Datacenters

CS 168 – Fall 2024

Agenda

- Logistics
- Data Centers
  - Fat Tree
  - Clos Topology
- ECMP
- Over/Underlay

Logistics

- HW4 (Applications & E2E)
  - released on Gradescope
  - due 11/18 (Mon)
- HW5 (Datacenters)
  - will be released on Gradescope (will be short)
  - due 11/25 (Mon)
- NO Discussion next week!
- NO office hours next week!
- Interested in Networking research?

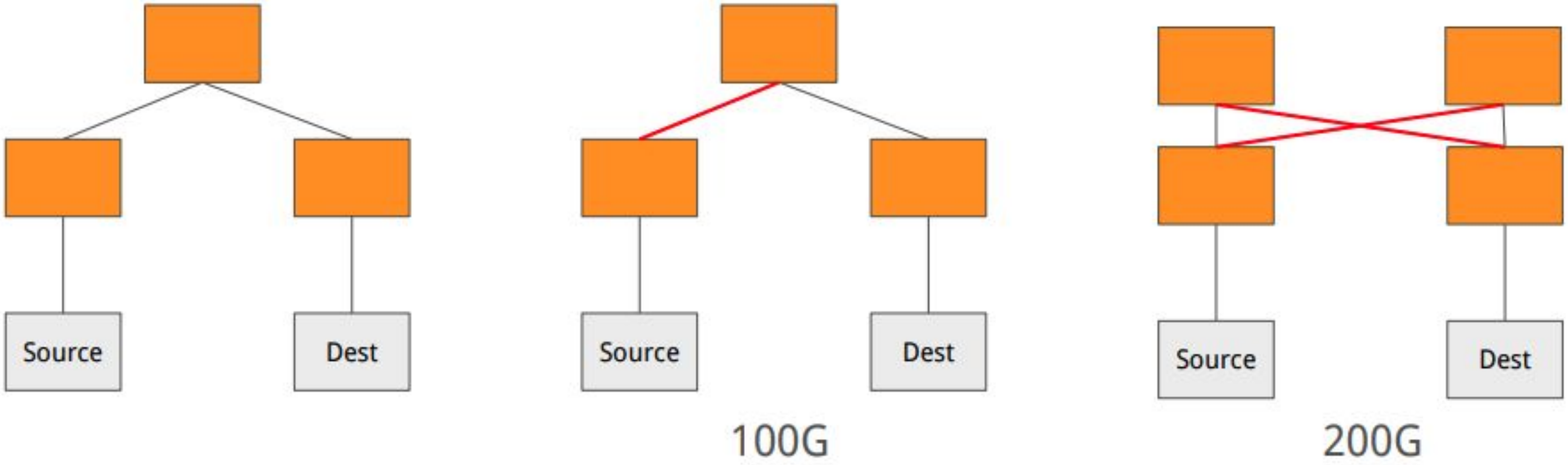
Datacenters

- How does one design datacenter networks?
- Consider new assumptions (i.e., different from what we learned in Internet).
    - **Single administrative control** over the network topology, traffic, and to some degree end hosts
    - Much more **homogenous**
    - Strong emphasis on **performance**
    - Few backwards compatibility requirements, **clean-slate solutions** welcome!
    - ...

We will focus on Data Center topology in the rest of the discussion

Bisection Bandwidth

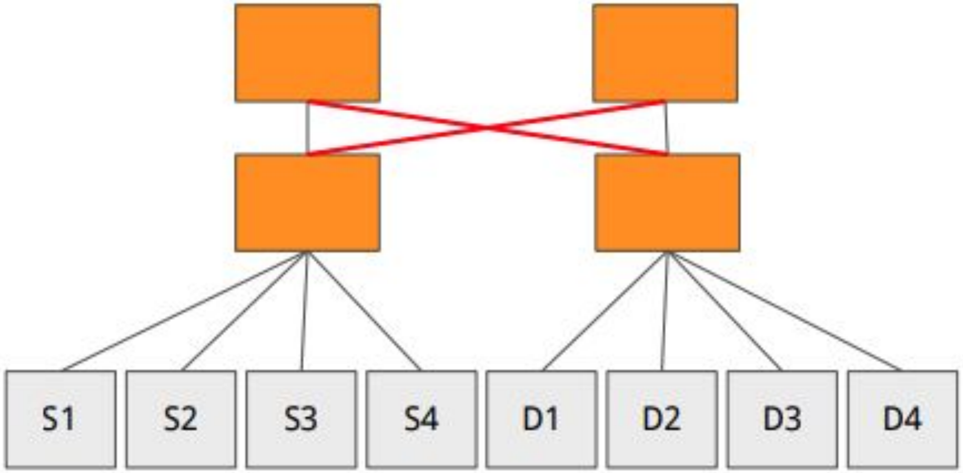
- We want a network with high bisection bandwidth:
  - Pick the number of links we must cut in order to partition a network into two equal halves
  - Bisection bandwidth is the sum of those bandwidths.



Bisection Bandwidth

- **Full bisection bandwidth**: Nodes in one partition can communicate simultaneously with nodes in the other partition at full rate.
    - Given N nodes, each with access link capacity R, bisection bandwidth = N/2 x R
- **Oversubscription**, informally, how far from the full bisection bandwidth we are
    - Formally: ratio of worst-case achievable bandwidth to full bisection bandwidth.

Bisection Bandwidth



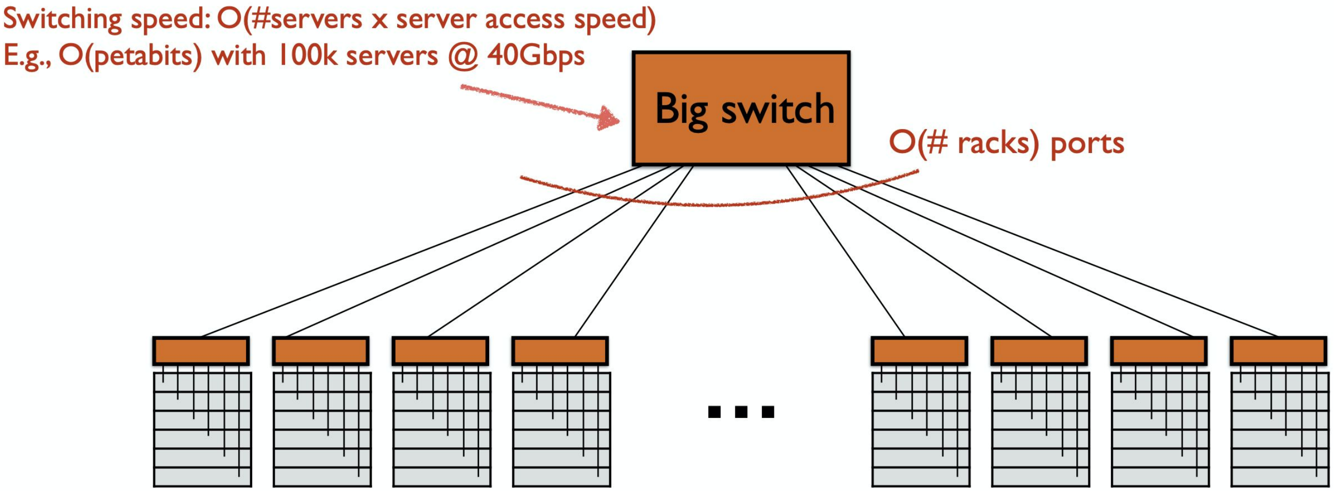
Bisection Bandwidth: 200G

Full Bisection Bandwidth: (8/2)*100G = 400G

Oversubscription: 200/400 = 2x

Big switch abstraction

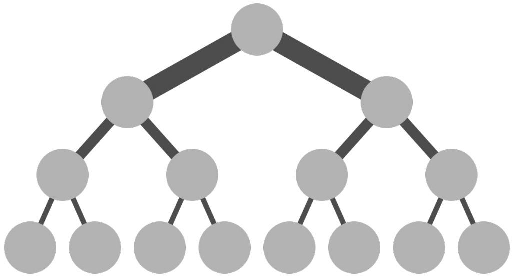
- We want an abstraction of a big switch
- Naive solutions:
  - Server-to-server full-mesh
  - A physical big switch?
- Either impractical or too costly ($$$); 10k hosts * 10k hosts = ~100M links
- Can we do better?


Switching speed: O(#servers x server access speed)
E.g., O(petabits) with 100k servers @ 40Gbps
Big switch
O(# racks) ports
...

Design 1: Fat tree (scale-up)

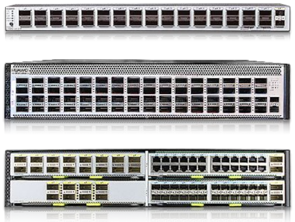
- Borrowed straight from the High Performance Computing (i.e., supercomputers) community

- Use of big, non-commodity switches



- Problem? Scales badly in terms of cost
  - ..only a few switch vendors can make big switches

- Scales badly in terms of fault-tolerance

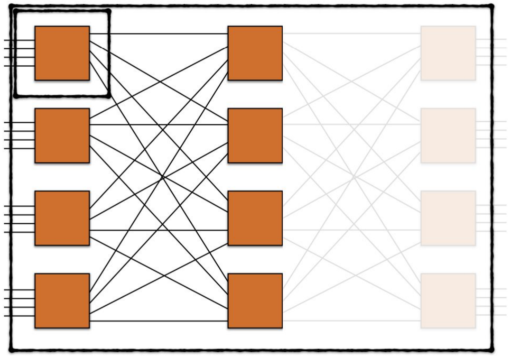


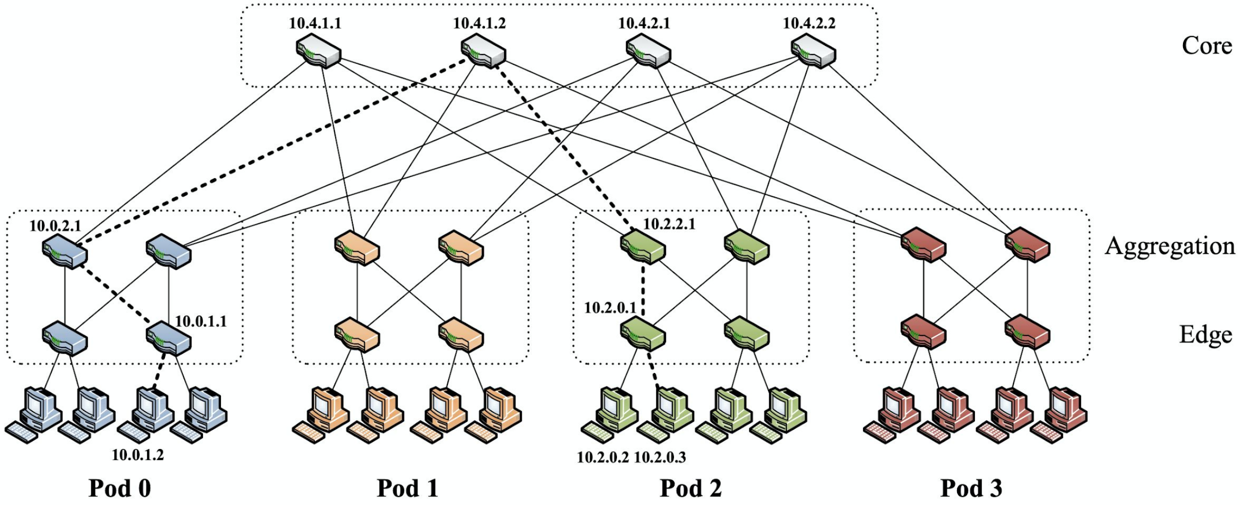
small switch

big switch

Design 2: Clos (scale-out)

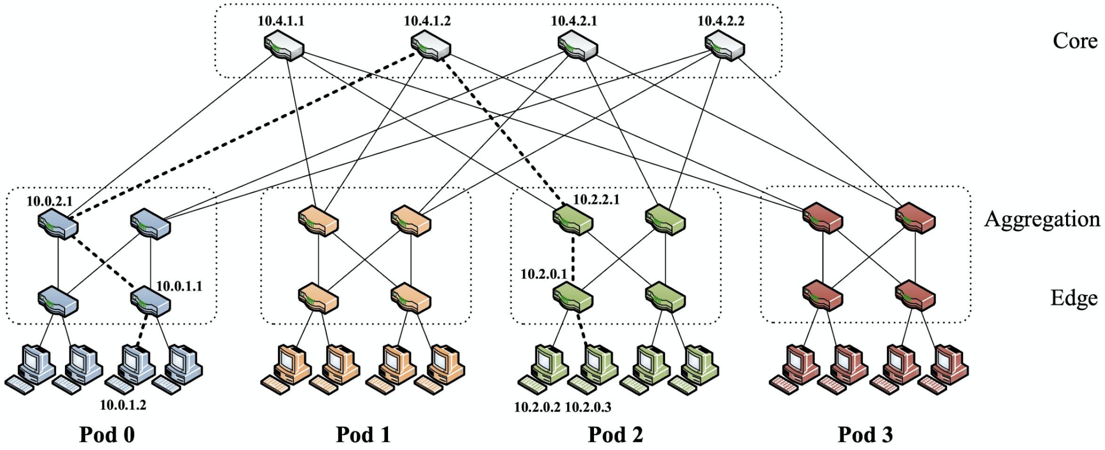
- Replace nodes in the fat tree with groups of cheap commodity switches
  - cheaper
  - high redundancy (bisection width)
- Allows *oversubscription* ratio of 1





A Closer look at Clos

- The concept of redundancy and the clos topology itself have many variations!
- We focus on the 3-tiered clos topology
- Homogenous switches (k ports) and links
- Each non-core switch has k/2 ports pointing north and k/2 pointing south



Caveats about clos

- Oversubscription ratio of 1 only with optimal load balancing
- Non-trivial to incrementally build and/or expand the network
    - e.g., port count k is fixed

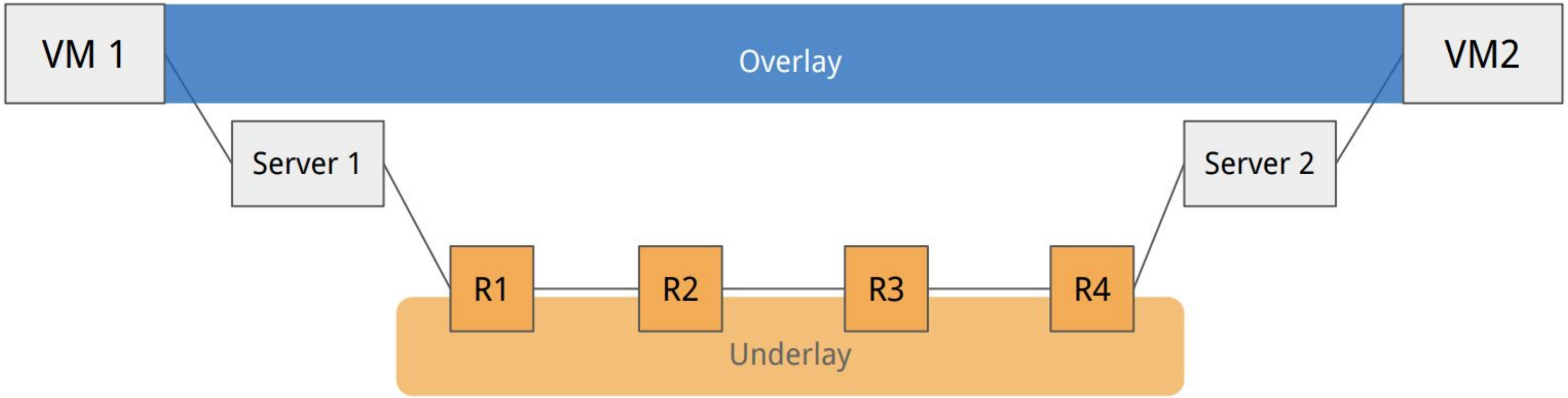
ECMP

- **ECMP - Equal Cost Multi-Path**
  - Goal: use multiple paths in network topology that are equal cost
  - Idea: Load-balance packets across different forwarding paths

- **ECMP Hash Function how to load balance packets:**
  - f(src_ip, dst_ip, protocol, src_port, dst_port)
  - “Per-flow” load balancing
    - packets belonging to the same "flow"are always routed along the same path across multiple available links

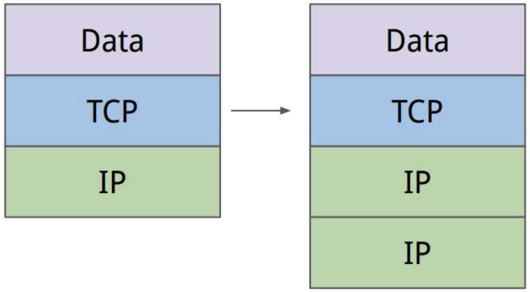
(Over/Under)lay

- Many hosts spin up (boot up) Virtual machines frequently
- Addressing could become a mess!
  - And won't scale!
- Thus, we build **overlay and underlay** networks

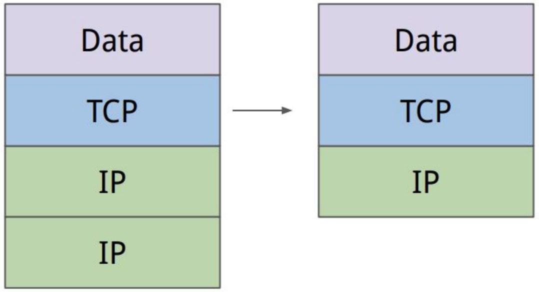


How? Encapsulation

- Encapsulation: put another header on the packet

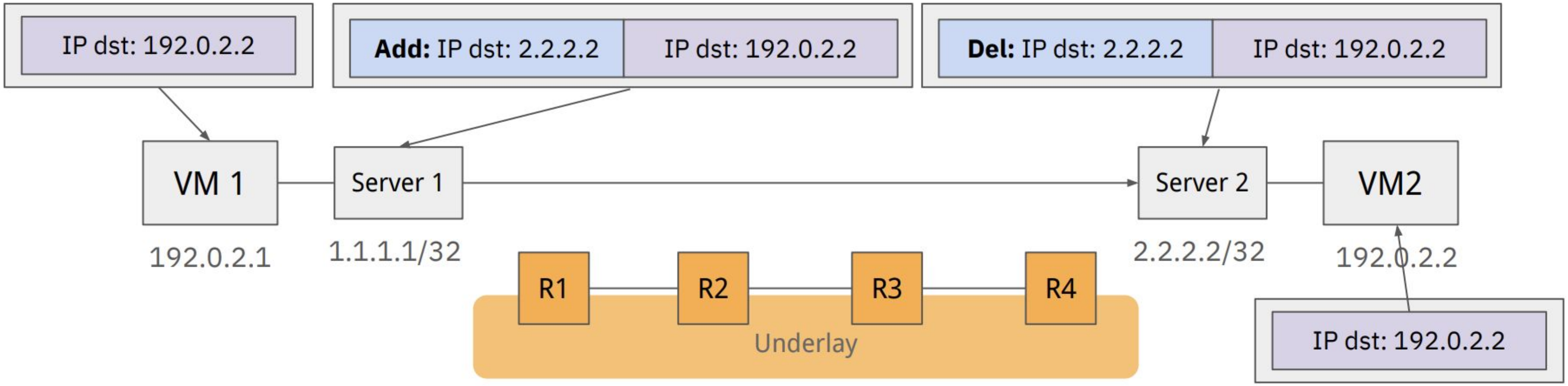


- Decapsulation: remove extra headers that were added for encapsulation



Example of Encapsulation

Encapsulating Packets

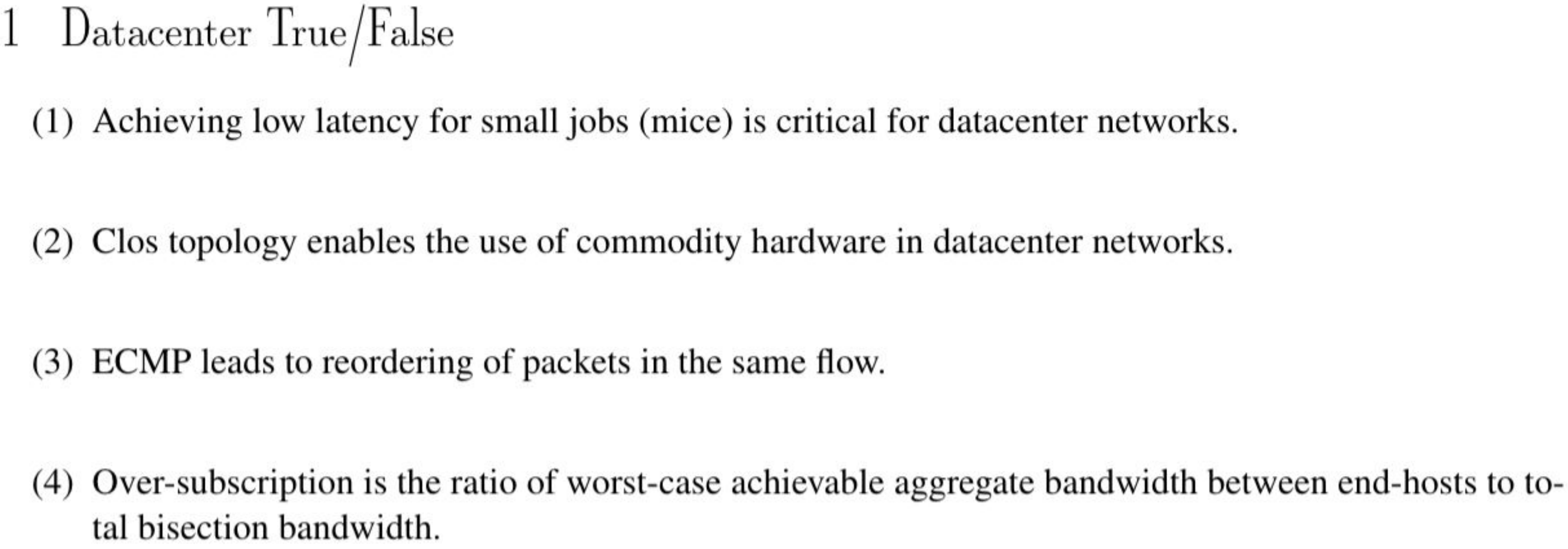


Using an *encapsulation* allows us to hide the "inner" overlay packet from the underlay.

source: lecture 20, slide 55

Feedback Form: https://tinyurl.com/cs168-disc-fa24

Worksheet - Q1



1 Datacenter True/False

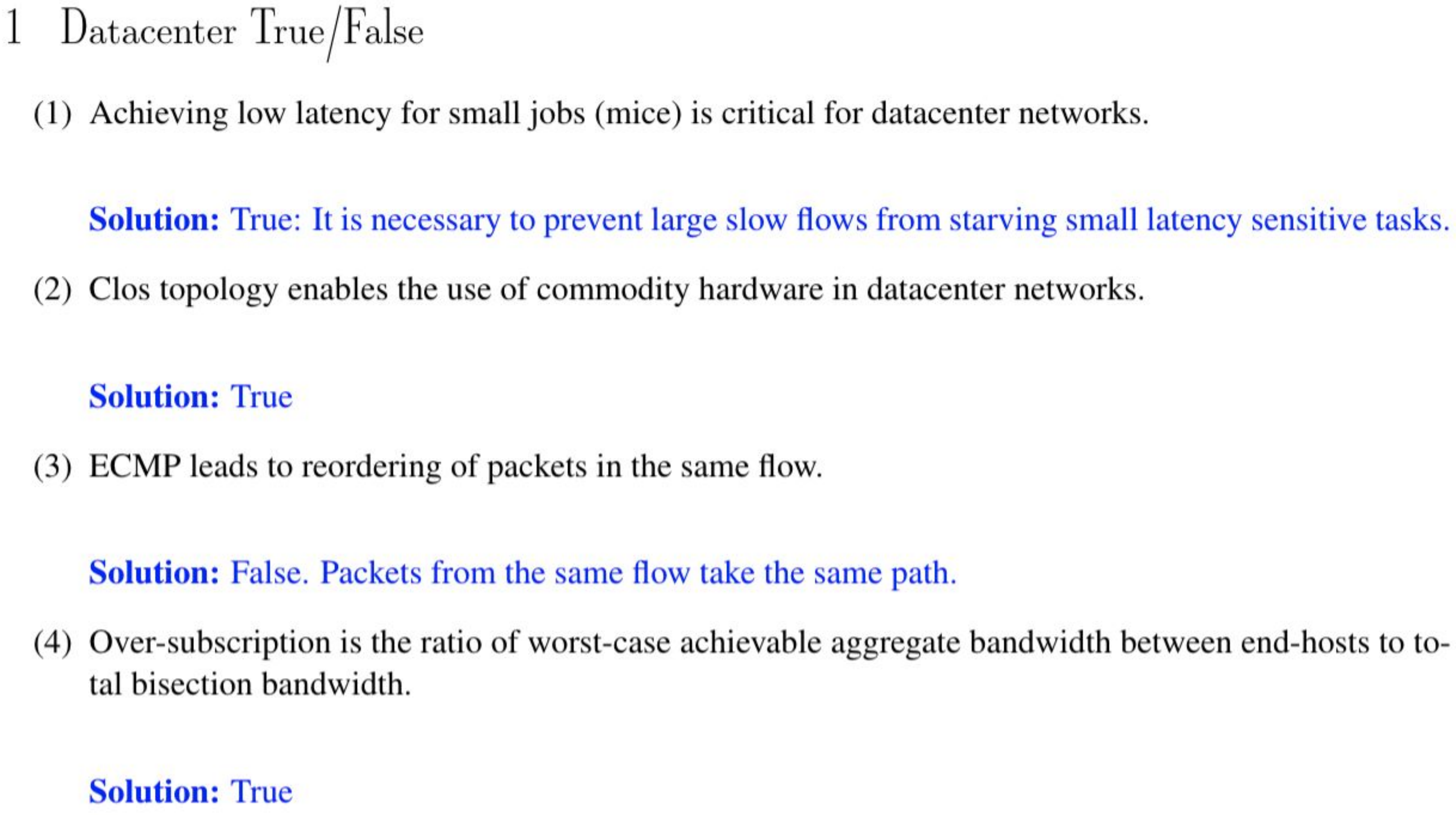
(1) Achieving low latency for small jobs (mice) is critical for datacenter networks.

(2) Clos topology enables the use of commodity hardware in datacenter networks.

(3) ECMP leads to reordering of packets in the same flow.

(4) Over-subscription is the ratio of worst-case achievable aggregate bandwidth between end-hosts to total bisection bandwidth.

Worksheet - Q1



1 Datacenter True/False

(1) Achieving low latency for small jobs (mice) is critical for datacenter networks.

Solution: True: It is necessary to prevent large slow flows from starving small latency sensitive tasks.

(2) Clos topology enables the use of commodity hardware in datacenter networks.

Solution: True

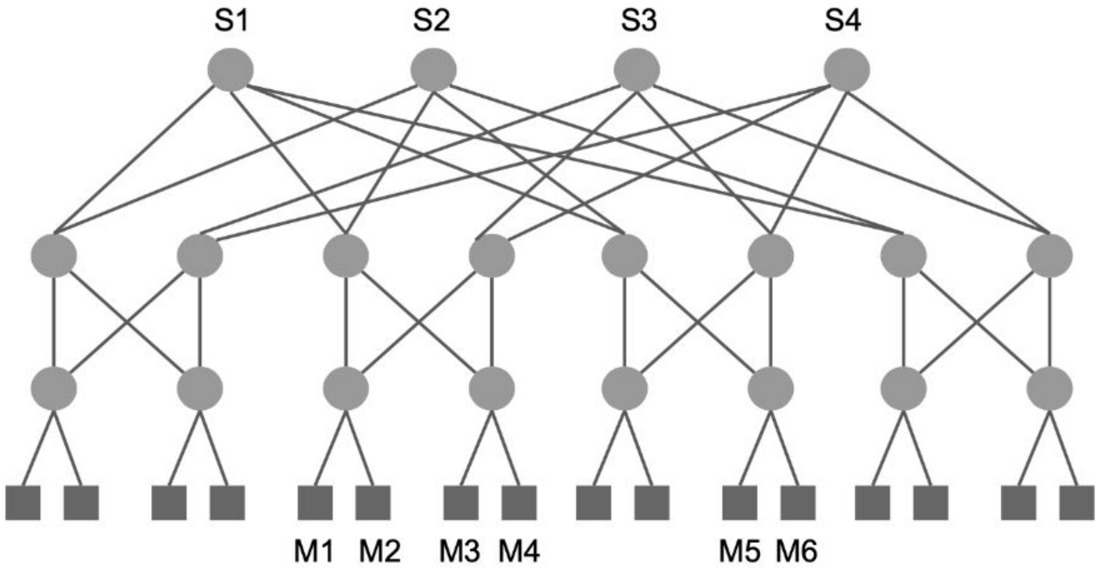
(3) ECMP leads to reordering of packets in the same flow.

Solution: False. Packets from the same flow take the same path.

(4) Over-subscription is the ratio of worst-case achievable aggregate bandwidth between end-hosts to total bisection bandwidth.

Solution: True

2 Clos-based Topology



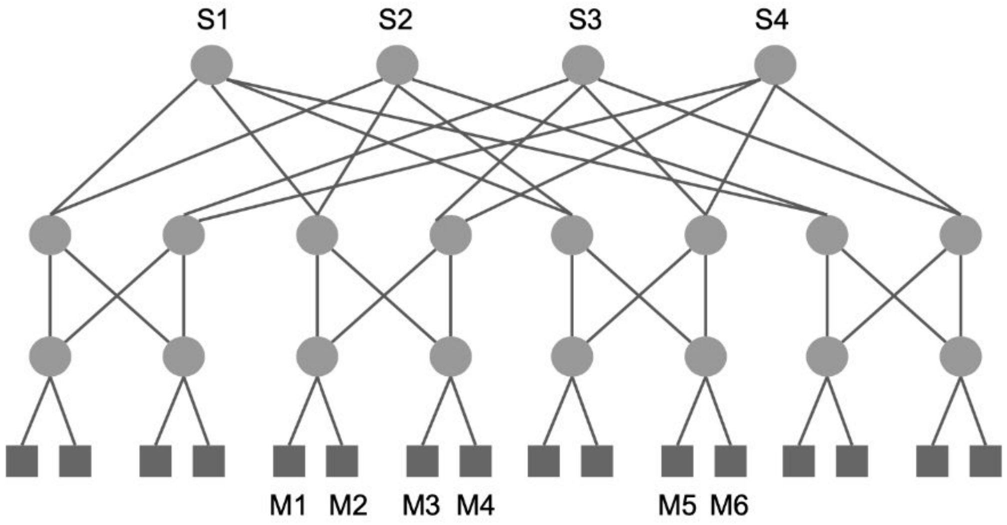
Given the datacenter network above:

(1) How many paths from an arbitrary server upward to an arbitrary root switch? Consider only minimal-length paths.

(2) How many paths can be set up between M1 to M3 in this topology? M1 to M5? Consider only minimal-length paths.

(3) How many of the M1 to M5 paths go through switch S2?

2 Clos-based Topology



Given the datacenter network above:

(1) How many paths from an arbitrary server upward to an arbitrary root switch? Consider only minimal-length paths.

Solution: 1.

(2) How many paths can be set up between M1 to M3 in this topology? M1 to M5? Consider only minimal-length paths.

Solution: 2 ; 4.

(3) How many of the M1 to M5 paths go through switch S2?

Solution: 1.

3 Bisection Bandwidth and ECMP

Consider the same network topology in Q2. Assume all links have capacity 10 Gbps and ECMP is used for multipath routing.

(1) What is the bisection bandwidth of this topology?

(2) Assume ECMP makes the best possible hashing/load balancing. If switch S1 and S3 fail, what is the available aggregate bandwidth from each four-server Pod to any other such Pod (a Pod contains four servers closest to each other)?

(3) Assume M1, M2 are sending flows to M5; and M3, M4 are sending flows to M6. TCP makes optimal use of the available bandwidth and ECMP makes the worst possible hashing. What is the approximate average data rate server M1 can send to server M5?

3 Bisection Bandwidth and ECMP

Consider the same network topology in Q2. Assume all links have capacity 10 Gbps and ECMP is used for multipath routing.

(1) What is the bisection bandwidth of this topology?

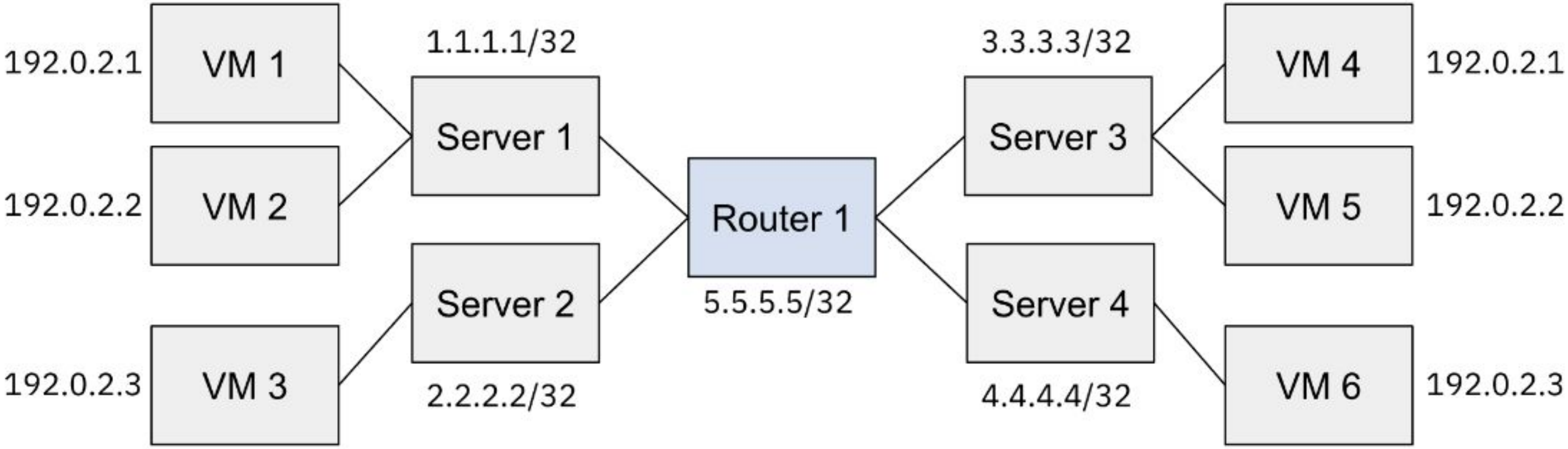
Solution: 80 Gbps.

(2) Assume ECMP makes the best possible hashing/load balancing. If switch S1 and S3 fail, what is the available aggregate bandwidth from each four-server Pod to any other such Pod (a Pod contains four servers closest to each other)?

Solution: 20 Gbps.

(3) Assume M1, M2 are sending flows to M5; and M3, M4 are sending flows to M6. TCP makes optimal use of the available bandwidth and ECMP makes the worst possible hashing. What is the approximate average data rate server M1 can send to server M5?

Solution: 2.5 Gbps.


192.0.2.1
VM 1
192.0.2.2
VM 2
192.0.2.3
VM 3
1.1.1.1/32
Server 1
Server 2
2.2.2.2/32
Router 1
5.5.5.5/32
3.3.3.3/32
Server 3
Server 4
4.4.4.4/32
VM 4
192.0.2.1
VM 5
192.0.2.2
VM 6
192.0.2.3

(1) If routing a packet from VM 1 to VM 6, what is the expected path?

(2) What encapsulation and decapsulation should the servers/router on the path from VM 1 to VM 6 perform (if any)? Fill in the following table with the destination address (both overlay and underlay) of the packets at each step.

Server/Router	Adds/Deletes Header	Overlay Address	Underlay Address

(3) How many destinations does Router 1 have to keep track of without encapsulation? How about with encapsulation? Consider how this isolation impacts VM scalability in a data center!

(1) If routing a packet from VM 1 to VM 6, what is the expected path?

Solution: S1-R1-S4

(2) What encapsulation and decapsulation should the servers/router on the path from VM 1 to VM 6 perform (if any)? Fill in the following table with the destination address (both overlay and underlay) of the packets at each step.

Server/Router	Adds/Deletes Header	Overlay Address	Underlay Address

Solution:

Server	Adds/Deletes Header	Overlay Address	Underlay Address
Server 1	Adds	192.0.2.3	4.4.4.4/32
Router 1	Neither	192.0.2.3	4.4.4.4/32
Server 4	Deletes	192.0.2.3	4.4.4.4/32

(3) How many destinations does Router 1 have to keep track of without encapsulation? How about with encapsulation? Consider how this isolation impacts VM scalability in a data center!

Solution: Without = all of them = 10, With = just the underlay / physical addresses = 4