Networking as a First-Class Cloud Resource

Boston University

Wednesday, October 17th, 2018

Rodrigo Fonseca

Brown University

This talk

- Joint work with
  - Da Yu (Brown),
  - Shuwen Sun, Raja Sambasivan, Orran Krieger, Piyanai Saowarattitada, Jason Hennessey (BU),
  - Luo Mai (Imperial)
- More of a *question* than an *answer* !

Cloud Interface Today

- CPU, Memory, Storage: fine grained choices
  - Counted 83 instance types on AWS recently
- Networking choices are a lot coarser
  - Per-VM maximum bandwidth (“up to 10Gpbs”)
  - Can specify virtual networks, reachability, firewall rules
  - But hardly any guarantees

At the same time…

- Applications have very different requirements
  - Low latency, high bandwidth, low variance, deadlines, path redundancy, …
  - Some applications have *less* requirements
  - State of the art is to overprovision BW, pray for latency

Can we do better?

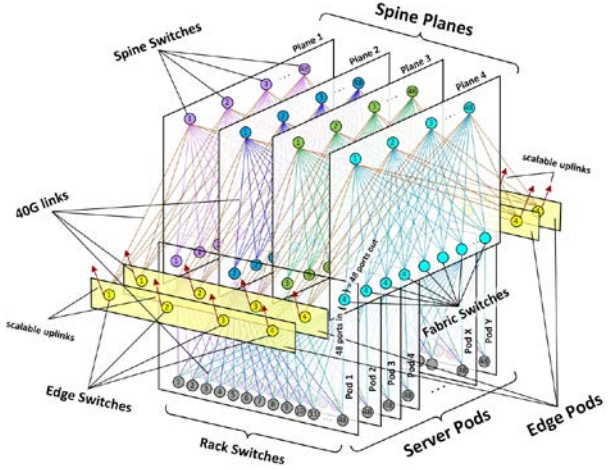
Datacenter Networking Innovation

“New” network topologies

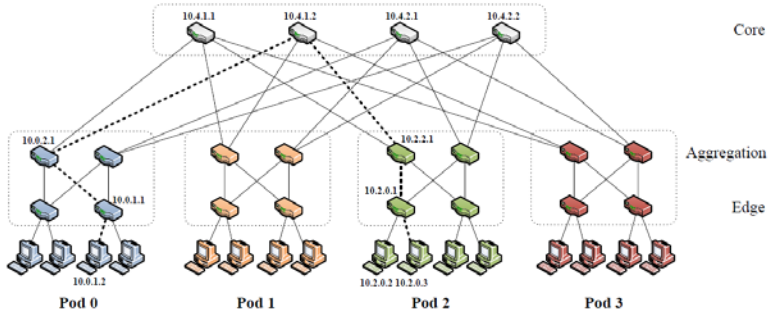
CLOS Fabrics

FatTrees

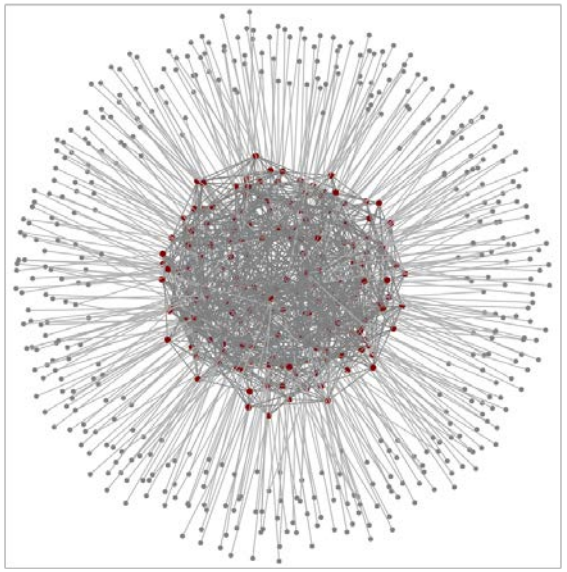
Randomized Topologies



Credit: Facebook



Fat Tree. Al-Fares et al. 2009



Jellyfish, Singla et al. 2012

Datacenter Networking Innovation

New Control Planes

SDN

Network Virtualization

NFV

Datacenter Networking Innovation

Programmable Dataplanes

In-band Network Telemetry
In-network caching
In-network Paxos
…

Datacenter Networking Innovation

New Transport Protocols

DCTCP

PDQ, D3, D^2TCP

pFabric, Qjump, NDP

Datacenter Networking Innovation

New Scheduling Algorithms

Coflows

FastPass

Programmable queueing disciplines

…

Datacenter Networking Innovation

New “crazy” ideas

Jellyfish topology

Disco Ball

Flywheels

Rotornet

…

Datacenter Networking Innovation

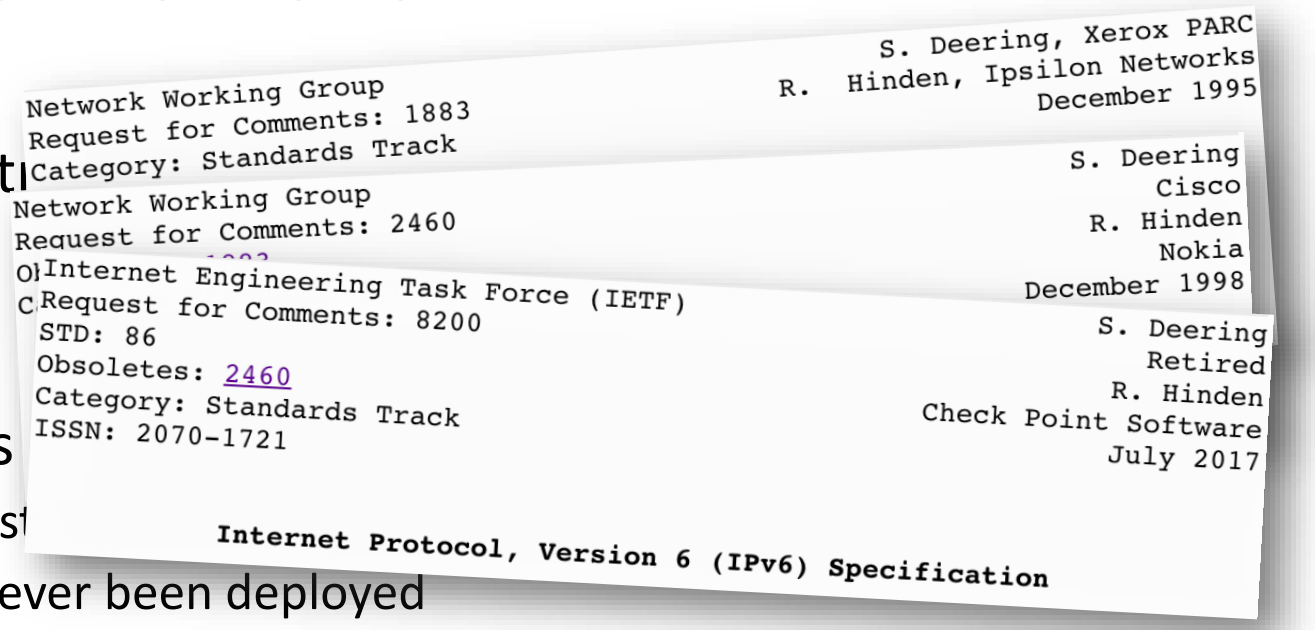
Much better than the Internet!

Why?

- Single administ

Really?

- Most protocols
  - Cannot coexist
  - Many have never been deployed



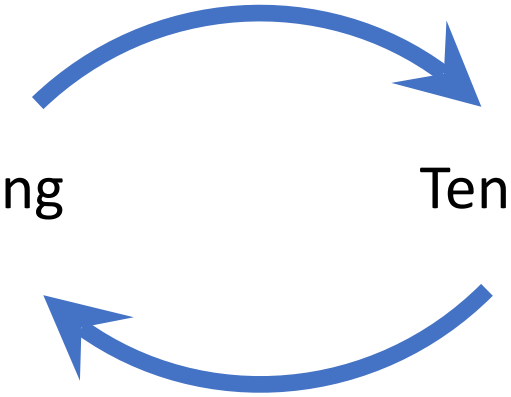
A single datacenter can be *more* ossified than the Internet!

Lack of Flexibility

Usually uniform topology, congestion control, scheduling

Really high bar for new proposals

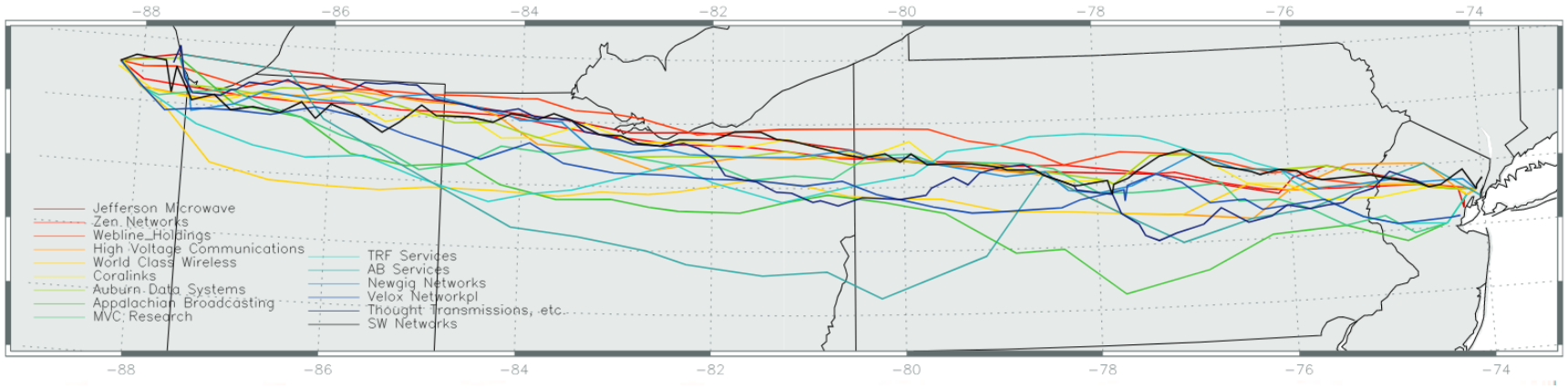
- Must be good for *all* kinds of traffic

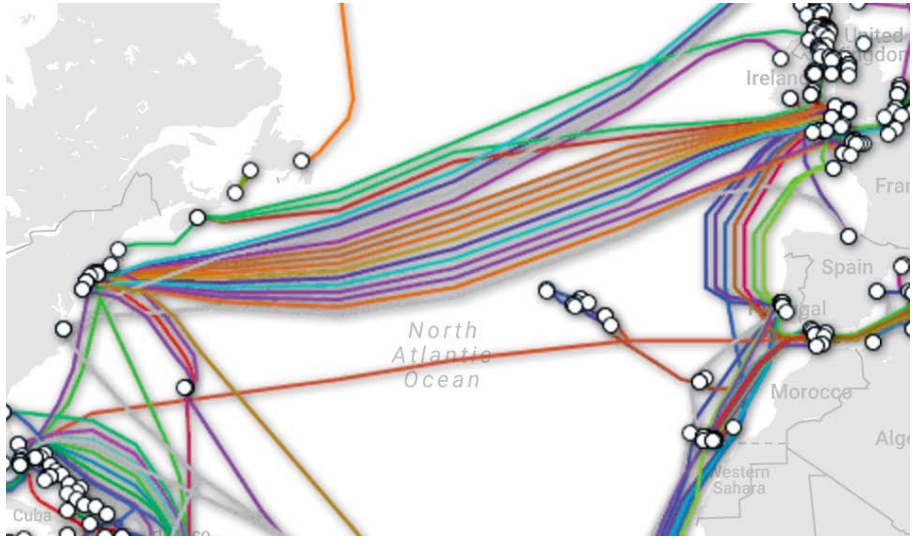


Providers: offer uniform networking

Tenants: don’t express requirements

Maybe we can learn from the Internet

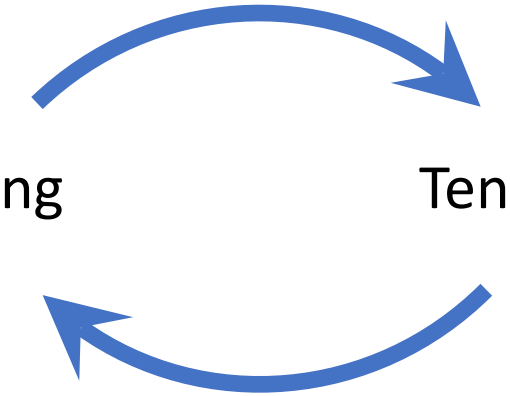




[1] Gregory Laughlin et al., ”Information Transmission Between Financial Markets in Chicago and New York”, arXiv:1302.5966 [q-fin.TR]
[2] https://www.submarinecablemap.com/

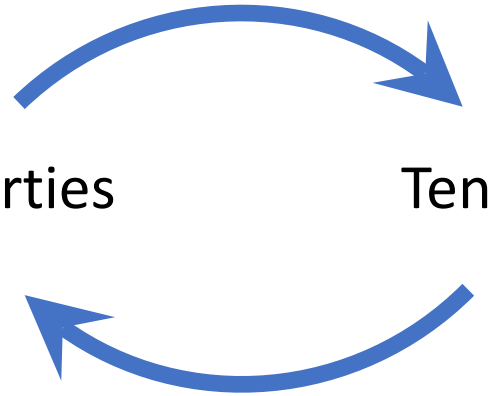
Catch-22

Providers: offer uniform networking



Tenants: don’t express requirements

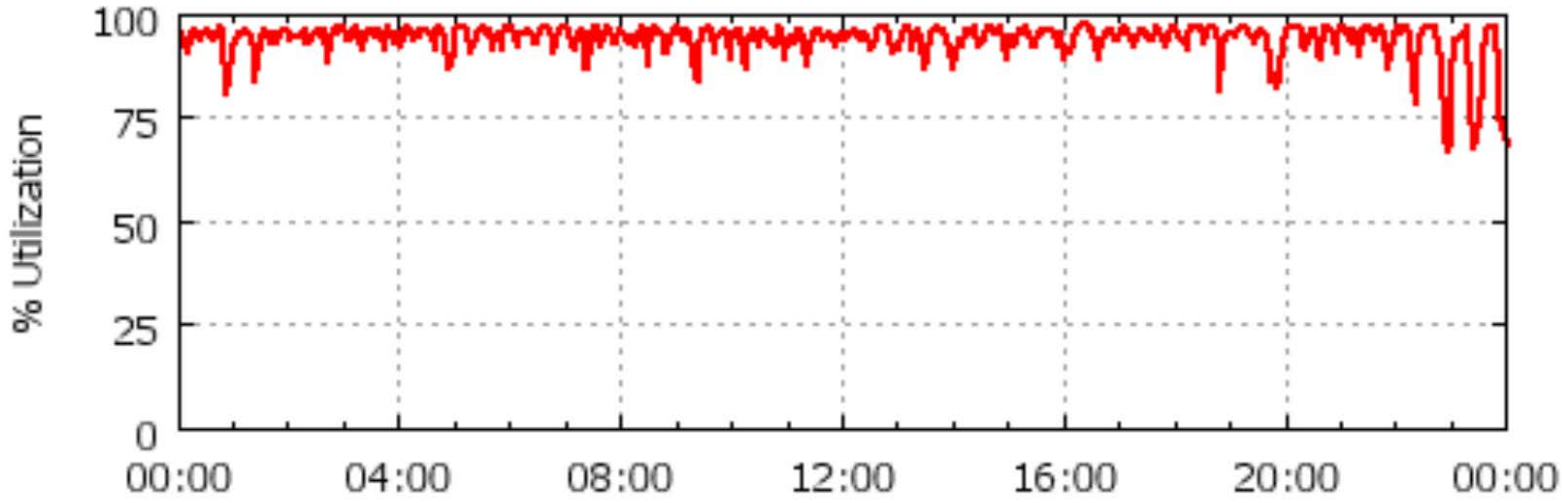
Providers: paths with different properties



Tenants: interface to express requirements

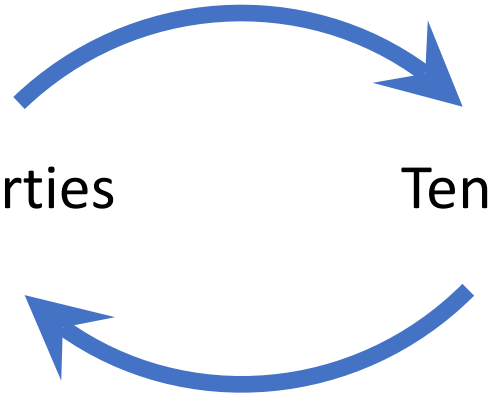
Example: Google B4*

- Mark packets as high priority / low priority
- Achieve near 100% utilization
  - (Trad. 30-40%)



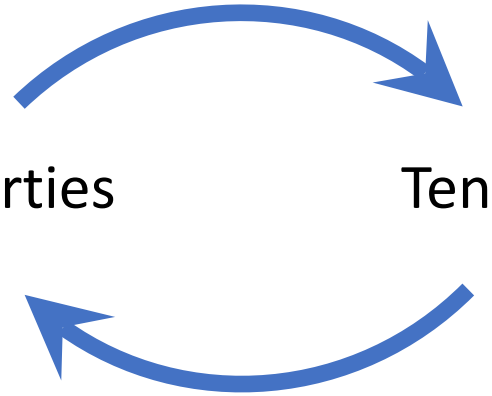
*Jain et al., B4: Experience with a Globally-deployed Software Defined Wan, Sigcomm 2013

Providers: paths with different properties



Tenants: interface to express requirements

FlexNet Datacenter Architecture



Providers: paths with different properties

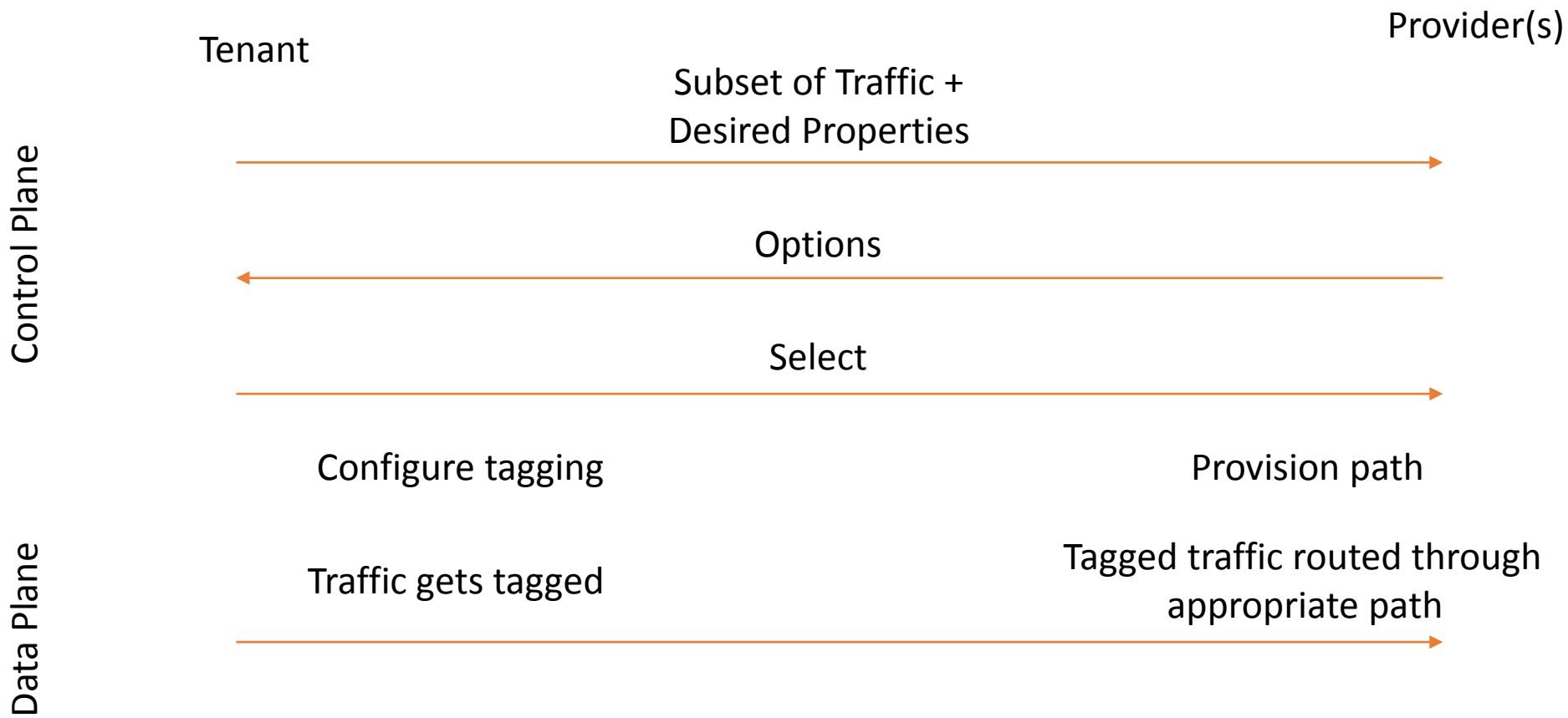
Tenants: interface to express requirements

Datacenter-assisted *tagging* of packets

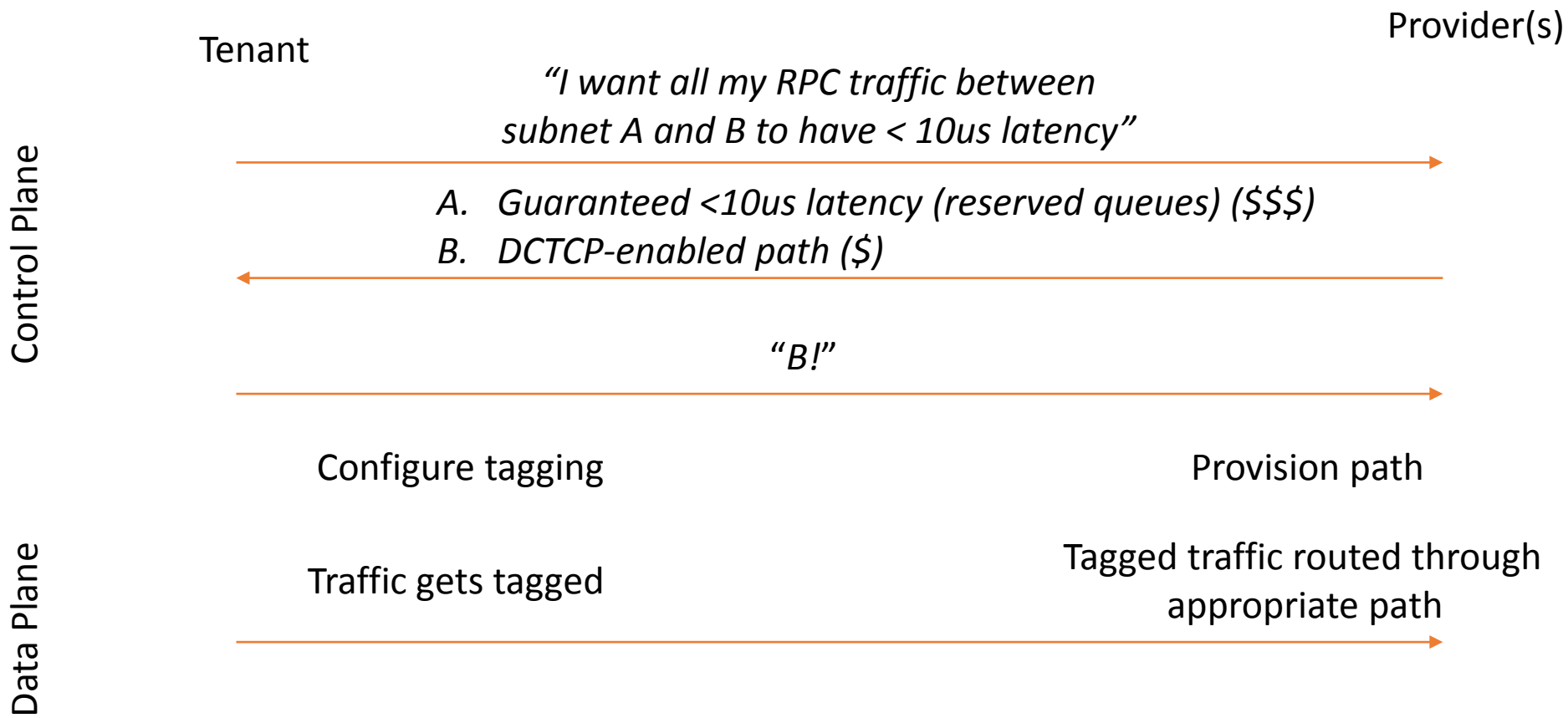
Out-of-band negotiation

Flexible forwarding

FlexNet Interface



FlexNet Interface



Flexible Tagging

- Any criterion based on packet headers (*flowspec)*
  - src or dest IP, port, protocol, or any combination
  - L7 features (e.g., HTTP cookie, header value)
  - Tenant id
  - …
- Could be done at hypervisor, vswitch, smartNIC, ToR
  - Provisioning can install match-action rules at any of these
- Could be implemented in any number of ways
  - VLAN tags, VXLAN tags, MPLS tags, encapsulated MAC addresses, etc…
- *Narrow waist* : only thing forwarding elements need to know

“Paths with Different Properties”

- On the same physical network
  - Could reserve bandwidth along a path
  - Could assign specific queues to a tag
  - Could configure all switches on a path to use DCTCP (ECN settings)
- On separate physical network
  - Could deploy pFabric-capable switches along a path
  - Could deploy 400Gbps Ethernet connecting a pair of racks, and provision to specific VMs/tags
  - Could upgrade the network and migrate progressively
- Paths can be offered with different prices

More on the interface

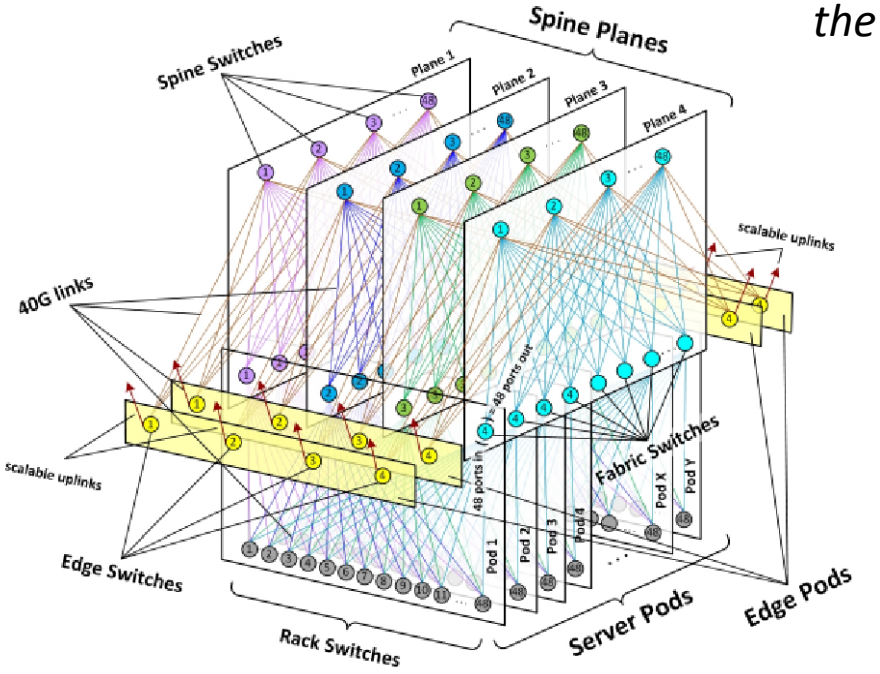
- Tenants usually configure subnets (“networks”)
- Our initial thought is to affect traffic among subnets
  - Assumes traffic within subnets is fine
- Could have this reach the nodes’ network interfaces as well
  - More on this later

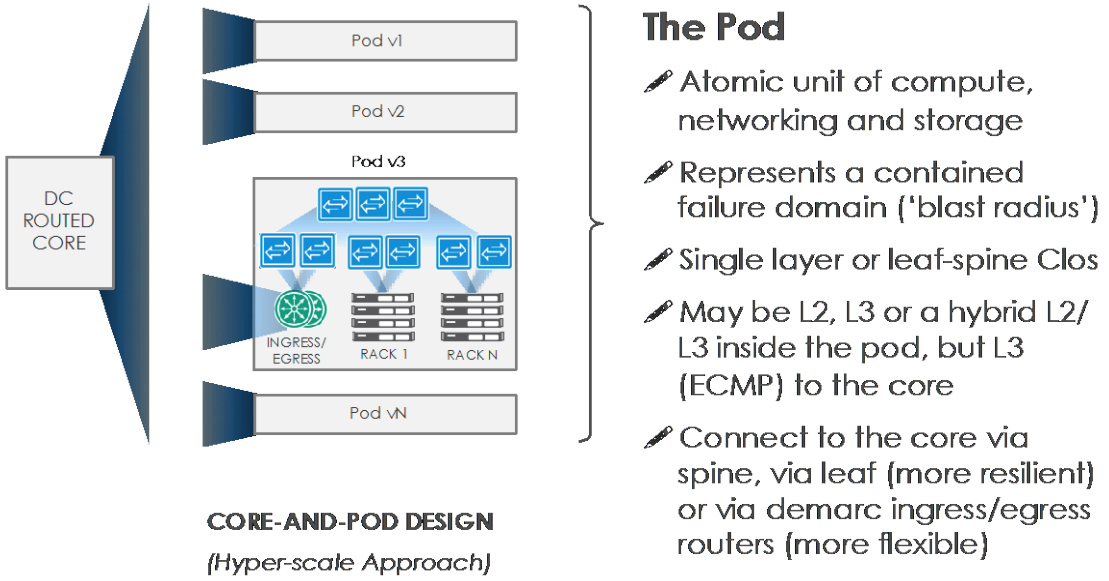
A FlexNet Prototype

Assumed datacenter architecture

Pod-and-core architecture

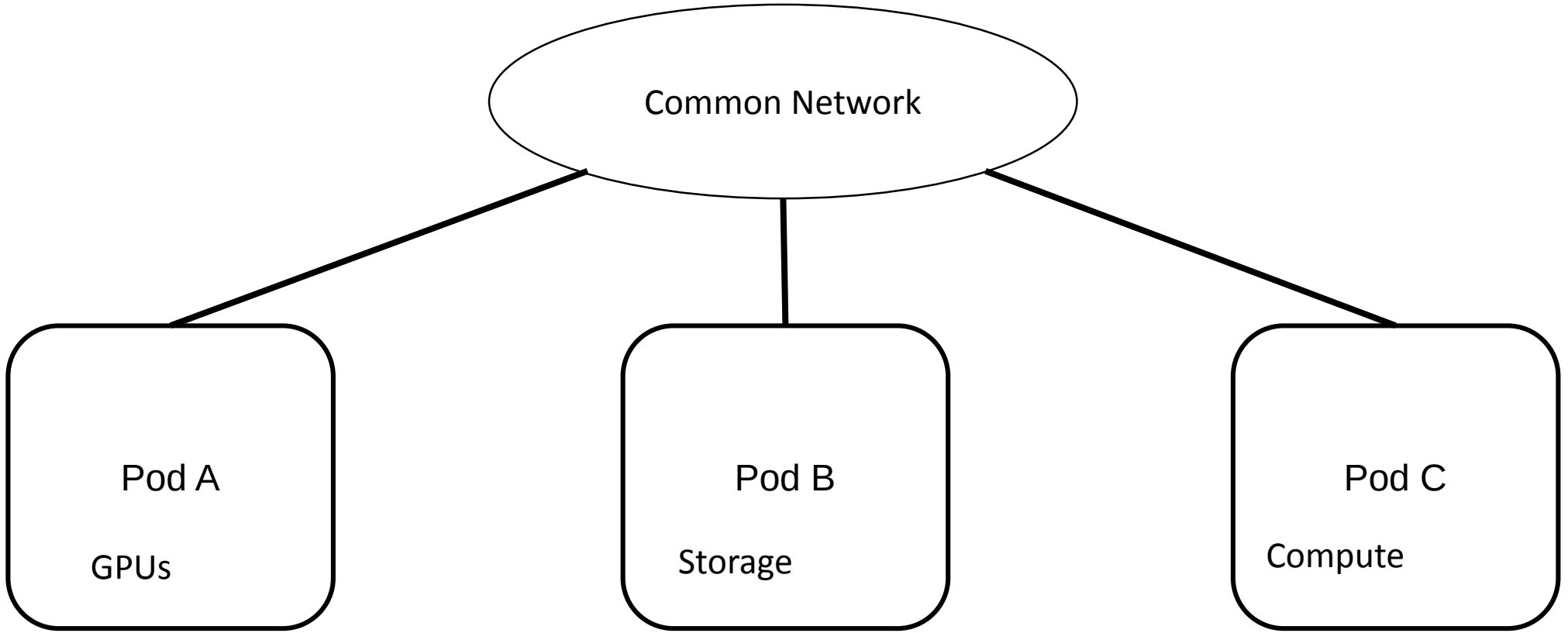
"Core and pod designs are increasingly recognized as practical adaptations of the hyperscale design approach, applicable to data centers worldwide."

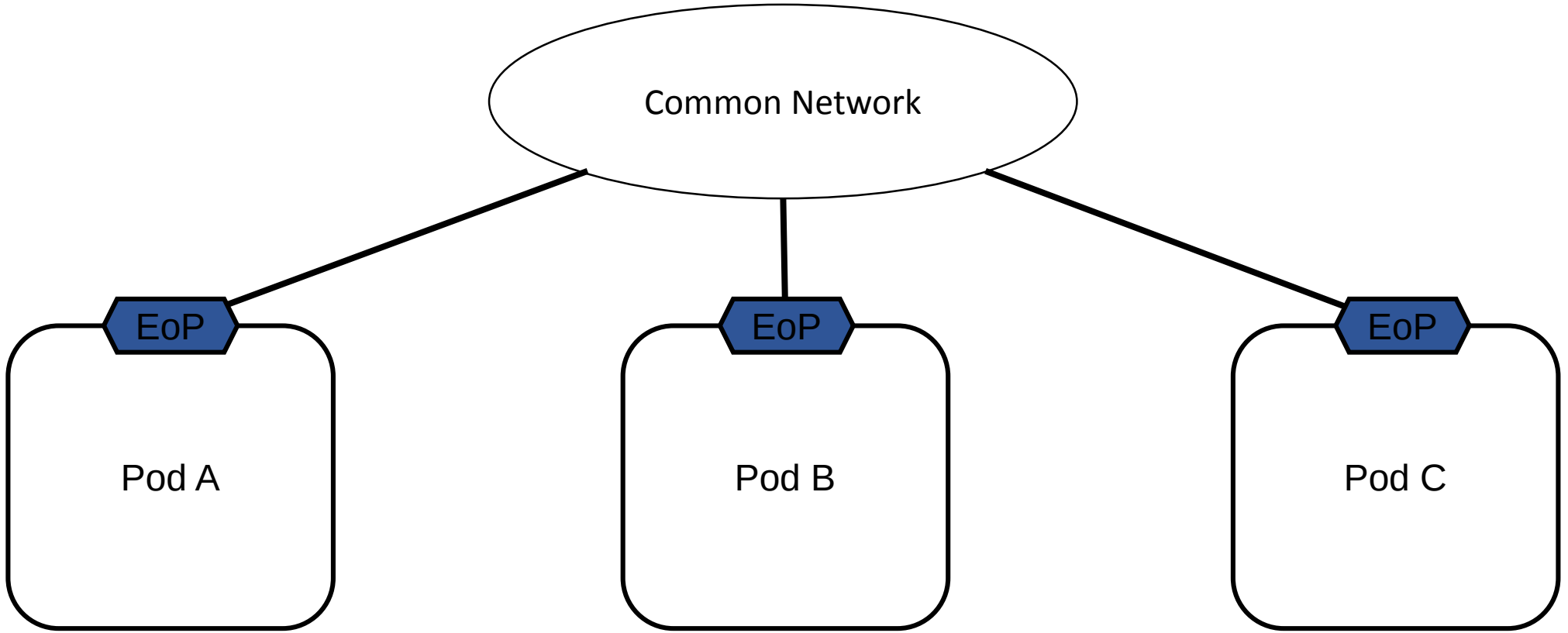


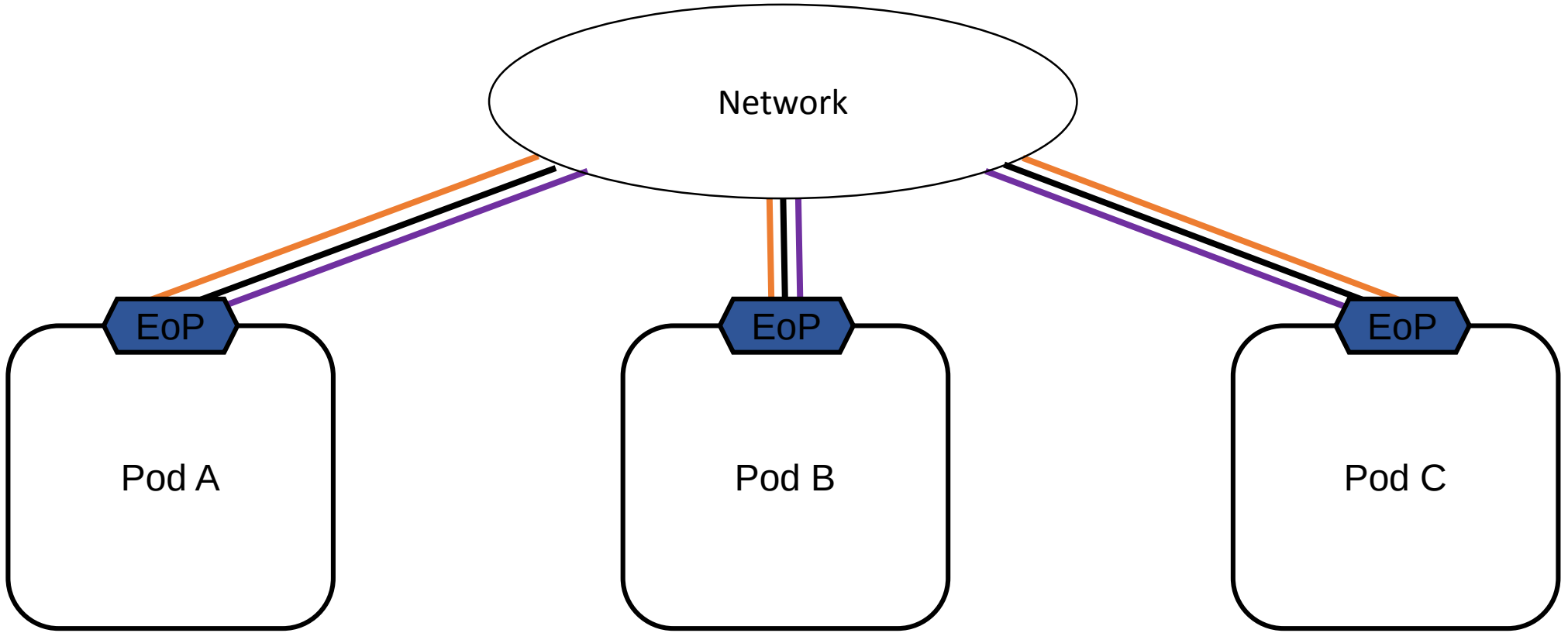


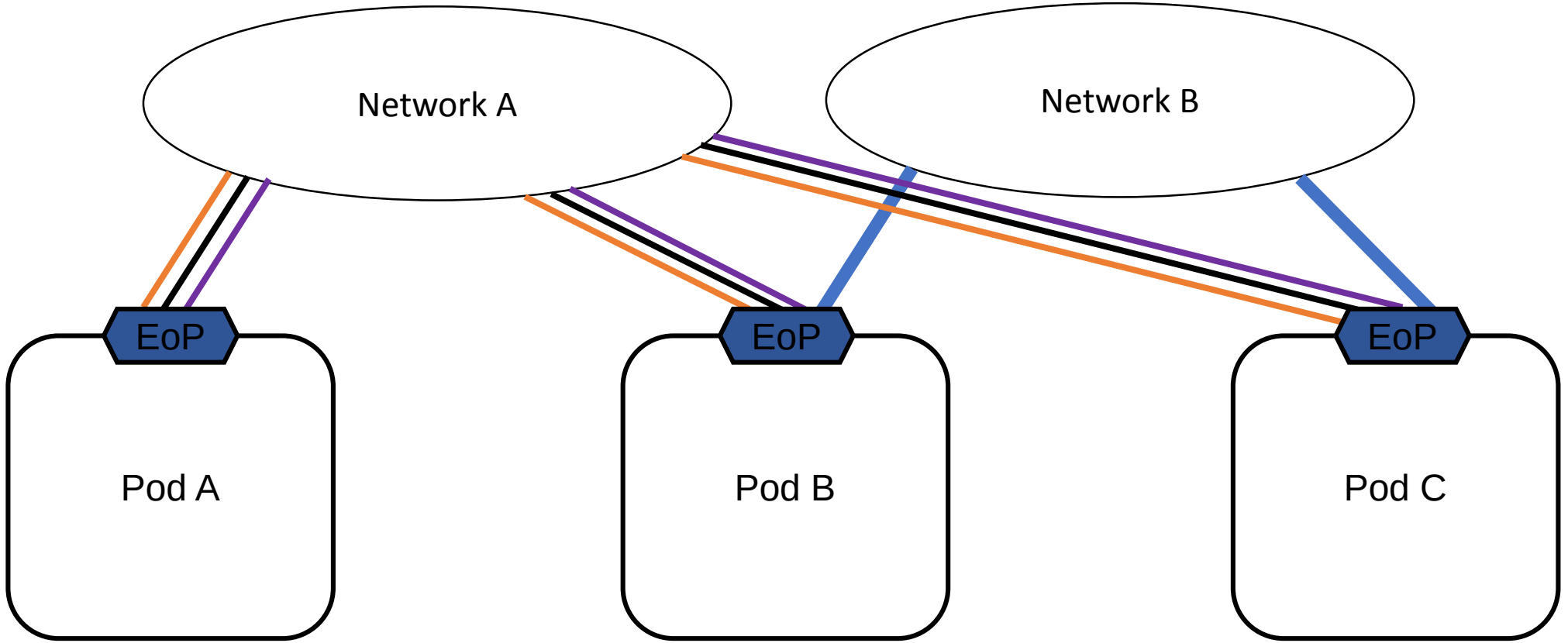
Facebook datacenter architecture

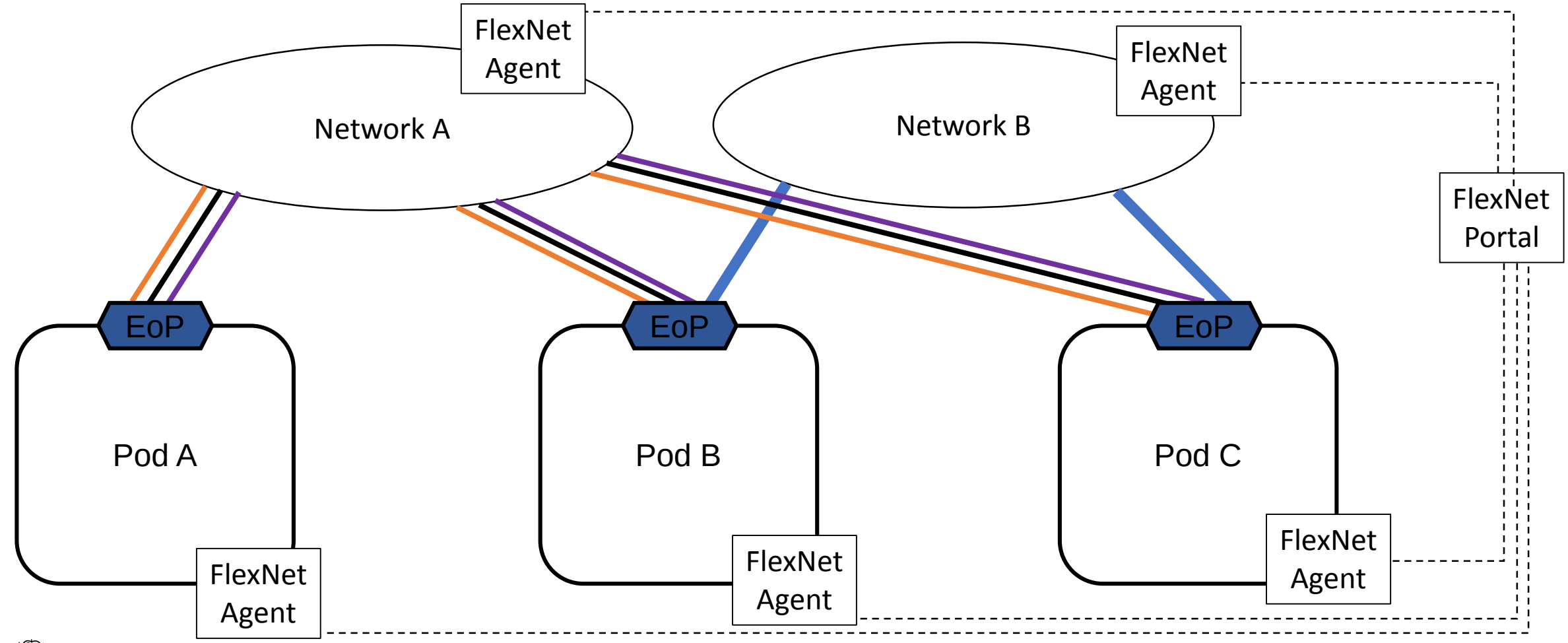
http://go.bigswitch.com/rs/974-WXR-561/images/Core-and-Pod%20Overview.pdf

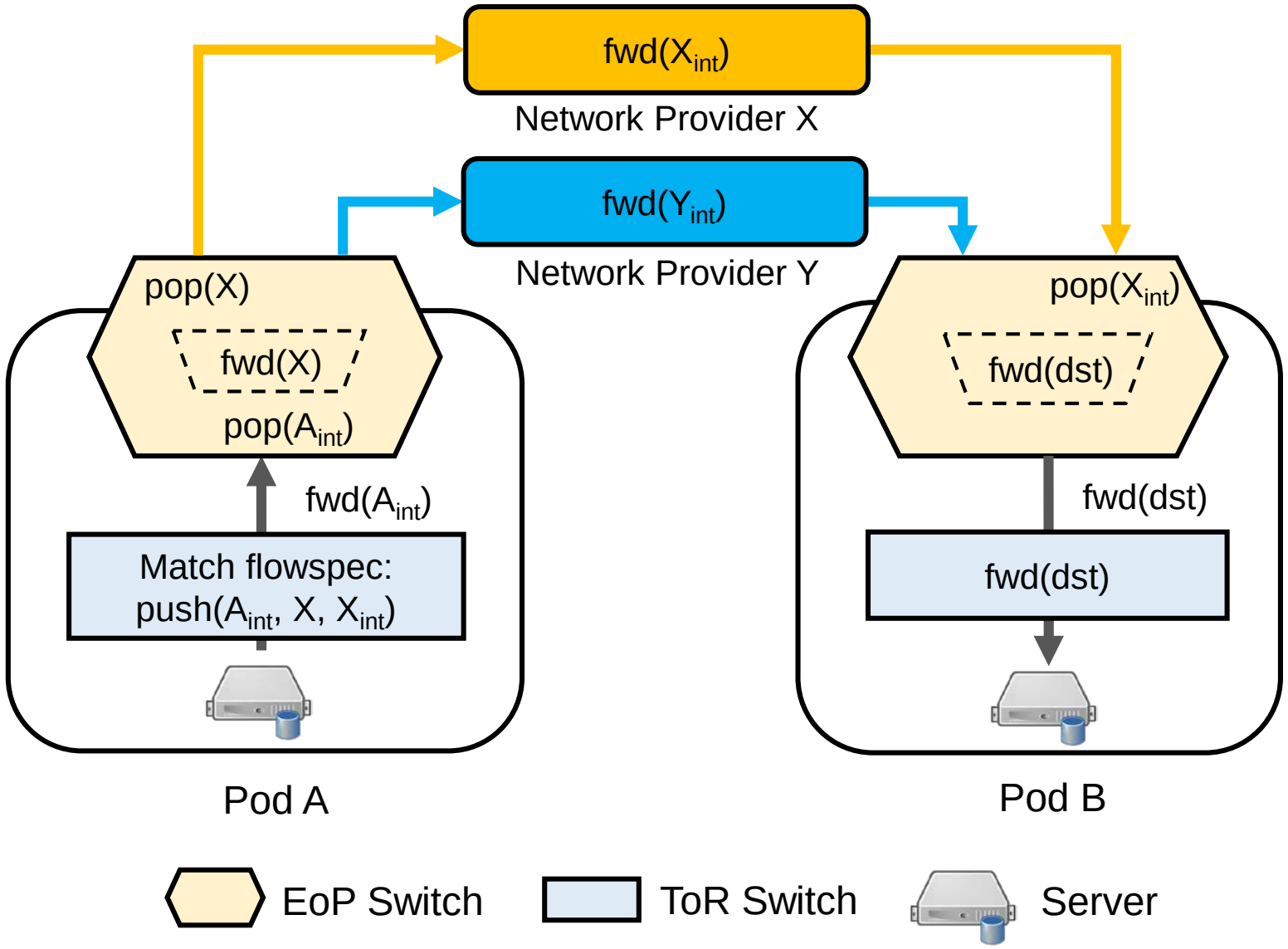

Common Network
Pod A
GPUs
Pod B
Storage
Pod C
Compute


Common Network
EoP
EoP
EoP
Pod A
Pod B
Pod C


Network
EoP
EoP
EoP
Pod A
Pod B
Pod C


Network A
Network B
EoP
EoP
EoP
Pod A
Pod B
Pod C


FlexNet Agent
Network A
FlexNet Agent
Network B
FlexNet Portal
EoP
EoP
EoP
Pod A
Pod B
Pod C
FlexNet Agent
FlexNet Agent
FlexNet Agent

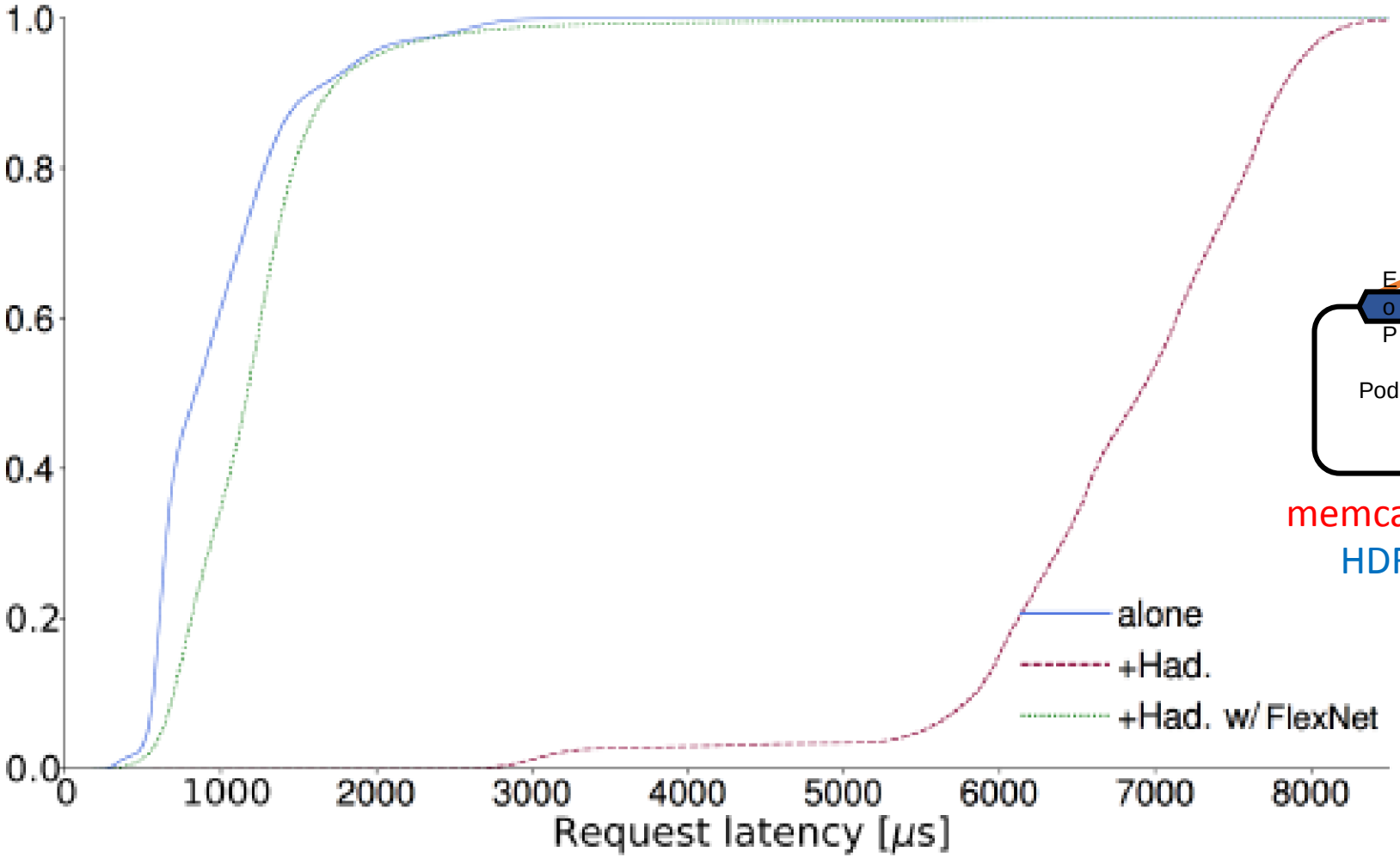

fwd(X_{int})
Network Provider X
fwd(Y_{int})
Network Provider Y
pop(X)
fwd(X)
pop(A_{int})
fwd(A_{int})
Match flowspec:
push(A_{int}, X, X_{int})
pop(X_{int})
fwd(dst)
fwd(dst)
fwd(dst)
Pod A
Pod B
EoP Switch
ToR Switch
Server

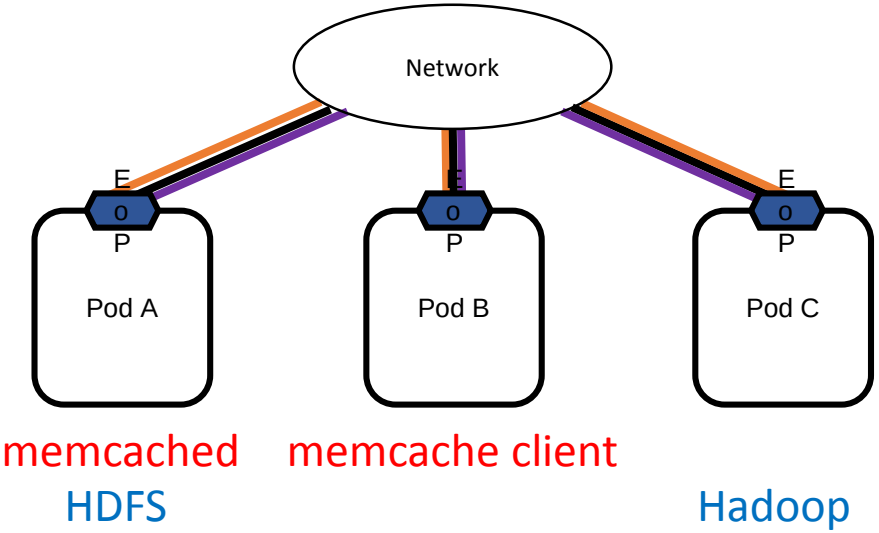
Current Prototype

- Complete end-to-end prototype
- Northbound portal in Python to negotiate paths
  - Configures tagging (pod), forwarding (EoP), paths (network)
- Commodity SDN switches + OpenVSwitch
  - VLANs at the pods
  - Nested MPLS to select Path

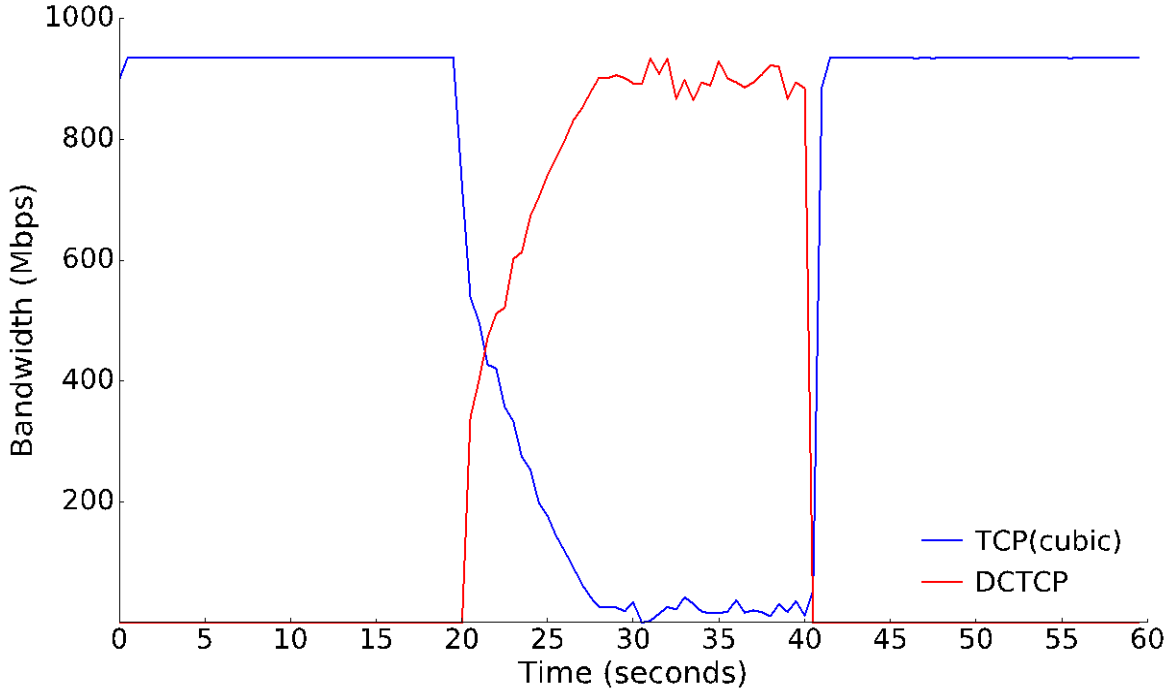
- Evaluation on a very small cluster
  - 3 pods, 4 physical SDN switches virtually split between EoP, Inter-Pod Networks, ToRs are ovs

Evaluation: Memcached + Hadoop

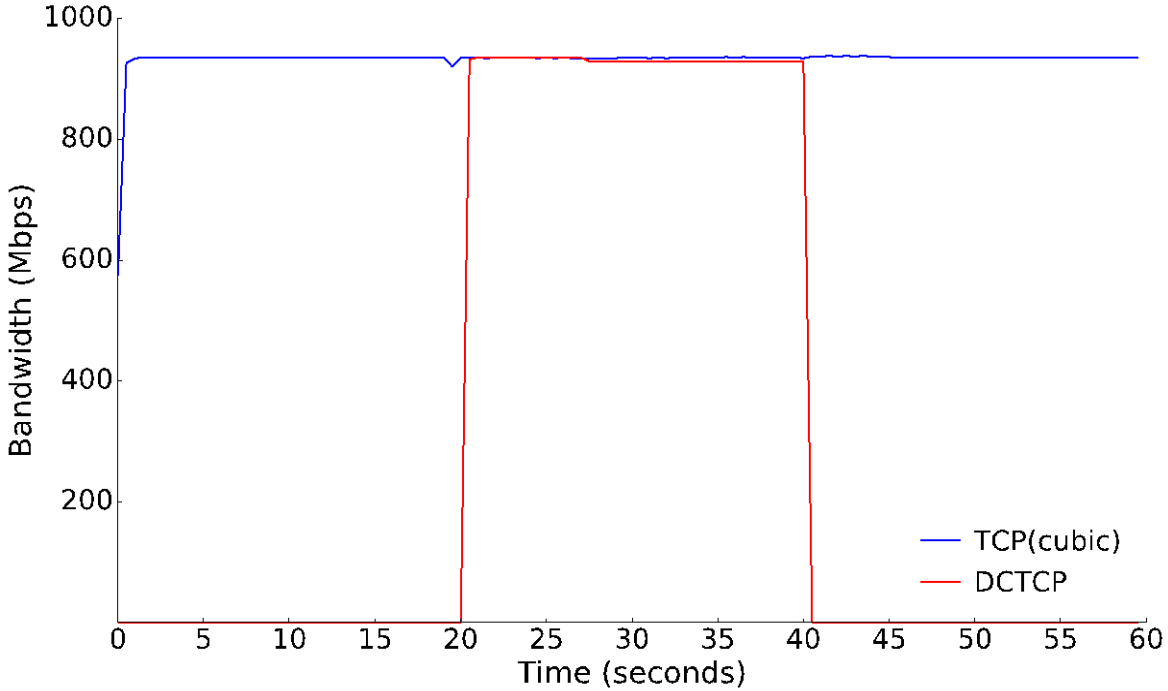




Evaluation: DCTCP vs Cubic

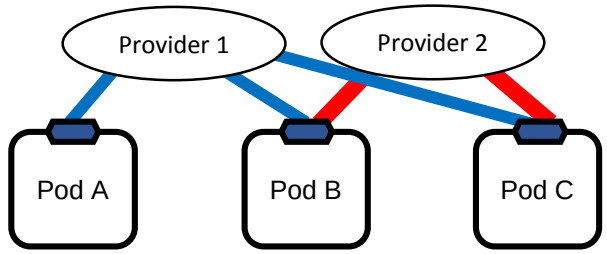


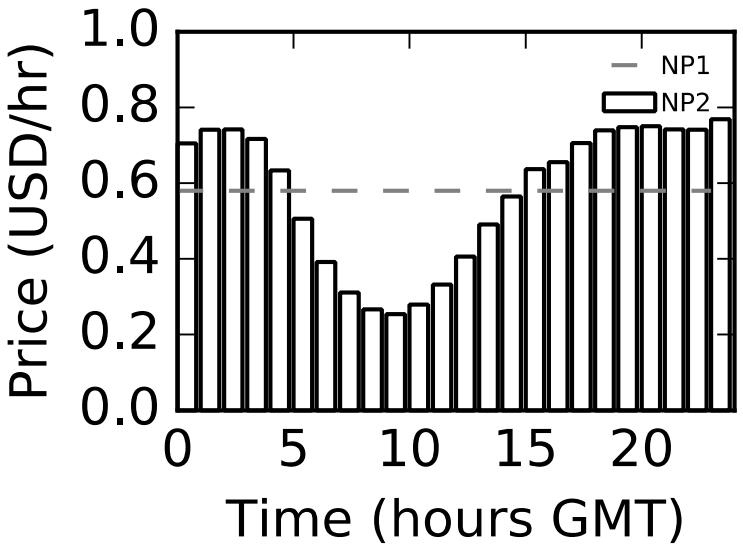
DCTCP sharing switches with Cubic

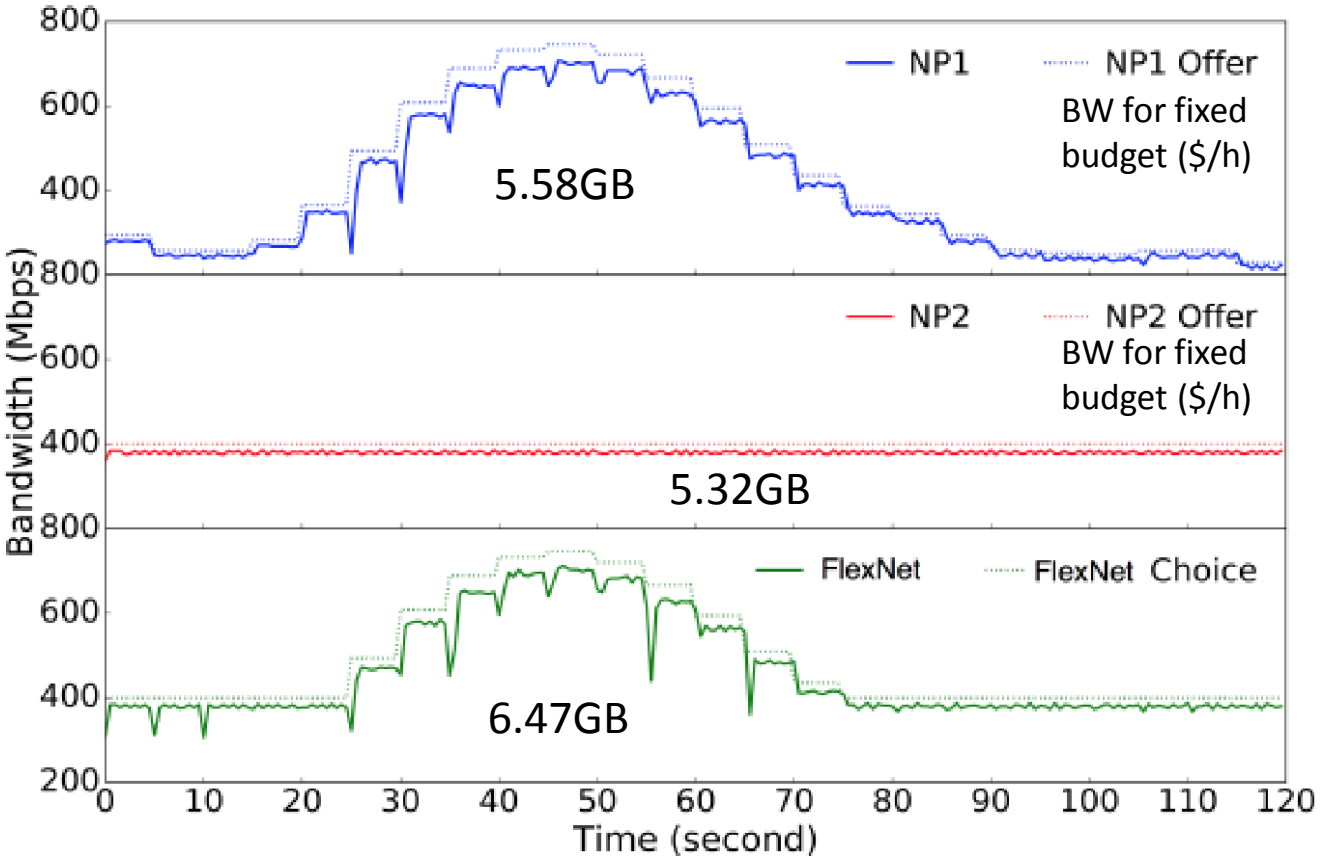


DCTCP using an exclusive path

Evaluation: Marketplace







Summary

- Flexnet: flowspec + tagging + diverse paths could break the conundrum
  - *Narrow waist: tagging*
- Better for tenants
  - Better guarantees, matching application properties
  - Better prices (?)
- Better for providers
  - More efficient use of resources
  - Deployment of niche technologies
  - Opportunity for monetization
  - Get more information from customers' demands
- Does this make sense to you? Or is today's situation good enough? ☺

Discussion

- More complex interface (vs. “don’t think about the network”)
  - Integrate with K8s provisioning / K8s Network Service Mesh?
- What to offer?
  - Measurable properties vs implementation specifics (what vs how)
- “Good enough”?
  - Azure Accelerated Networking (NSDI’18): 30+Gbps, < 15us, “free”
  - *We are not restricted to performance properties*
- Use cases
  - It would be great to get real use cases from MOC users!

Thank you!

rfonseca@cs.brown.edu

vs. Kubernetes Network Service Mesh

- Similar goals: different properties for network flows
- Example use cases:
  - VPN gateway to corporate office
  - Direct connection to ISP/Telco/Private link
  - Distributed virtual bridges
  - NFV Chaining
    - Security, IDS, VPN
  - Guaranteed latency/bandwidth
  - Load balancing