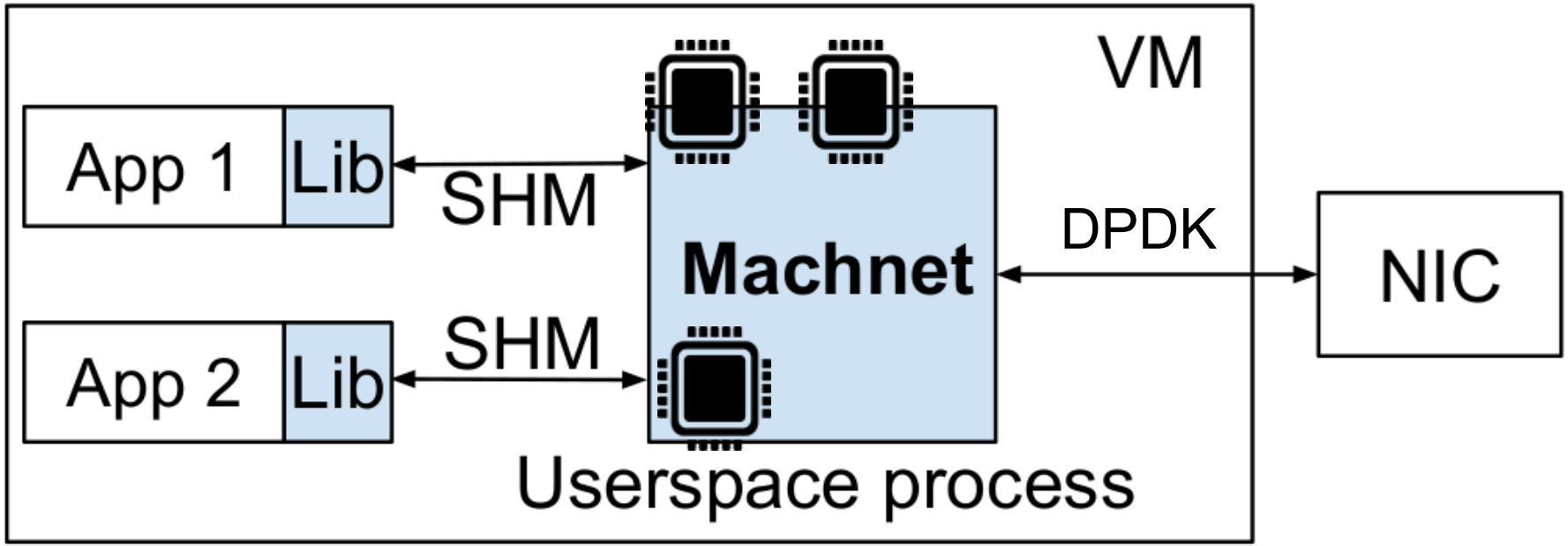
Machnet should support multiple CPU cores for higher throughput



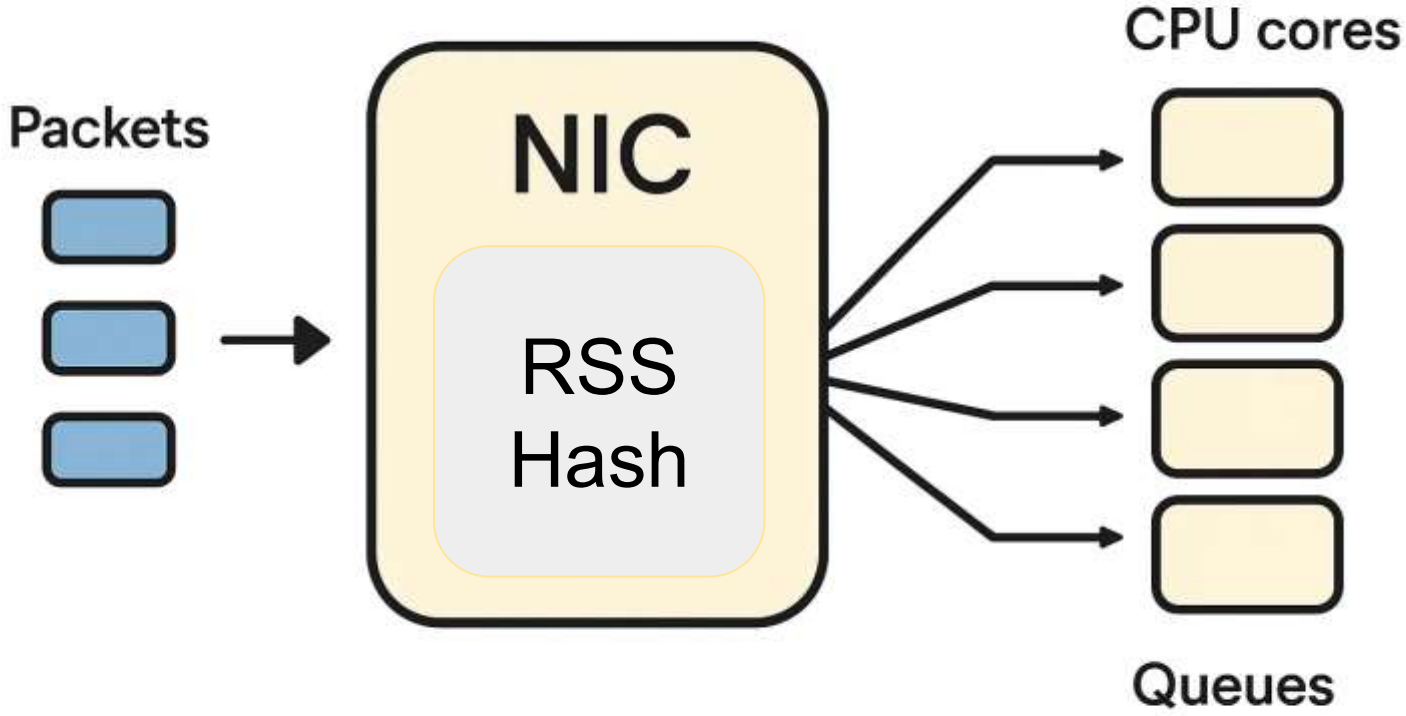
Machnet should support multiple CPU cores for higher throughput



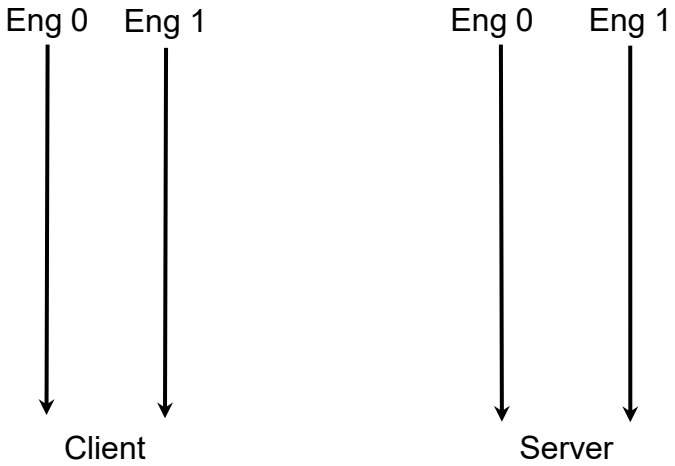
Machnet should support multiple CPU cores for higher throughput



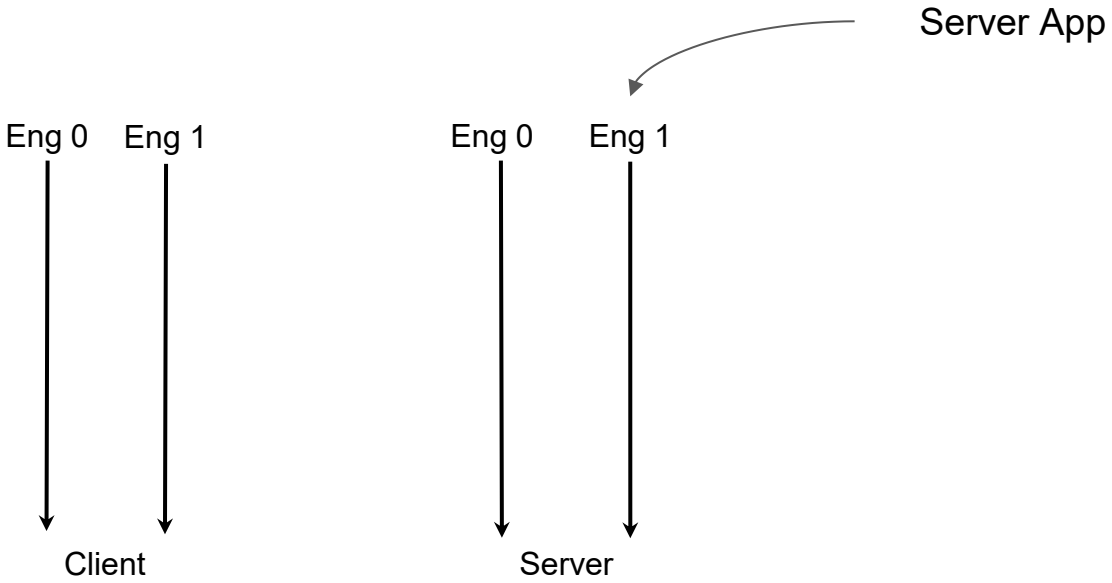
Receive Side Scaling (RSS)



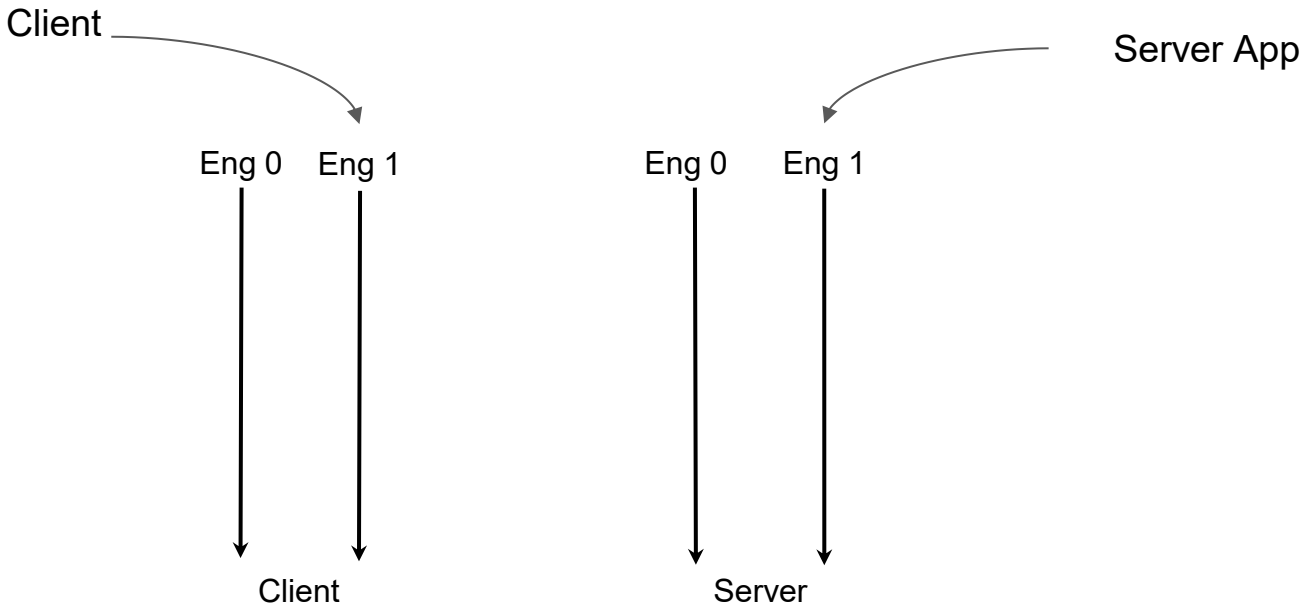
RSS-- design!



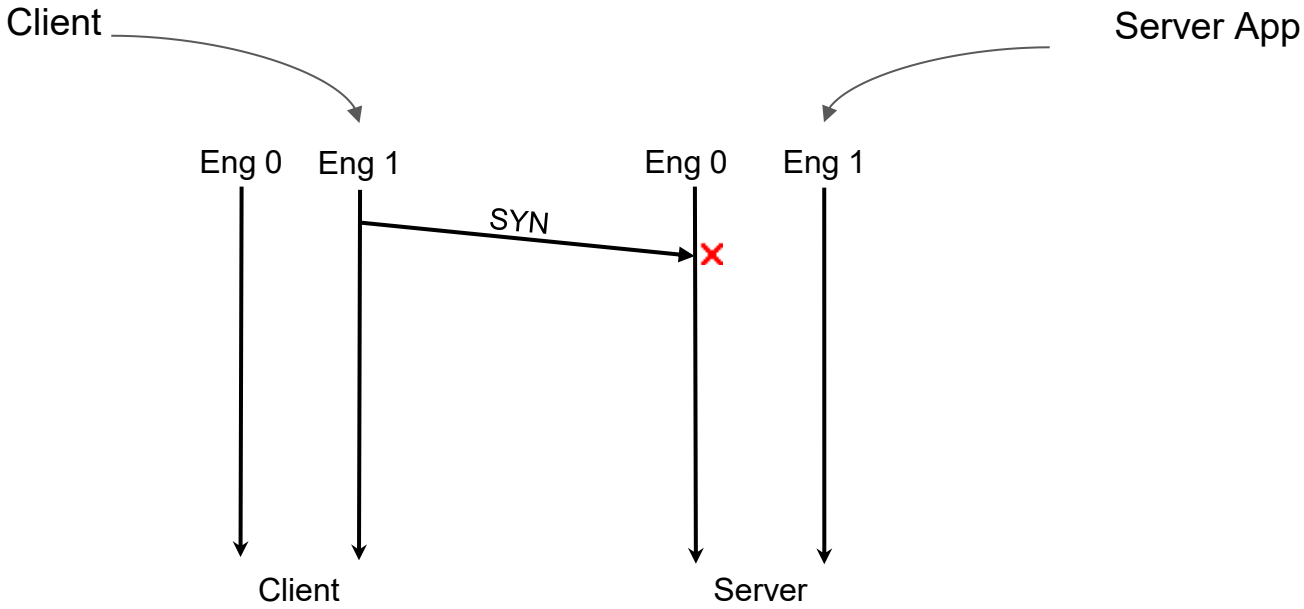
RSS-- design



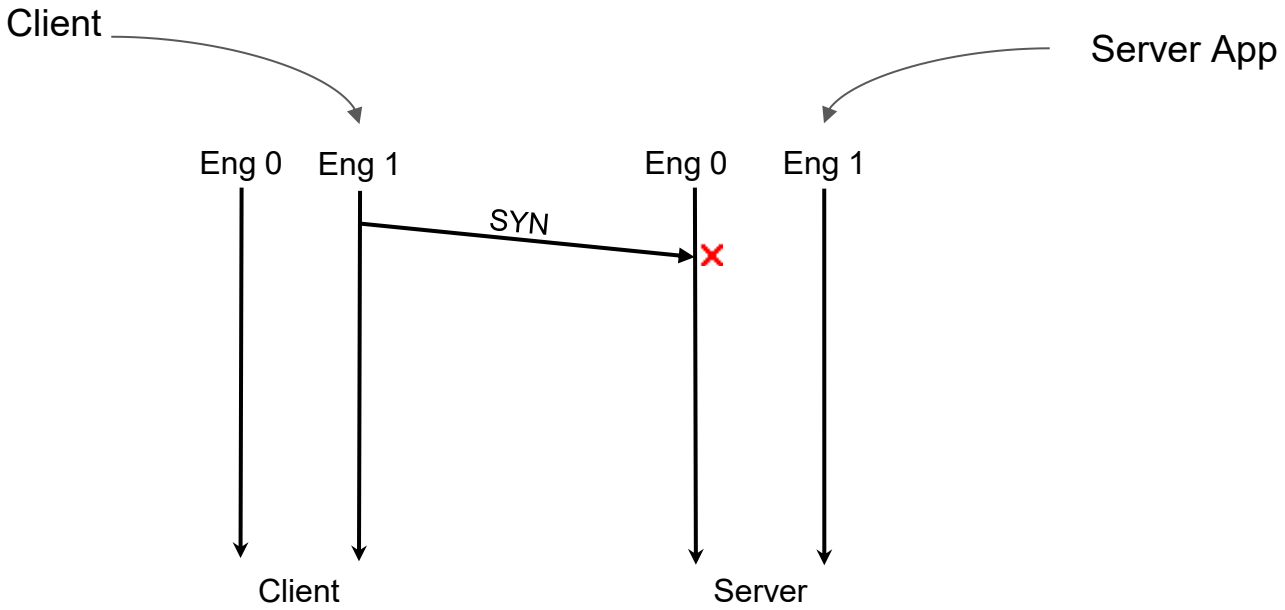
RSS-- design!



RSS-- design!

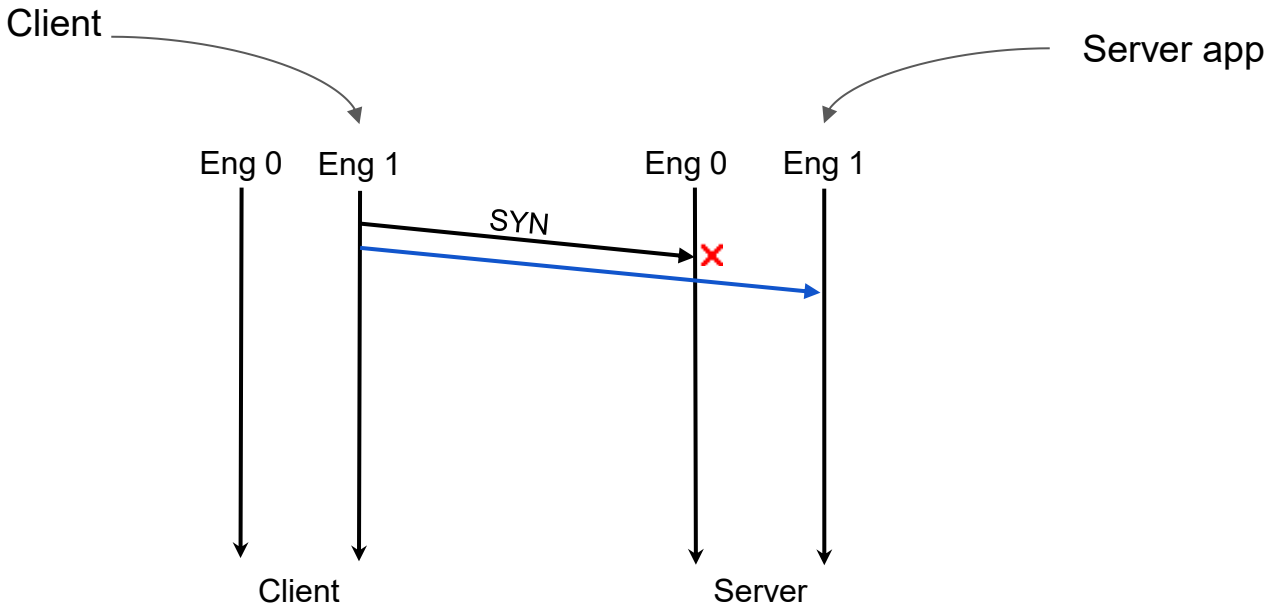


RSS-- design!

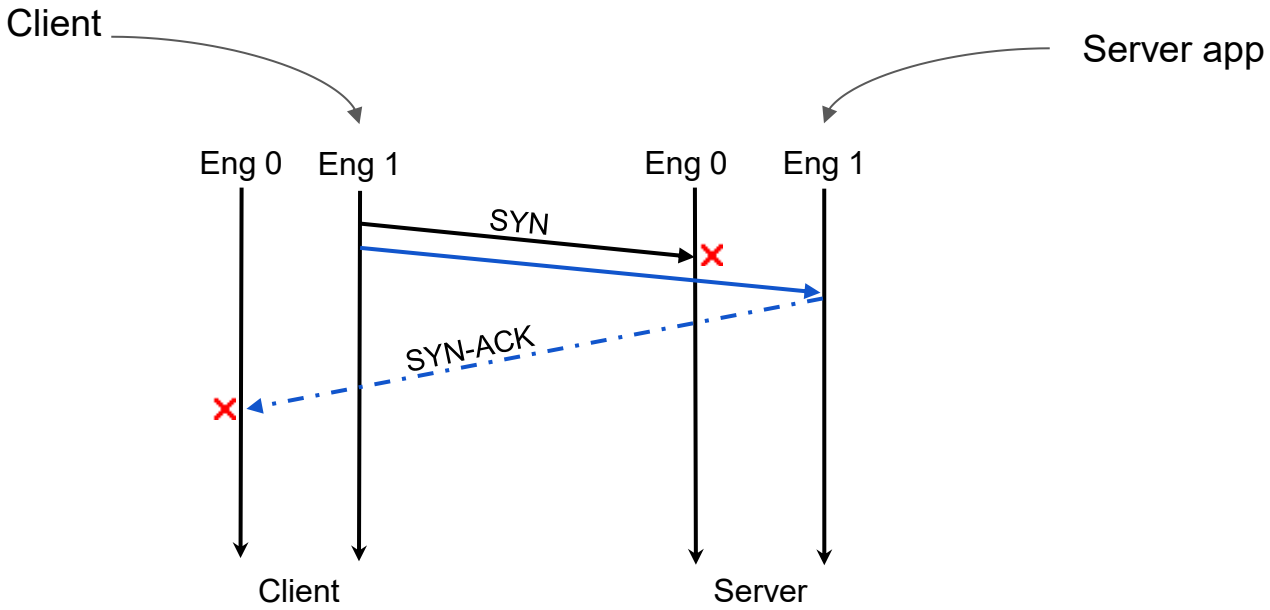


No connection

RSS-- design!

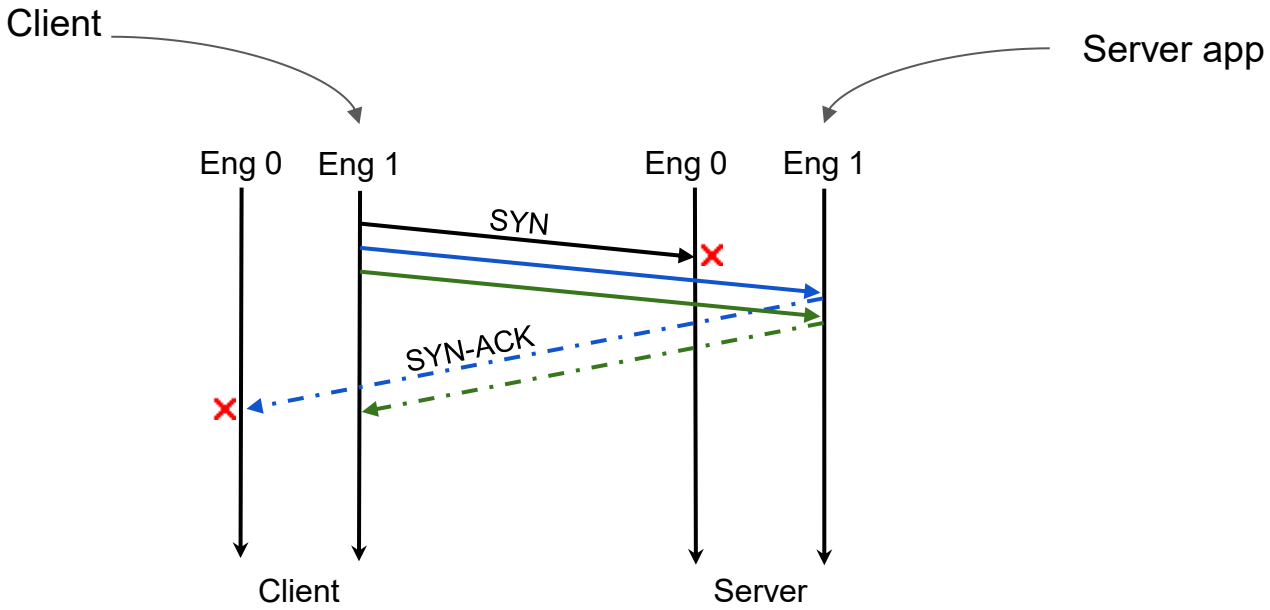


RSS-- design!

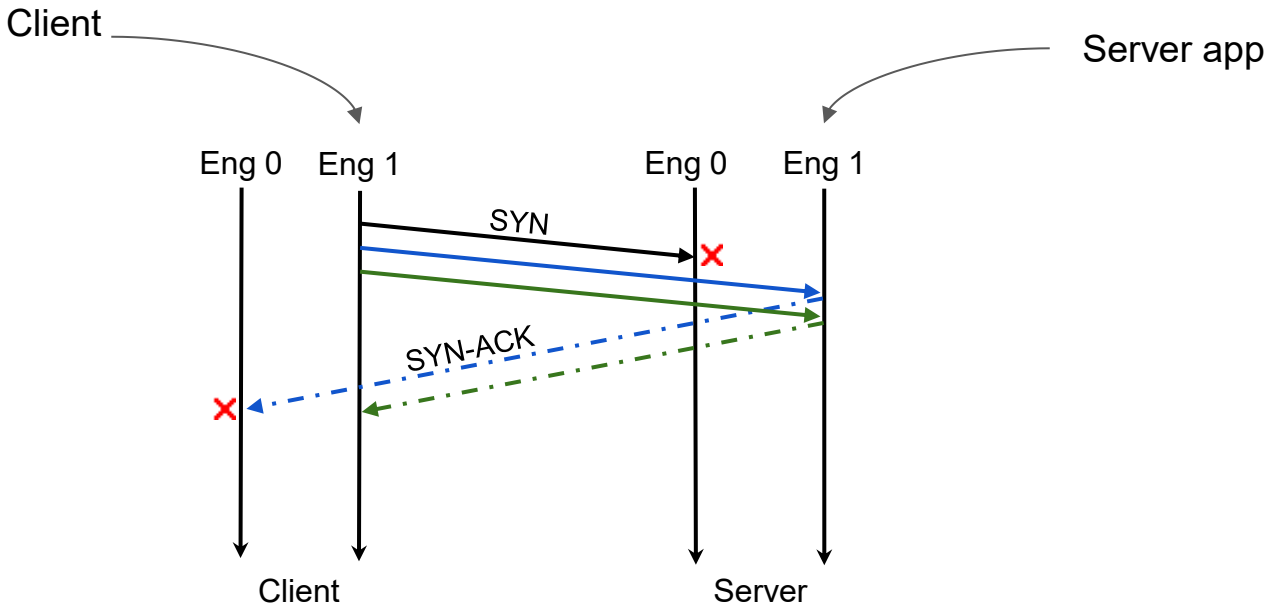


No connection

RSS-- design!

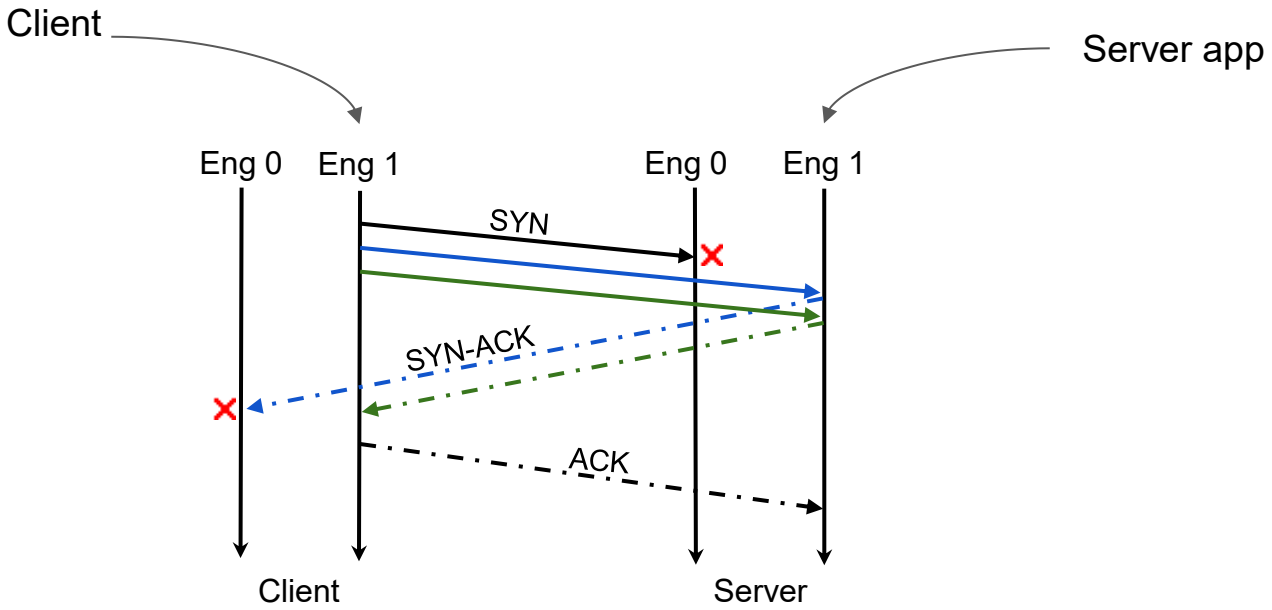

Client
Server app
Eng 0
Eng 1
Eng 0
Eng 1
SYN
SYN-ACK
Client
Server

RSS-- design!



HIT! We found a pair of port numbers that match!

RSS-- design!



Connection fully established

Machnet is compatible with major cloud providers!

Machnet is compatible with major cloud providers!

Cloud provider	Size	p50	p99	p99.9
Microsoft Azure	64 B	27	32	49
	32 kB	81	97	159
Amazon EC2	64 B	48	53	57
	32 kB	224	240	257
Google Cloud	64 B	65	111	164
	32 kB	221	273	335

What is the performance of real world applications using Machnet?

What is the performance of real world applications using Machnet?

Key-value store FASTER

What is the performance of real world applications using Machnet?

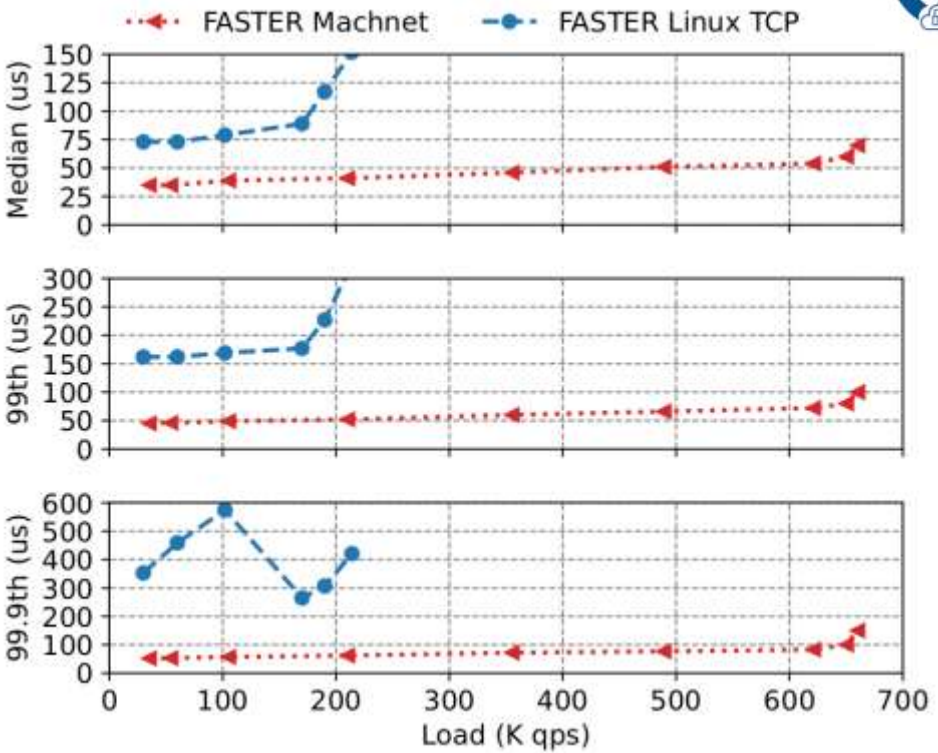
Key-value store FASTER

Raft Consensus protocol

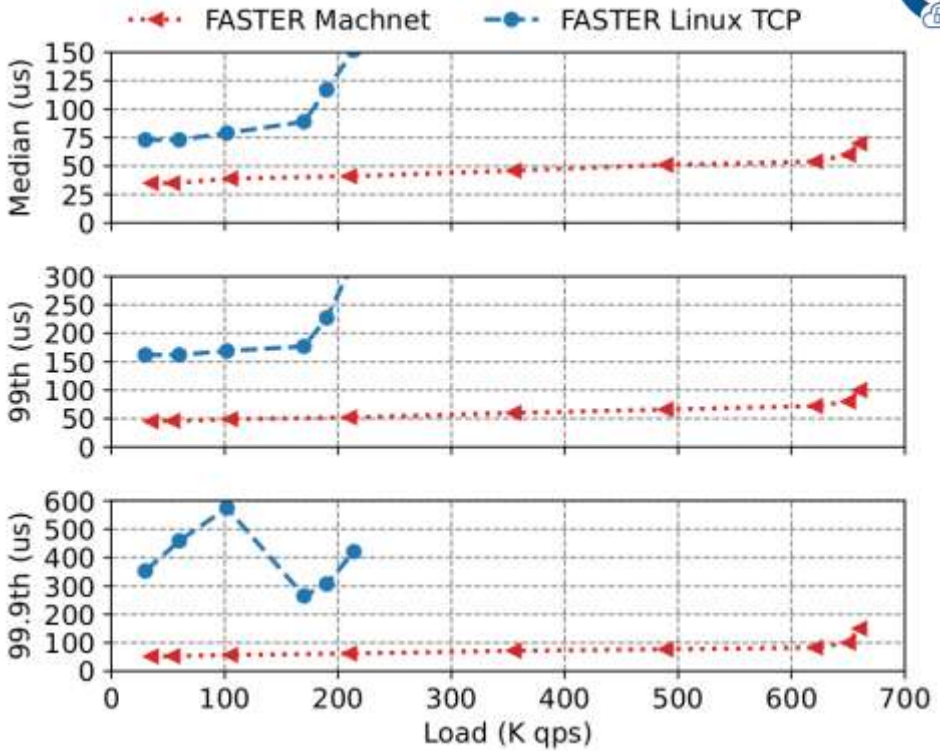
Any real world applications?

FASTER

Any real world applications?



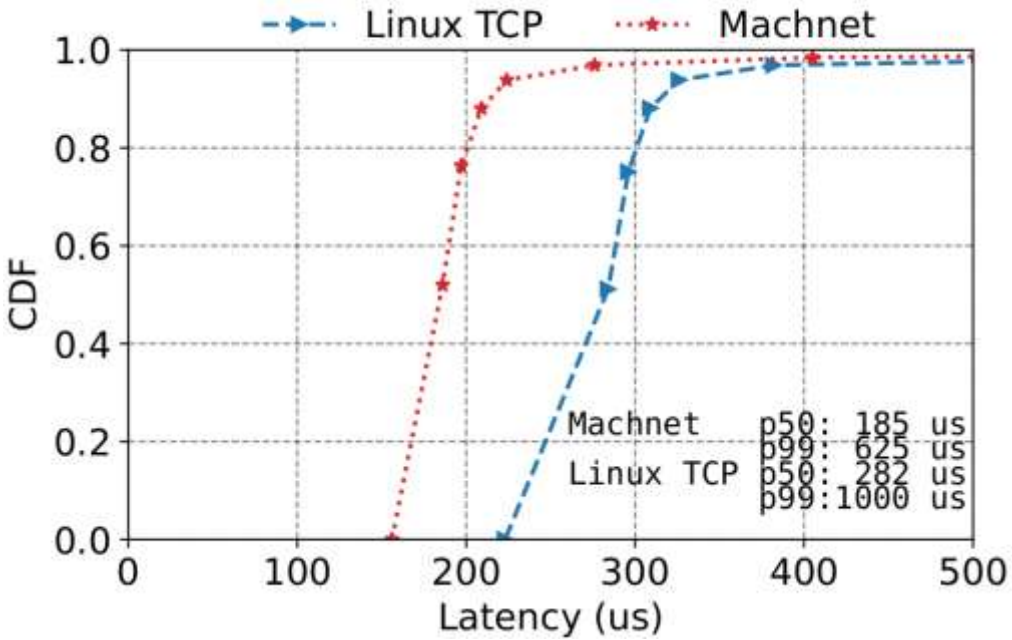
Any real world applications



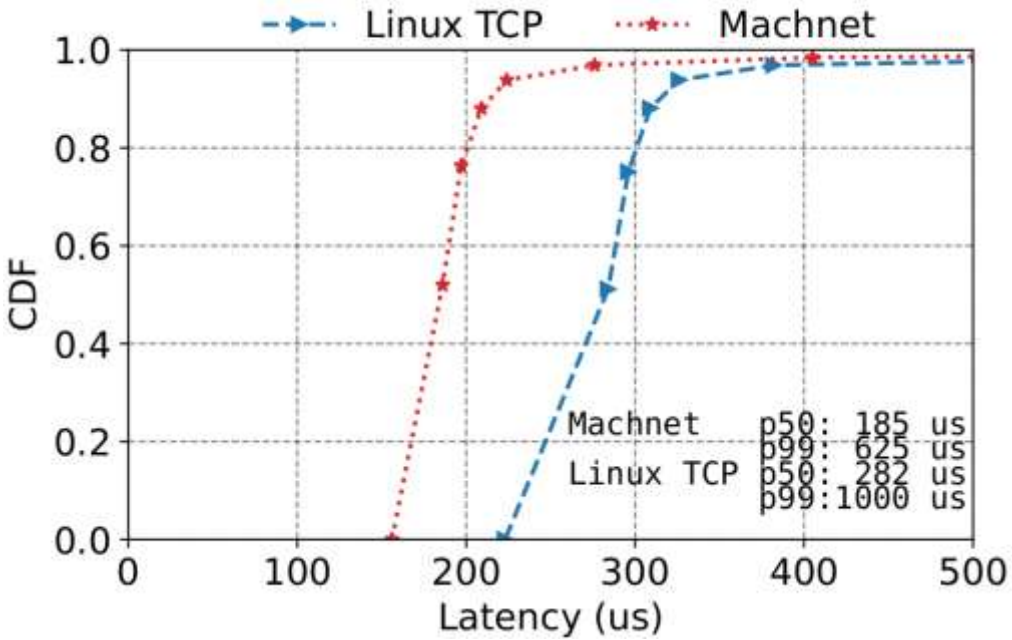
FASTER key-value runs with Machnet and is actually FASTER

Any real world applications?

Any real world applications?



Any real world applications?



2x better tail latency at 99.9th

Takeaways

Takeaways

Cloud VMs are a decade behind in terms of employing hardware features!

Takeaways

Cloud VMs are a decade behind in terms of employing hardware features!

None of existing solutions can improve Cloud VMs latency

Takeaways

Cloud VMs are a decade behind in terms of employing hardware features!

None of existing solutions can improve Cloud VMs latency

DPDK created a graveyard of userspace networking stacks due to its development pace

README Code of conduct MIT license Security

Machnet: Easy kernel-bypass messaging between cloud VMs

Build and Register Machnet as Latest passing

Machnet provides an easy way for applications to reduce their datacenter networking latency via kernel-bypass (DPDK-based) messaging. Distributed applications like databases and finance can use Machnet as the networking library to get sub-100 microsecond tail latency at high message rates, e.g., **750,000 1KB request-reply messages per second on Azure F8s_v2 VMs with 61 microsecond P99.9 round-trip latency**. We support a variety of cloud (Azure, AWS, GCP) and bare-metal platforms, OSs and NICs, evaluated in docs/PERFORMANCE_REPORT.md.

Machnet

Machnet: Easy kernel-bypass messaging between cloud VMs

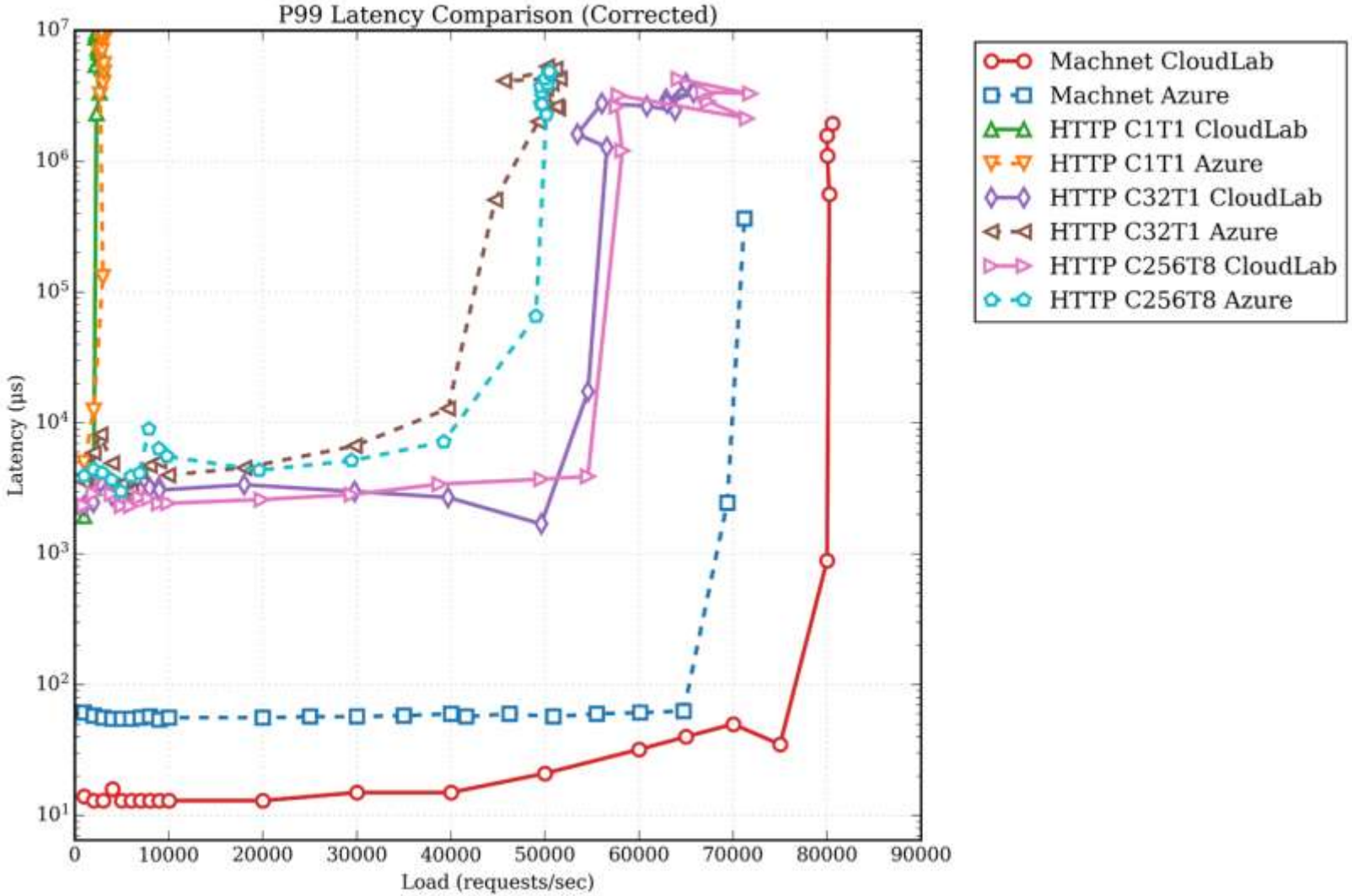
Machnet Tutorial - 5min ⏱️

Check out our white paper 📜

Interests from database community

mongoDB®

Axiom


P99 Latency Comparison (Corrected)
Latency (µs)
Load (requests/sec)
Machnet CloudLab
Machnet Azure
HTTP C1T1 CloudLab
HTTP C1T1 Azure
HTTP C32T1 CloudLab
HTTP C32T1 Azure
HTTP C256T8 CloudLab
HTTP C256T8 Azure

Our discord https://discord.gg/Usexu9fEg3