Modularized Data Center Cube

Presented by:
Mark Overholt & Shiguang Wang

Outline

- The Cost of a DC
- BCube: one way to implement MDC
- MDCube
- Discussion & References

Where costs go in the DC

Amortized Cost	Component	Sub-Components
~45%	Servers	CPU, memory, storage systems
~25%	Infrastructure	Power distribution and cooling
~15%	Power draw	Electrical utility costs
~15%	Network	Links, transit, equipment

Greatest DC costs go to servers

- Example:
- Assuming 50,000 servers, $3,000 per server, a 5% cost of money, and a 3 year amortization, the amortized cost of servers comes to $52.5 million per year.
- (50,000 * 3000 * 1.05 /3 = 52.5 M)
- Achieving high utilization is an important goal
  - *i.e. useful work accomplished per dollar invested*
- Unfortunately… remarkably low utilization

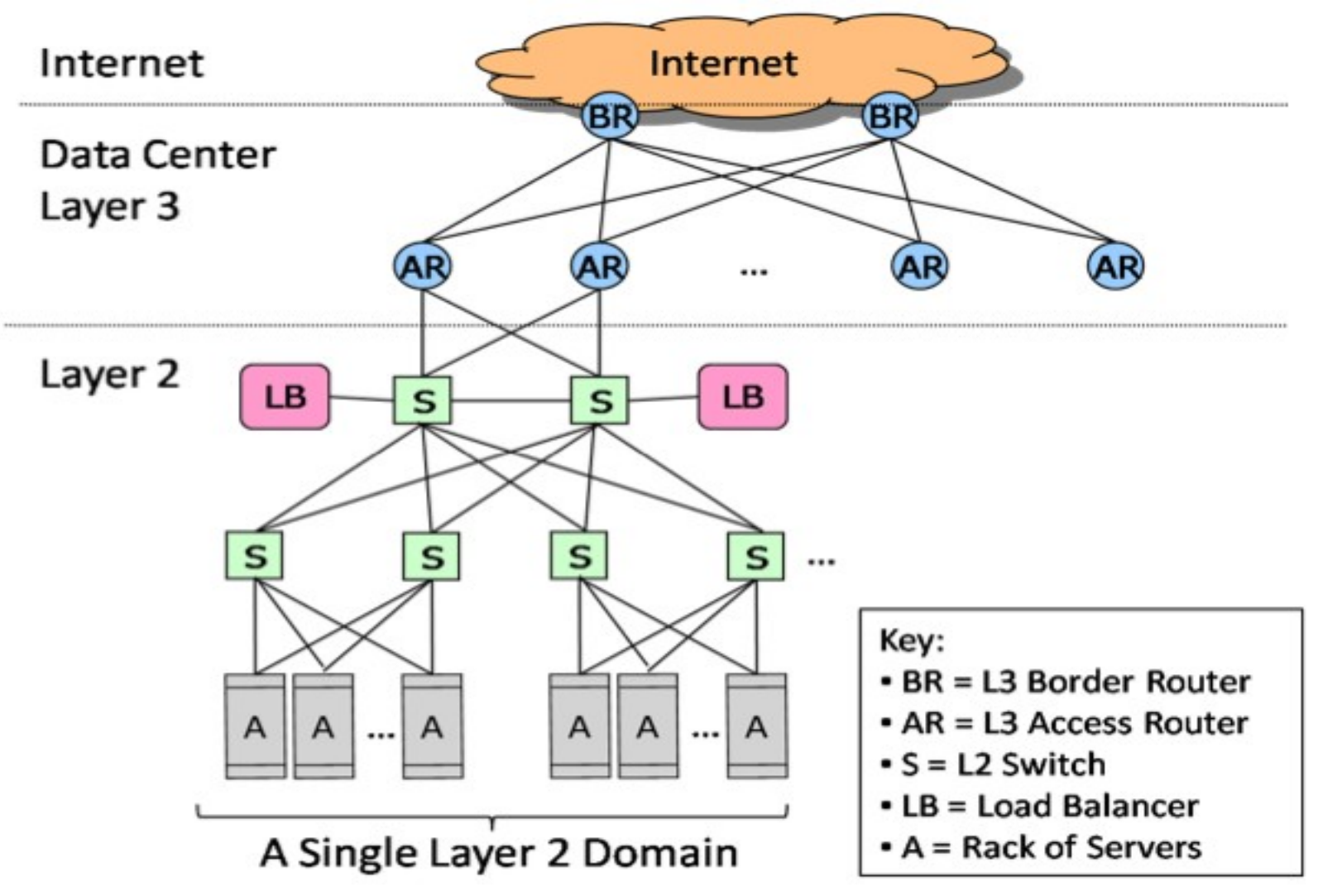
  E.g. 10%

A Possible Solution is improve Agility

- Agility:
  - the ability to dynamically grow and shrink resources to meet demand and to draw those resources from the most optimal location

  - *Conventional DC structure is not agile*

Conventional network architecture for a DC



Drawbacks of Conventional Network Architecture

- Servers dedicated to the application
- Fragmentation of resources
- Poor server to server connectivity

Several proposals to improve agility

- Fat tree, VL2
- BCube, MDCube
- Jellyfish
- …

BCube

- A new network architecture designed for shipping-container based, modular data centers.

MDC is possible

Figure 1: 20-foot ISO 668 Shipping Container

Figure 3: Rackable Systems Data Center in a Box

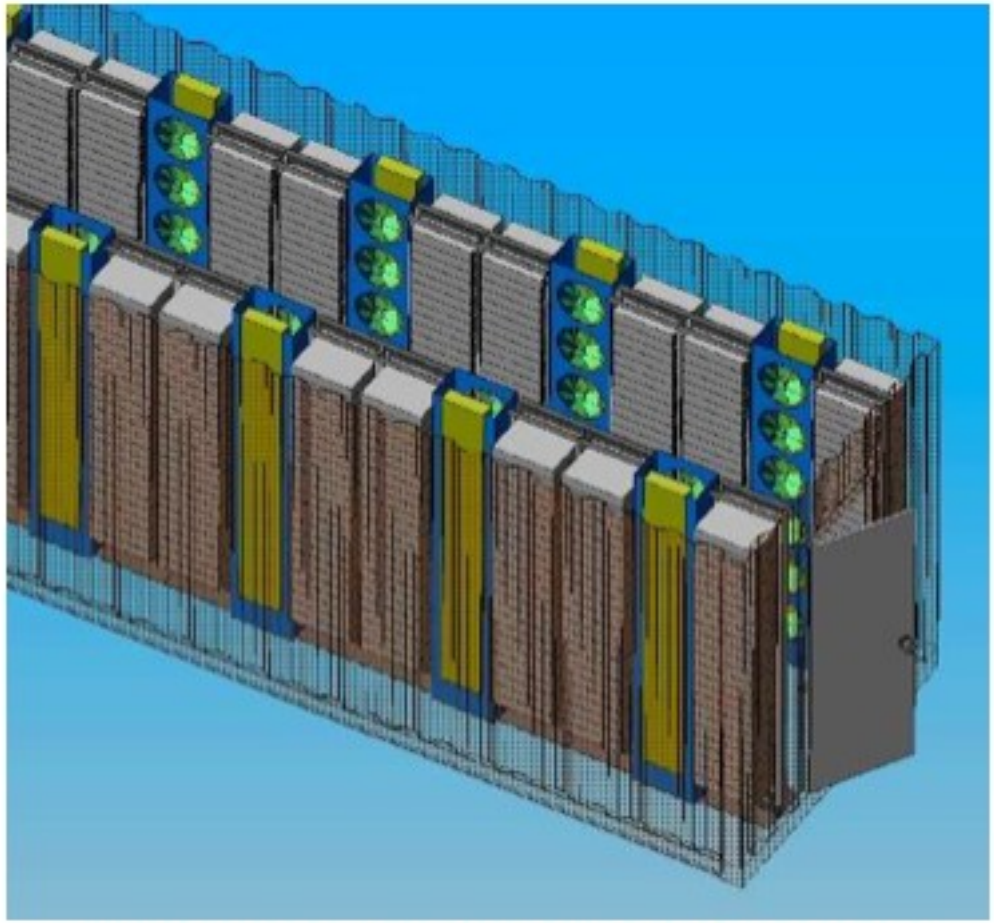


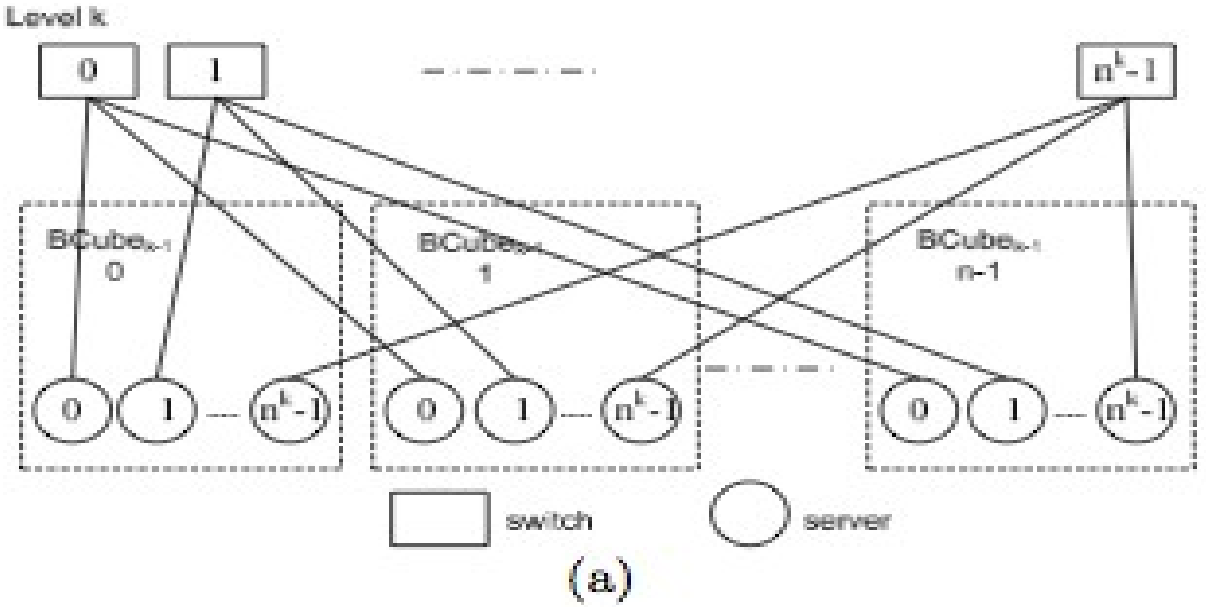
Figure 4: Rackable Systems Container Cooling Design

Figure 5: Sun Microsystems Black Box

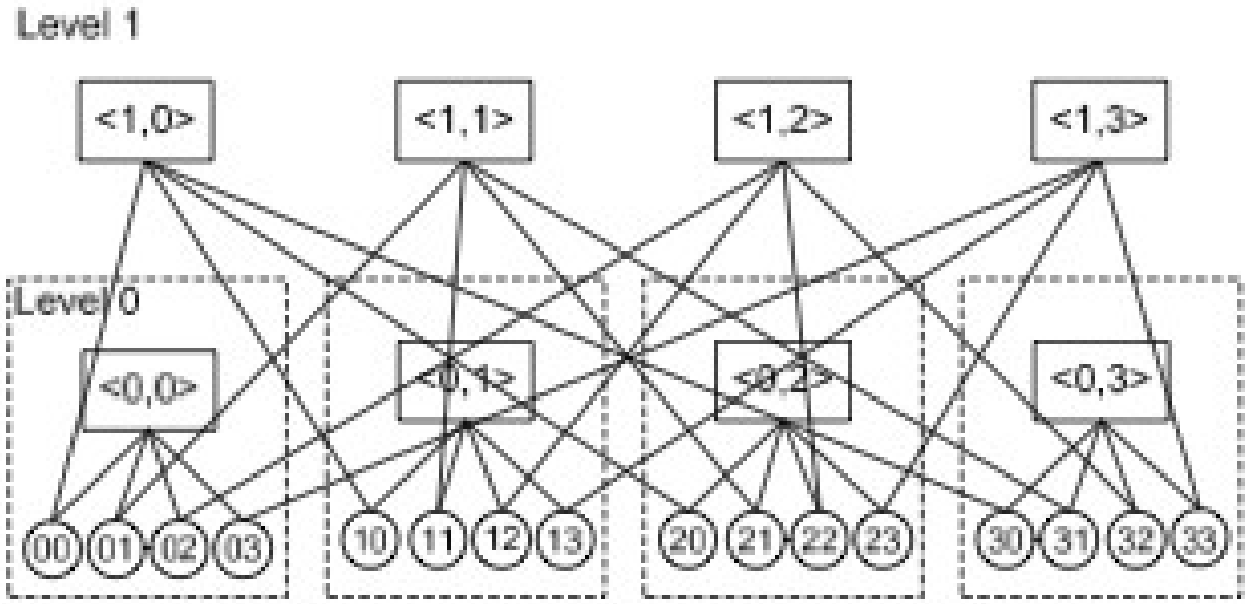
BCube

- Two types of devices
  1. Server w\ multiple ports
  2. Switches w\ constant ports

- Recursively defined structures
  - BCube(0) : n servers connecting to an n-port switch
  - BCube(k) : n BCube(k-1) and n^k n-port switches
  - (white board illustration)


Level k
0
1
n^k-1
$BCube_{k-1}$ 0
$BCube_{k-1}$ 1
$BCube_{k-1}$ n-1
0
1
n^k-1
switch
server

(a)


Level 1
<1,0>
<1,1>
<1,2>
<1,3>
Level 0
<0,0>
<0,1>
<0,2>
<0,3>
00
01
02
03
10
11
12
13
20
21
22
23
30
31
32
33

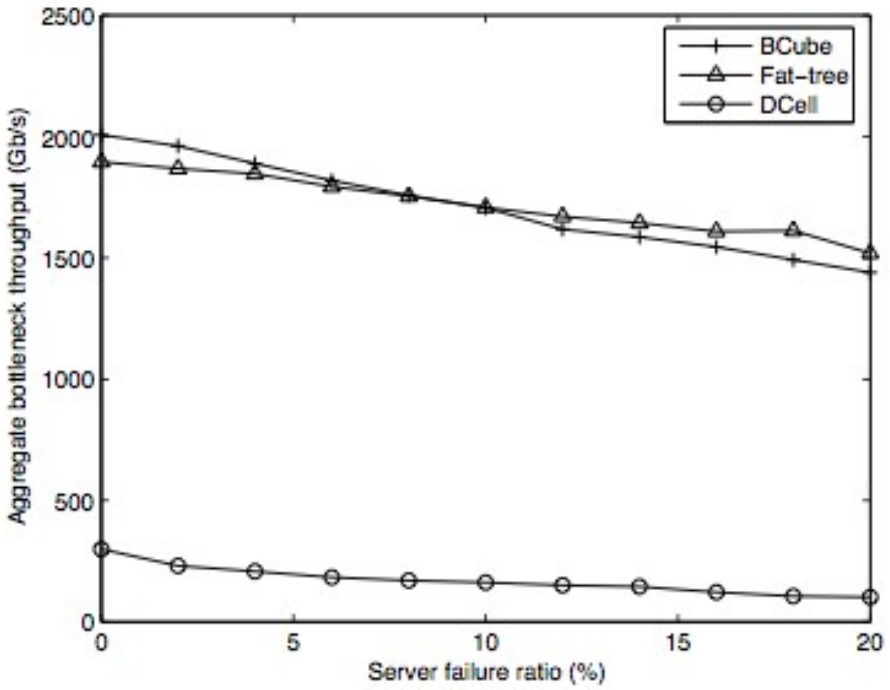
Properties of BCube

1. There are k+1 node disjoint paths btw any two servers in Bcube(k)
2. For any (<=k+2) servers, we can build edge-disjoint complete graphs
3. In Bcube(k), we can construct (k+1) edge-disjoint server spanning trees
4. The ABT for a Bcube network under the all-to-all traffic model is n(N-1)/(n-1)

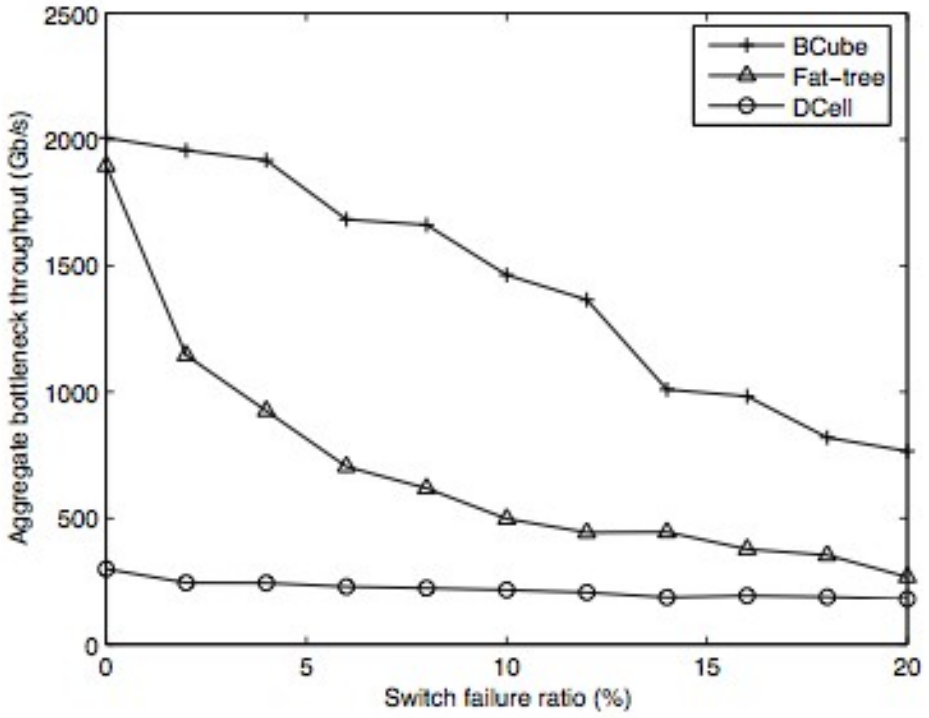
	Tree	Fat-tree	DCell$^+$	BCube
One-to-one	1	1	$k'+1$	$k+1$
One-to-several	1	1	$k'+1$	$k+1$
One-to-all	1	1	$\leq k'+1$	$k+1$
All-to-all(ABT)	n	N	$\frac{N}{2^{k'}}$	$\frac{n(N-1)}{n-1}$
Traffic balance	No	Yes	No	Yes
Graceful degradation	bad	fair	good	good
Wire No.	$\frac{n(N-1)}{n-1}$	$N\log_{\frac{n}{2}}\frac{N}{2}$	$(\frac{k'}{2}+1)N$	$N\log_n N$
Switch upgrade	No	Yes	No	No

$^+$A level-2 ($k'=2$) DCell with $n=8$ is enough for shipping-container. Hence k' is smaller than k.

Table 1: Performance comparison of BCube and other typical network structures.


BCube
Fat-tree
DCell
Aggregate bottleneck throughput (Gb/s)
Server failure ratio (%)
0
500
1000
1500
2000
2500
0
5
10
15
20

(a)


BCube
Fat-tree
DCell
Aggregate bottleneck throughput (Gb/s)
Switch failure ratio (%)
0
500
1000
1500
2000
2500
0
5
10
15
20

(b)

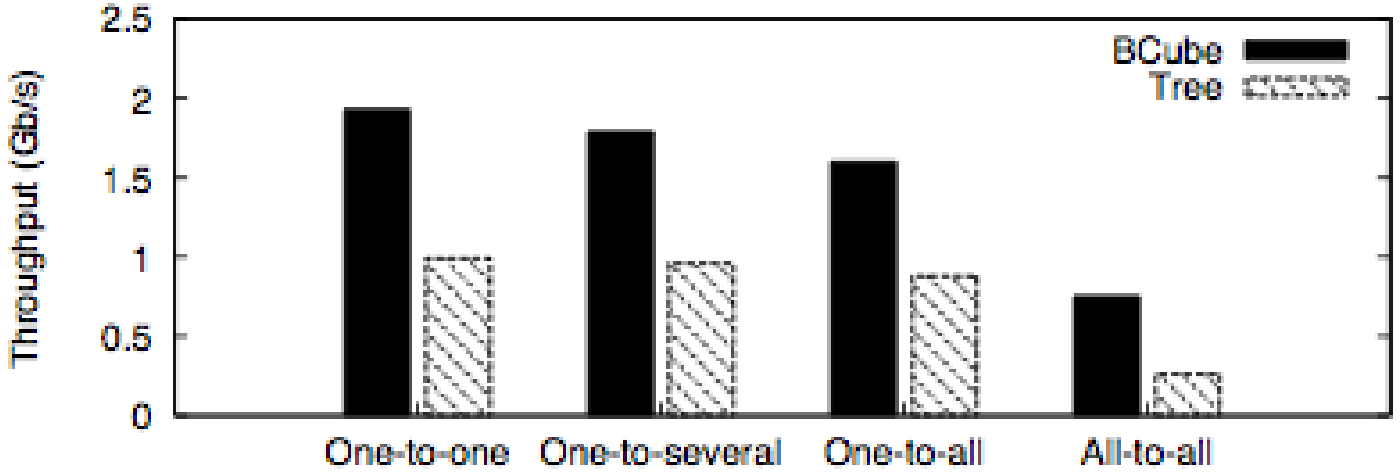


Figure 10: Per-server throughput of the bandwidth-intensive application support experiments under different traffic patterns.

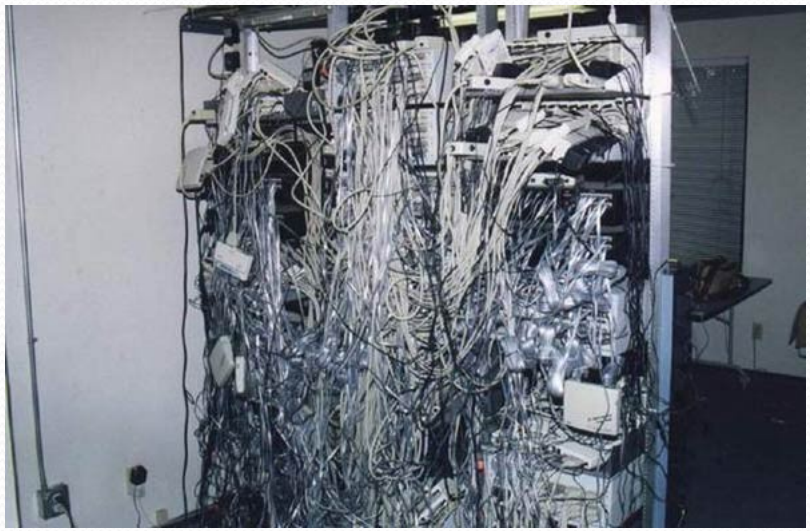
What is MDCube?

- Using self-contained modules of servers is a great idea! But how do you connect multiple containers together?

- MDCube is a way to inter-connect multiple Bcube containers.

InterConnect Challenges

- 1. There is a high Inter-Connect bandwidth requirement.
- 2. Keeping the cost of the Inter-Connect structure down.
- 3. Keeping the complexity of the cabling down.



How other methods handle the Challenges

- Traditional Implementations do fairly well at #1, at the cost of Challenges 2 and 3...
- In order to provide the bandwidth requirement, traditional approaches have to scale up or out.
- The physical distance between a large number of containers becomes a practical cabling barrier.

The Switches of BCube

- BCube uses COTS switches inside the containers, each containing many 1GB ports, and up to 4 high speed, 10GB interfaces.

- The idea is to connect the (unused) highspeed ports to peer switches in other containers.

Creating a MDCube

- Treat each BCube as a node, and each switch in the BCube as a port.
- Connect a port from each Bcube to a port in each other Bcube in your dimension. (A 1-d MDCube is a completely connected graph of Bcubes).

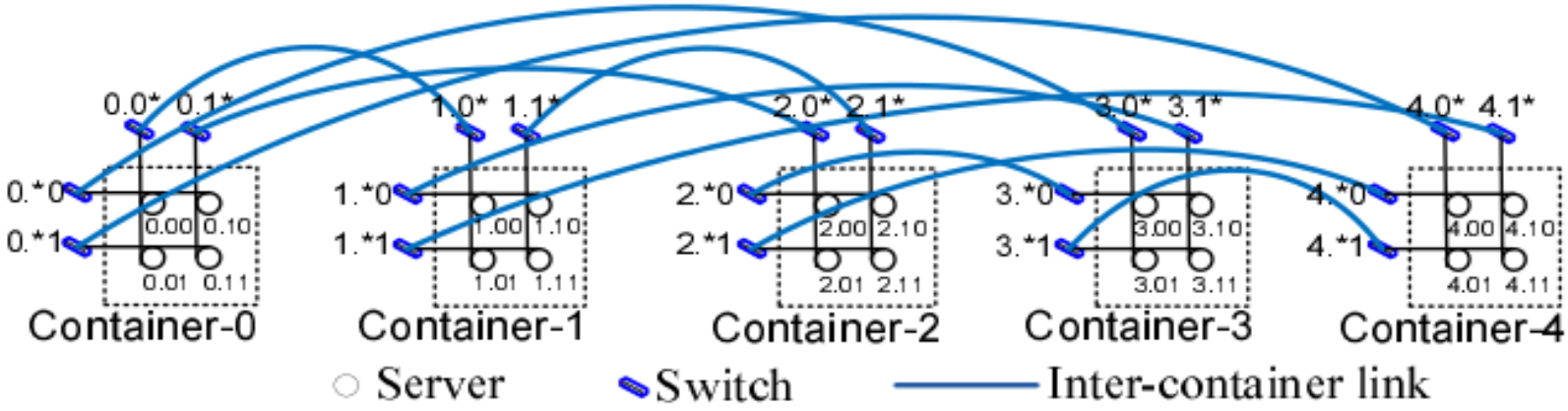


Figure 8: The 1-D MDCube testbed built from 5 $BCube_1$ containers with n=2, k=1.

Example MDCubes

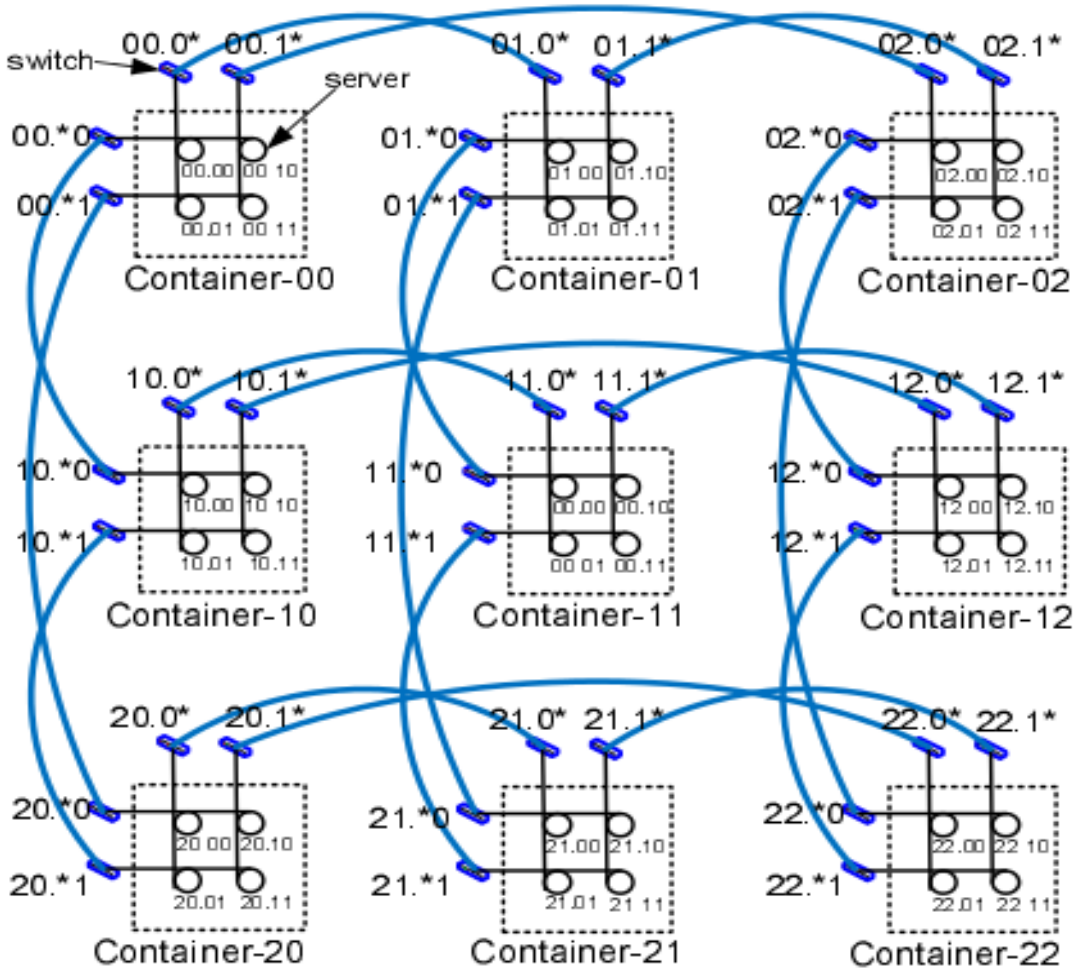


Figure 3: A 2-D MDCube is constructed from 9=3*3 BCube_1 Containers with n=2, k=1.

Size Limitations

- Size limitations aren't that big of a deal.
- A Bcube using 48 port switches:
  Means a Bcube1 could have 2304 servers and 96 switches, per cube.
- A 1-d MDCube built from these BCubes has a max of 97 containers, or 220,000 servers
- A 2-d MDCube has a max of 2401 containers, or 5,500,000 servers.

(There is a proof of size constraints of MDCube in the paper)

Does MDCube succeed?

- Challenge 1: High Inter-Connect bandwidth

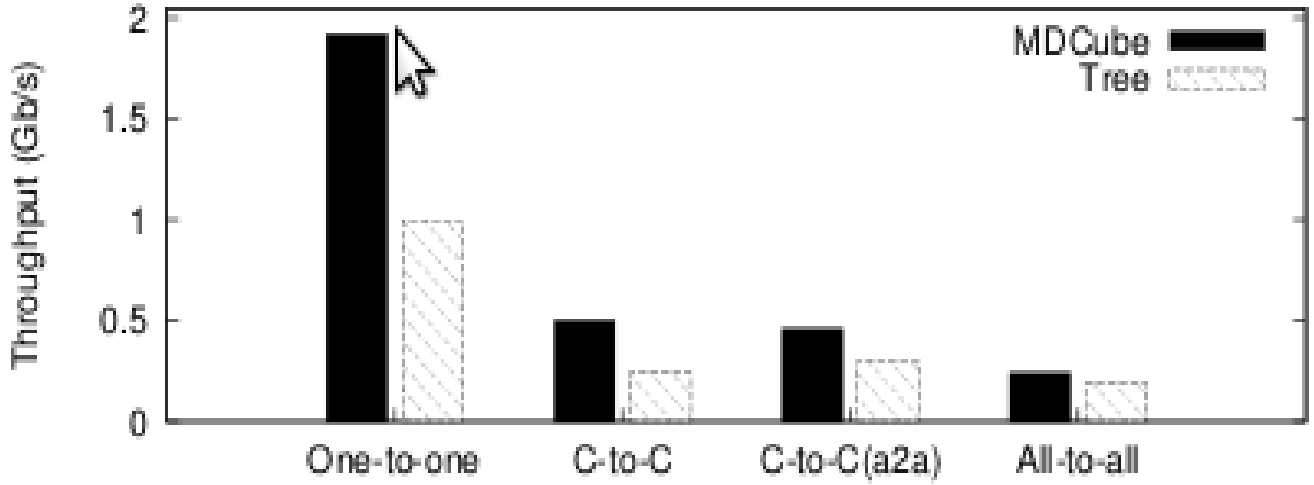


Figure 10: Per-server throughput under different traffic patterns.

Does MDCube Succeed?

- Challenge 2: Low inter-connect structure cost.
- No extra switches need, other than the ones already in each Bcube.
- Uses COTS Switches

Does MDCube Succeed?

- Challenge 3: Low Cabling Complexity
- The number of interconnects is low compared with the number of containers
- The length of any interconnect is small in each dimension of the MDCube.

MDCube During Failure

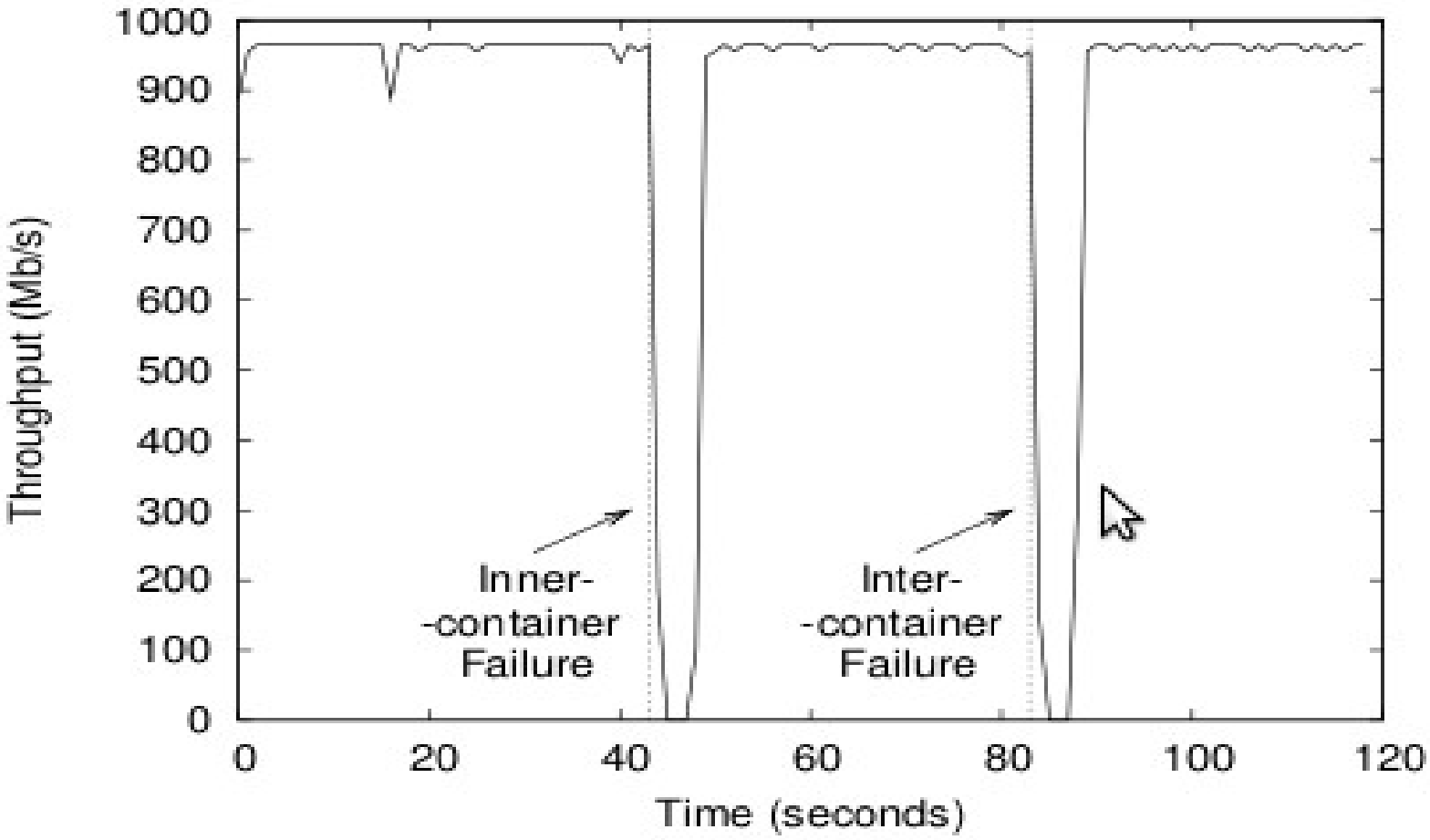


Figure 11: TCP connection experiences failures.

MDCube Discussion

- How does MDCube compare to Helios in terms of complexity of design and cost of infrastructure?
- Will MDCube scale better or worse than Helios in terms of hardware? Latency?

References

- Primary: H. Wu, G. Lu, D. Li, C. Guo, and Y. Zhang. MDCube: A High Performance Network Structure for Modular Data Center Interconnection. In ACM CoNEXT ’09.
- Secondary: C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, and S. Lu. BCube: A High Performance, Server-centric Network Architecture for Modular Data Centers. In Proc. SIGCOMM, 2009.
- Secondary: Greenberg, A., et al., “The Cost of a Cloud: Research Problems in Data Center Networks”, CCR, v39, n1, Jan’09
- Secondary: J. Hamilton. Architecture for modular data centers. In Third Biemnial Conference on Innovative Data Systems, 2007