ASPIRE

DIABLO: Simulating Datacenter Network at Scale using FPGAs

Zhangxi Tan, Zhenghao Qian, Xi Chen

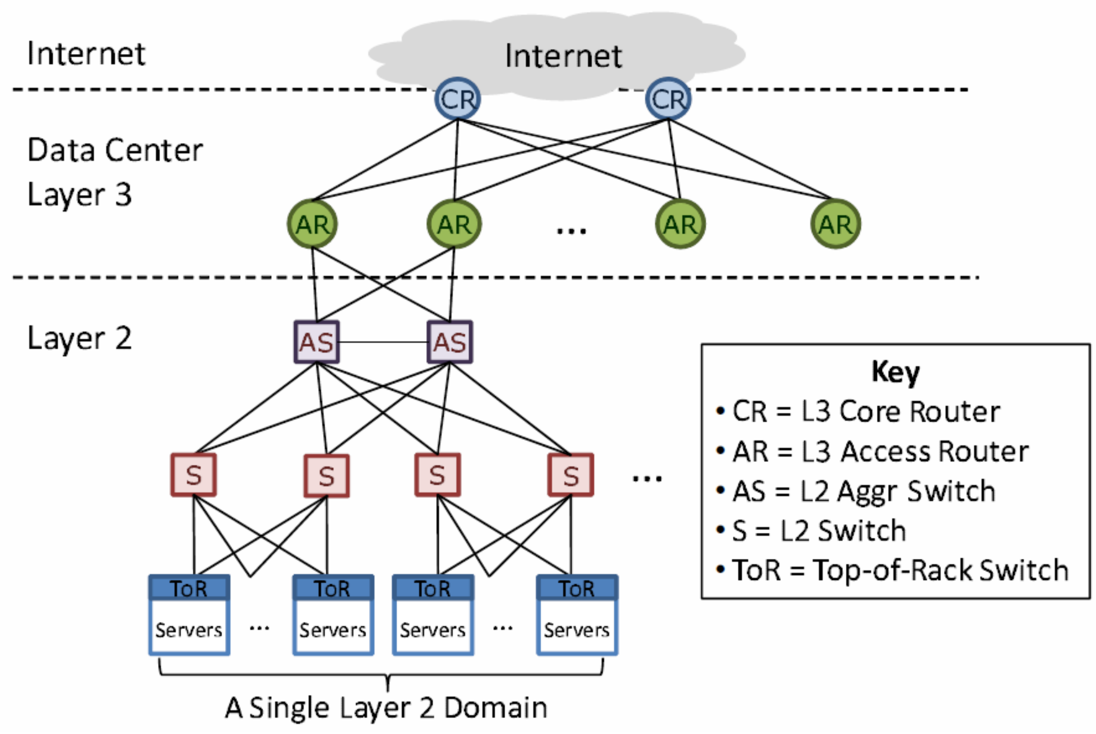
Krste Asanovic, David Patterson

UC Berkeley

September 2013

Datacenter Network Architecture Overview

- Conventional datacenter network (Cisco's perspective)



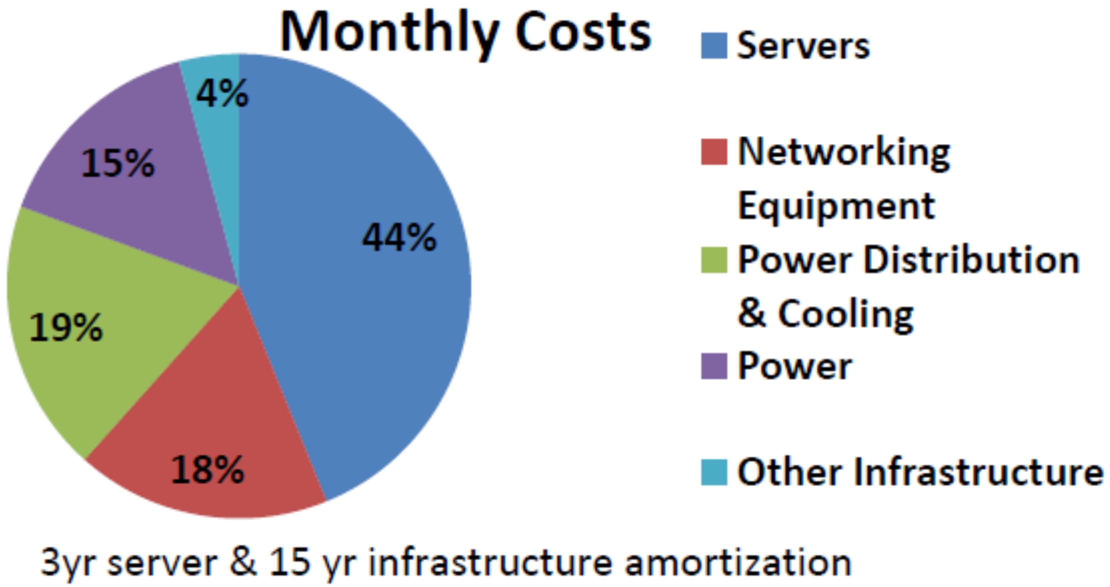
Key

- CR = L3 Core Router
- AR = L3 Access Router
- AS = L2 Aggr Switch
- S = L2 Switch
- ToR = Top-of-Rack Switch

Figure from "VL2: A Scalable and Flexible Data Center Network"

Observations

- Network infrastructure is the "SUV of datacenter"
  - 18% monthly cost (3rd largest cost)
  - Large switches/routers are expensive and unreliable
  - Important for many optimizations:
    - Improving server utilization
    - Supporting data intensive map-reduce jobs



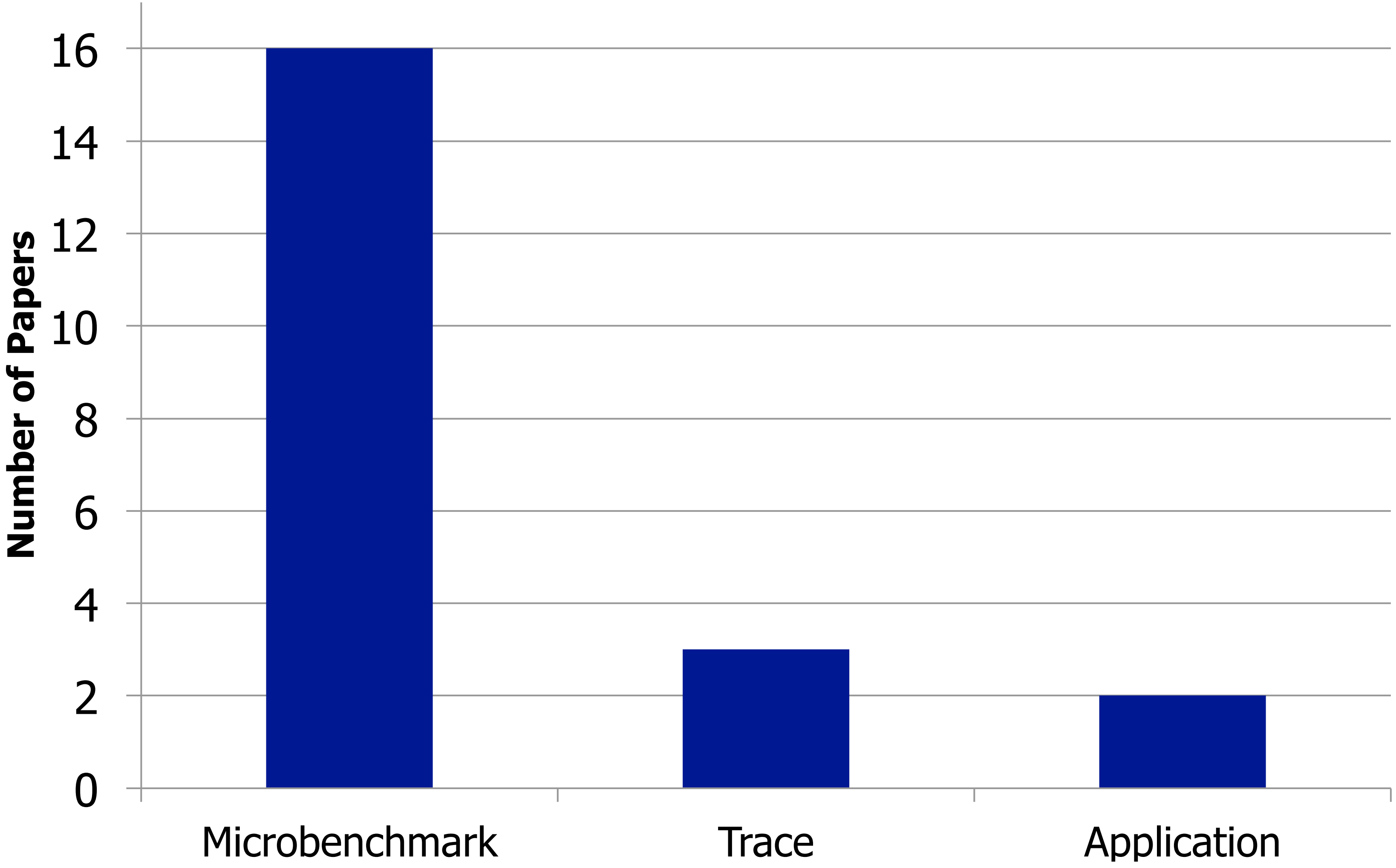
Source: James Hamilton, Data Center Networks Are in my Way, Stanford Oct 2009

Advances in Datacenter Networking

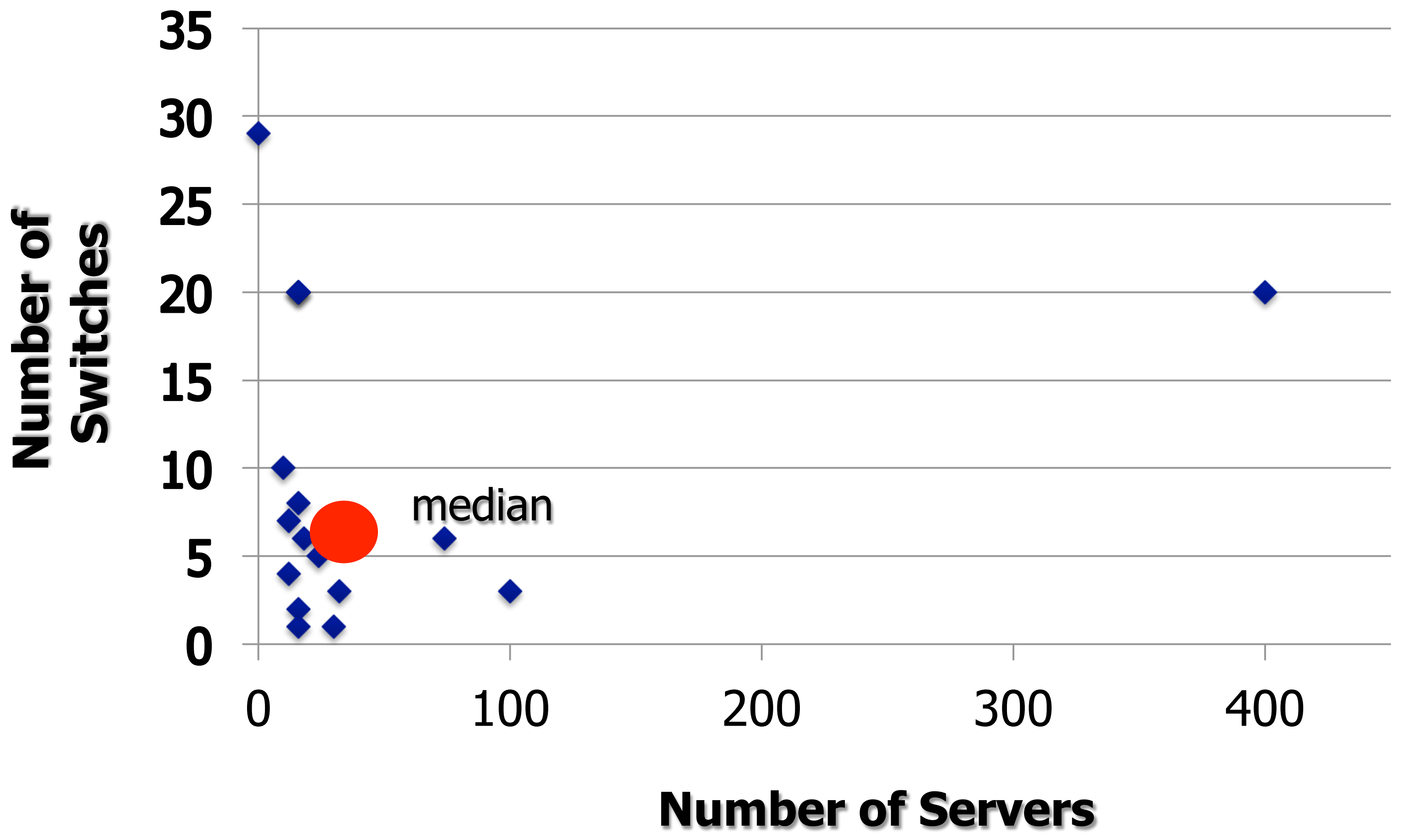
- Many new network architectures proposed recently focusing on new switch designs
  - Research : VL2/monsoon (MSR), Portland (UCSD), Dcell/Bcube (MSRA), Policy-aware switching layer (UCB), Nox (UCB), Thacker's container network (MSR-SVC)
  - Product : Google g-switch, Facebook 100 Gbps Ethernet and etc.

- Different observations lead to many distinct design features
  - Switch designs
    - Packet buffer micro-architectures
    - Programmable flow tables
  - Application and protocols
    - ECN support

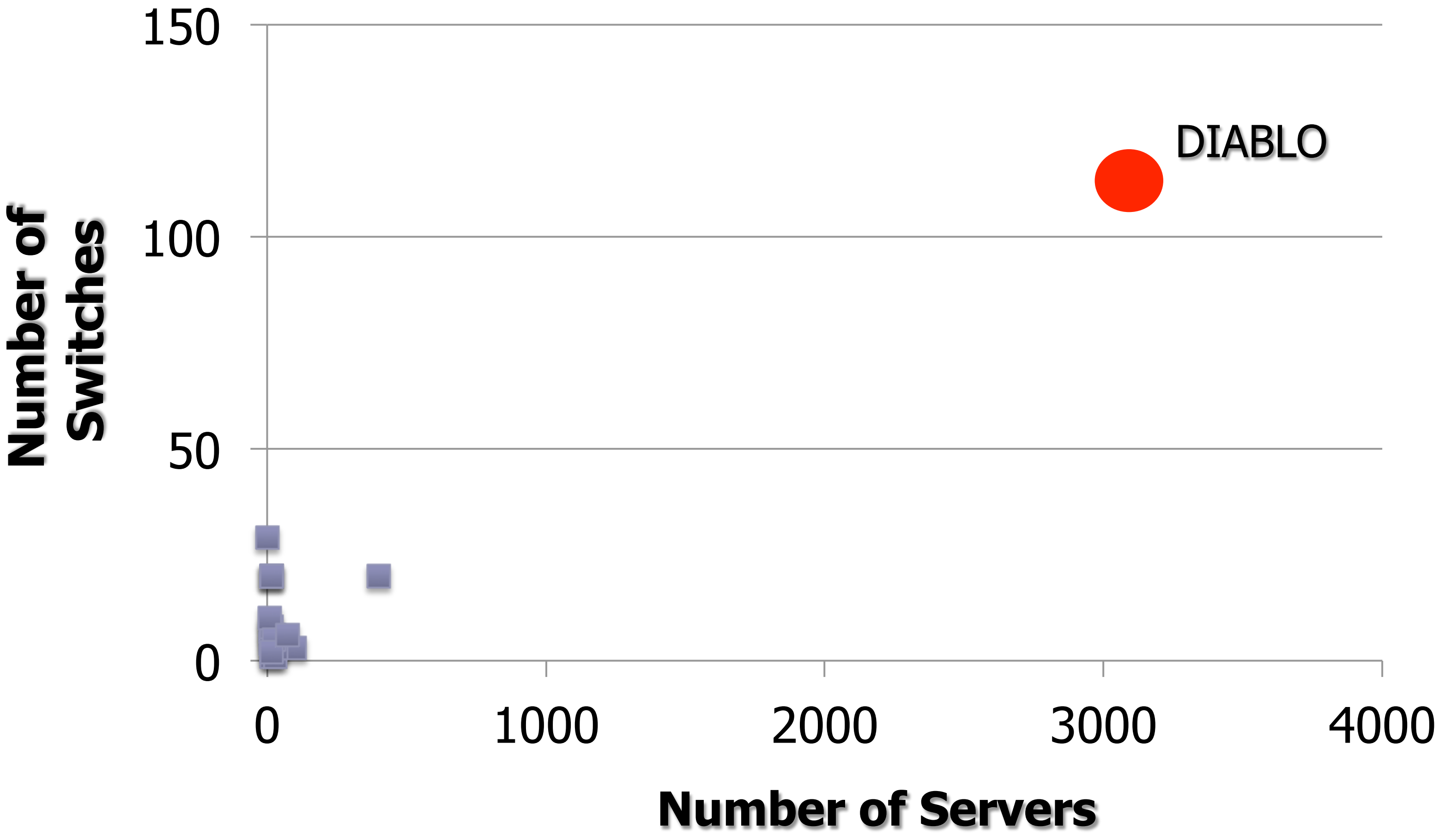
Workload Used in recent SIGCOMM Papers



Testbed Scale in Recent SIGCOMM Papers



DIABLO 1 vs. Others

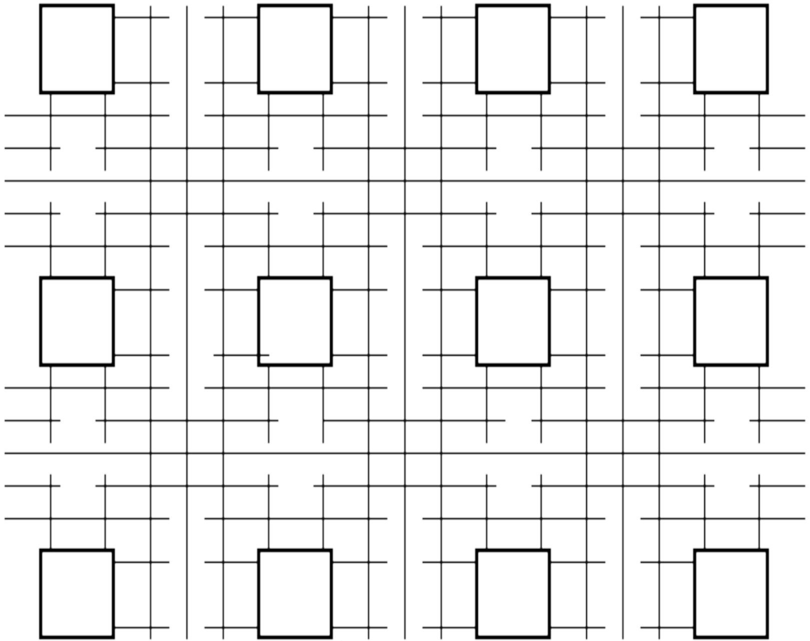


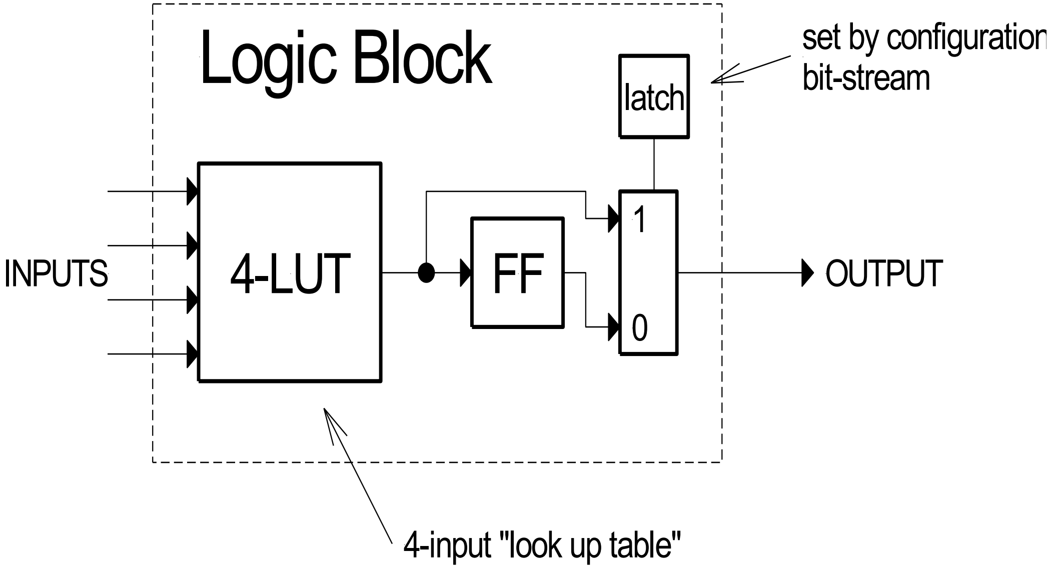
A wish list of networking evaluations

- Evaluating networking designs is hard
  - □ Datacenter scale at O(10,000) -> need scale
  - □ Switch architectures are massively parallel -> need performance
    - Large switches has 48~96 ports, 1K~4K flow tables/port. 100~200 concurrent events per clock cycle
  - □ Nanosecond time scale -> need accuracy
    - Transmit a 64B packet on 10 Gbps Ethernet only takes ~50ns, comparable to DRAM access! Many fine-grained synchronizations in simulation
  - □ Run production software -> need extensive application logic

My proposal

- Use Field Programmable Gate Array (FPGAs)



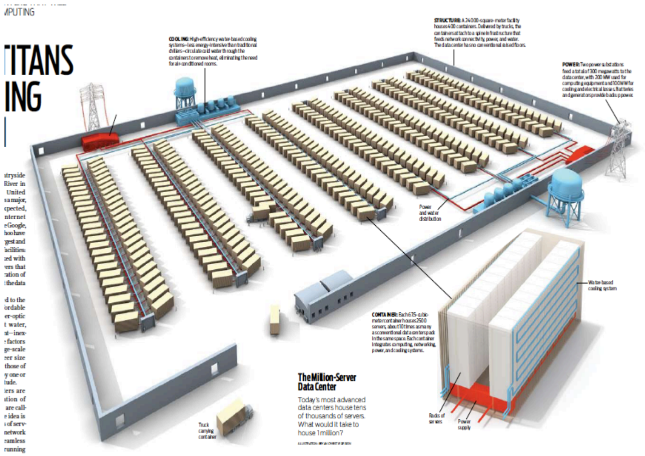


- DIABLO: Datacenter-in-A-Box at Low Cost
  - □ Abstracted execution-driven performance models on FPGAs
    - Not FPGA computers/accelerators
  - □ Evaluating datacenter as a computer system (FAME)
  - □ Cost ~$12 per node at O(10,000)

Tan et al. "A case for FAME: FPGA architecture model execution", ISCA'10

DIABLO Overview

- Build a “wind tunnel” for datacenter network using FPGAs
  - Simulate O(10,000) nodes: each is capable of running real software
  - Simulate O(1,000) datacenter switches (all levels) with detail and accurate timing
  - Simulate O(100) seconds in target
  - Runtime configurable architectural parameters (link speed/latency, host speed)

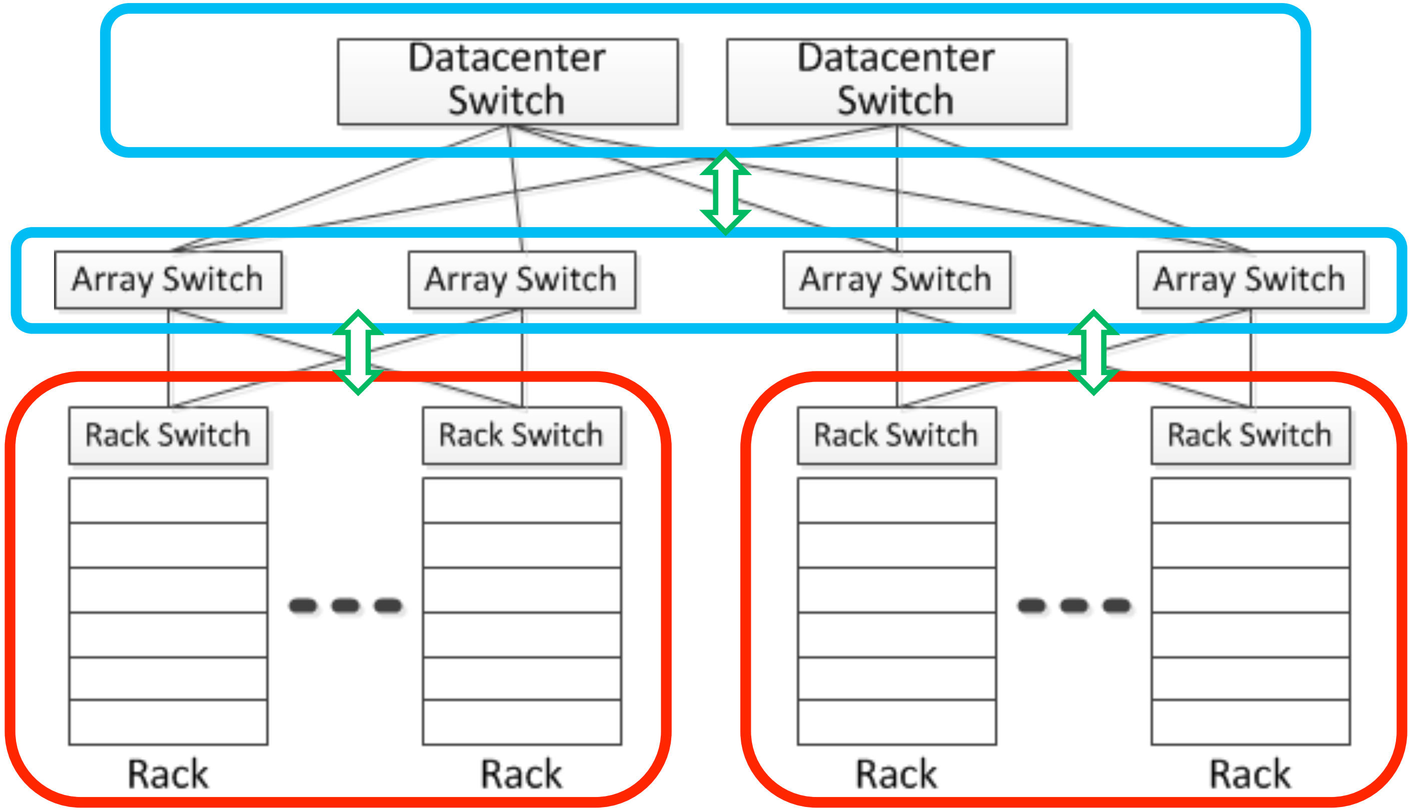

MPUTING
TITANS
ING
COOLING: High-efficiency water-based cooling systems—less energy-intensive than traditional chillers—circulate cold water through the containers to remove heat, eliminating the need for air-conditioned rooms.
STRUCTURE: A 24 000-square-meter facility houses 400 containers. Delivered by trucks, the containers attach to a spine infrastructure that feeds network connectivity, power, and water. The data center has no conventional raised floors.
POWER: Two power substations feed a total of 300 megawatts to the data center, with 200 MW used for computing equipment and 100 MW for cooling and electrical losses. Batteries and generators provide backup power.
Power and water distribution
Water-based cooling system
CONTAINER: Each 67.5-cubic-meter container houses 2500 servers, about 10 times as many as conventional data centers pack in the same space. Each container integrates computing, networking, power, and cooling systems.
The Million-Server Data Center
Today's most advanced data centers house tens of thousands of servers. What would it take to house 1 million?
Truck carrying container
Racks of servers
Power supply

Executing real instructions, moving real bytes in the network!

DIABLO Models

- Server models
  - Built on top of RAMP Gold : SPARC v8 ISA, 250x faster than software simulators
  - Run full Linux 3.5 with a fixed CPI timing model
- Switch models
  - Two types: circuit-switching and packet-switching
  - Abstracted models focusing on switch buffer configurations
    - Model after Cisco Nexsus switch + a Broadcom patent
- NIC models
  - Scather/gather DMA + Zero-copy drivers
  - NAPI polling support

Mapping a datacenter to FPGAs



- Modularized single-FPGA designs: two types of FPGAs
  - □ Connecting multiple FPGAs using multi-gigabit transceivers according to physical topology
  - □ 128 servers in 4 racks per FPGA; one array/DC switch per FPGA

DIABLO Cluster Prototype

- 6 BEE3 boards total 24 Xilinx Virtex5 FPGAs
  - □ Physical characteristics:
    - Full-custom FPGA implementation with lots of reliability features @ 90/180 MHz
    - Memory: 384 GB (128 MB/node), peak bandwidth 180 GB/s
    - Connected with SERDES @ 2.5 Gbps
    - Host control bandwidth: 24 x 1 Gbps control bandwidth to the switch
    - Active power: ~1.2 kwatt
  - □ Simulation capacity
    - 3,072 simulated servers in 96 simulated racks, 96 simulated switches
    - 8.4 B instructions / second

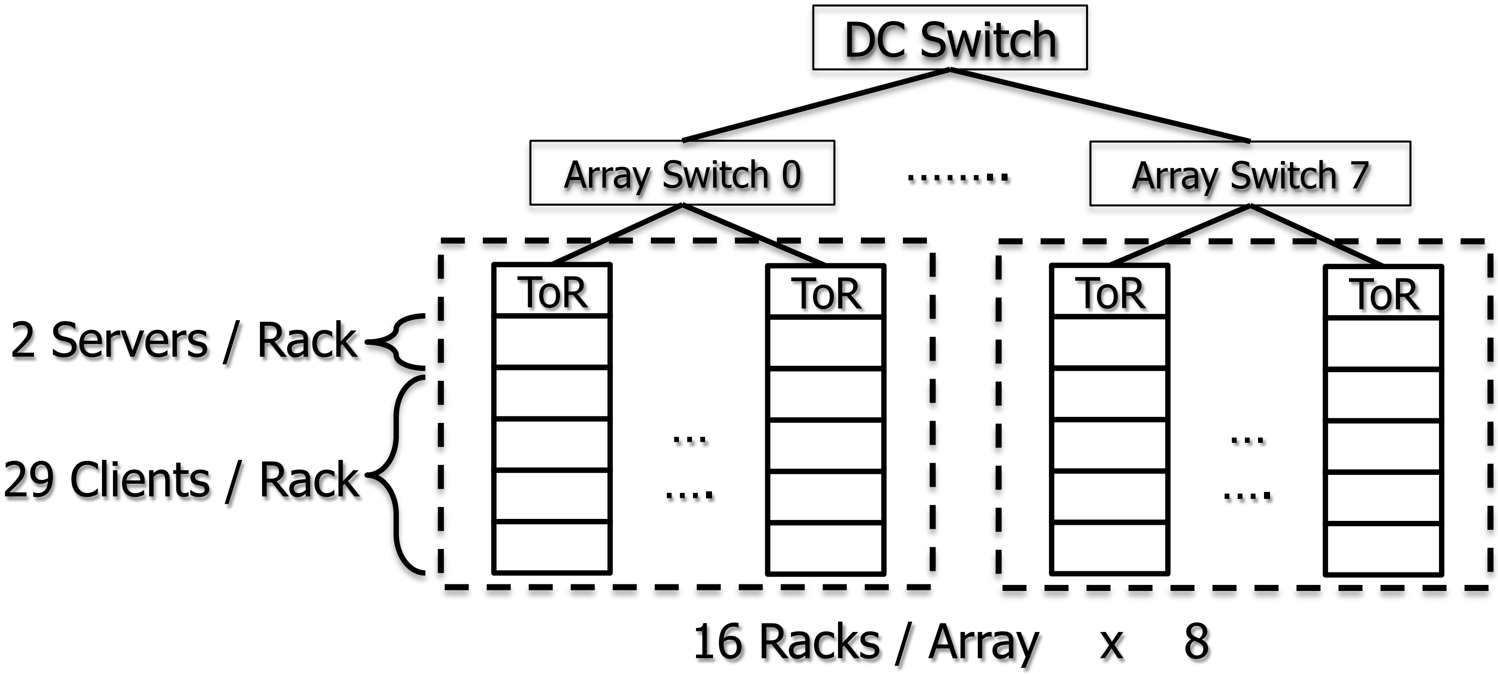
Simulator Scaling for a 10,000 system

	Server Model Pipelines per chip	Simulated Servers per chip	Total FPGAs
DIABLO 1 (65nm 2007-era Virtex 5)	4	128	88 (22 BEE3s)
DIABLO 2 (28nm 2013-era Virtex 7)	32	1024	12

- Total cost @ 2013: ~$120K
  - □ Board cost: $5,000 * 12 = $60,000
  - □ DRAM cost: $600 * 8 * 12 = $57,000

Running production datacenter software: memached

- Popular distributed key-value store app, used by many large websites: Facebook, twitter, Flickr,…
- Unmodified memcached + clients in libmemcached
  - Clients generate traffic based on Facebook statistics (SIGMETRICS'12)

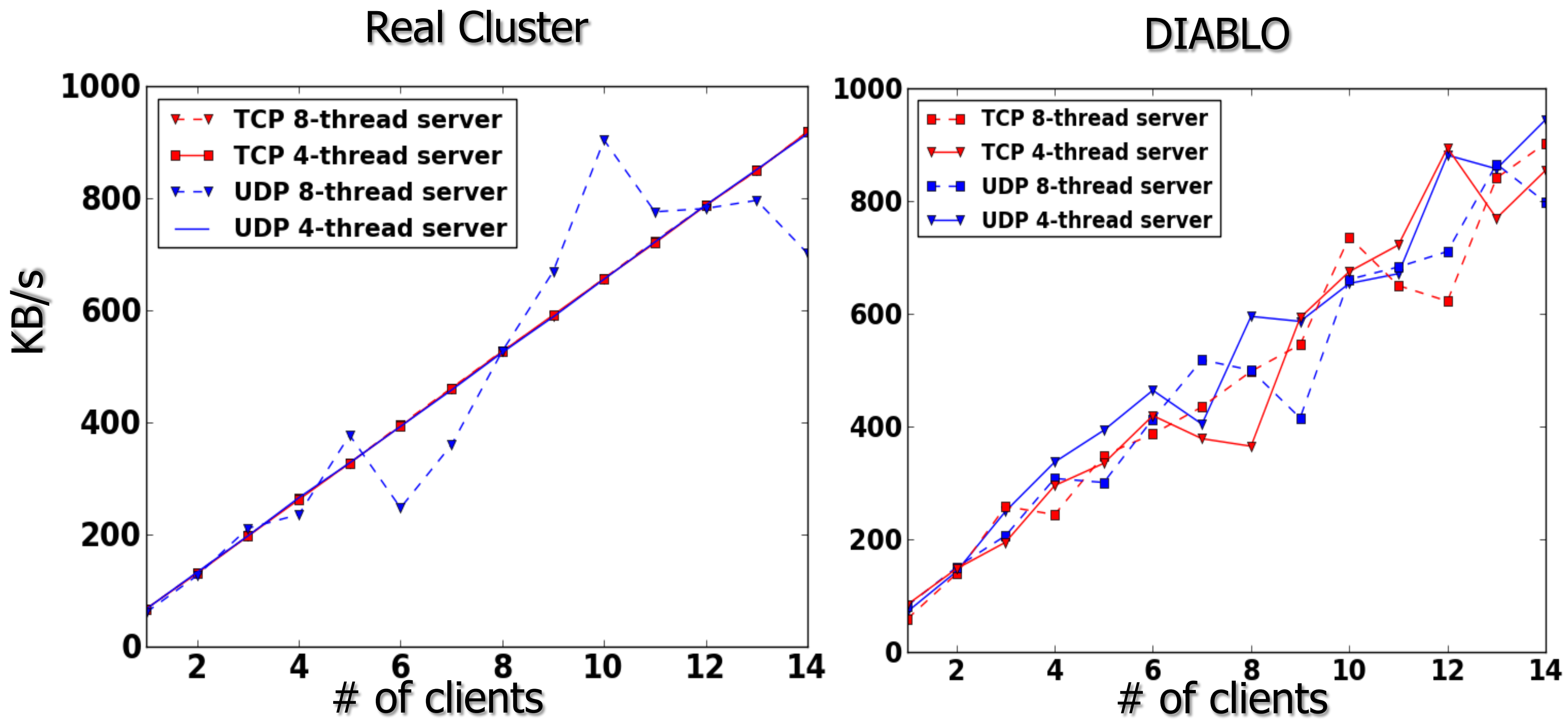


Validation against a single-rack physical system

- 16-node cluster 3 GHz Xeon + 16 port Asante IntraCore 35516-T switch
  - □ Physical hardware configuration: two servers + 1 ~ 14 clients
  - □ Software configurations
    - Server protocols: TCP/UDP
    - Server worker threads: 4 (default), 8

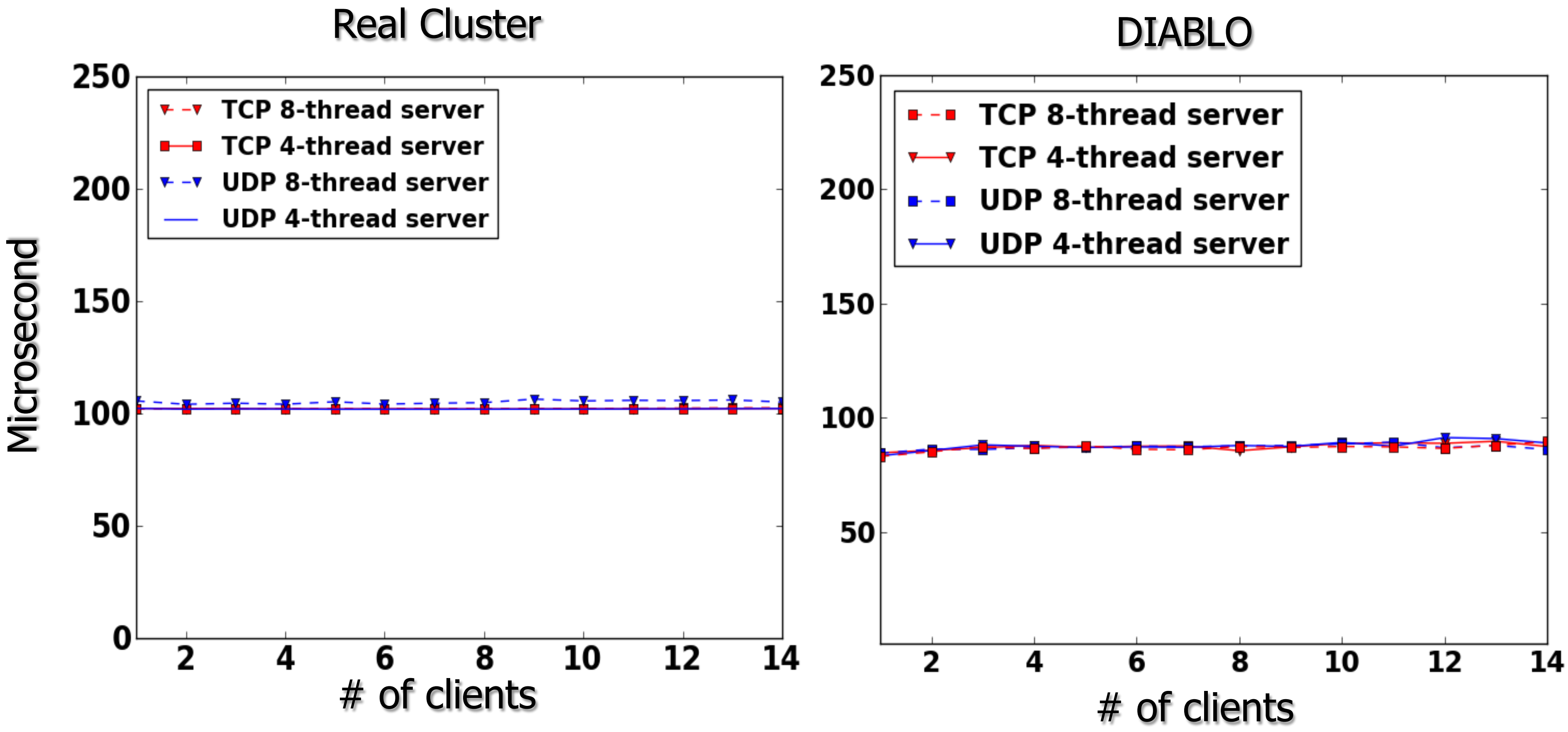
- Simulated server : single-core @ 4 GHz fixed CPI
  - □ Different ISA, different CPU performance

Validation: Server throughput



- Absolute values are close

Validation: Client latencies



- Similar trend as throughput, but absolute value is different due to different network hardware

Experiments at scale

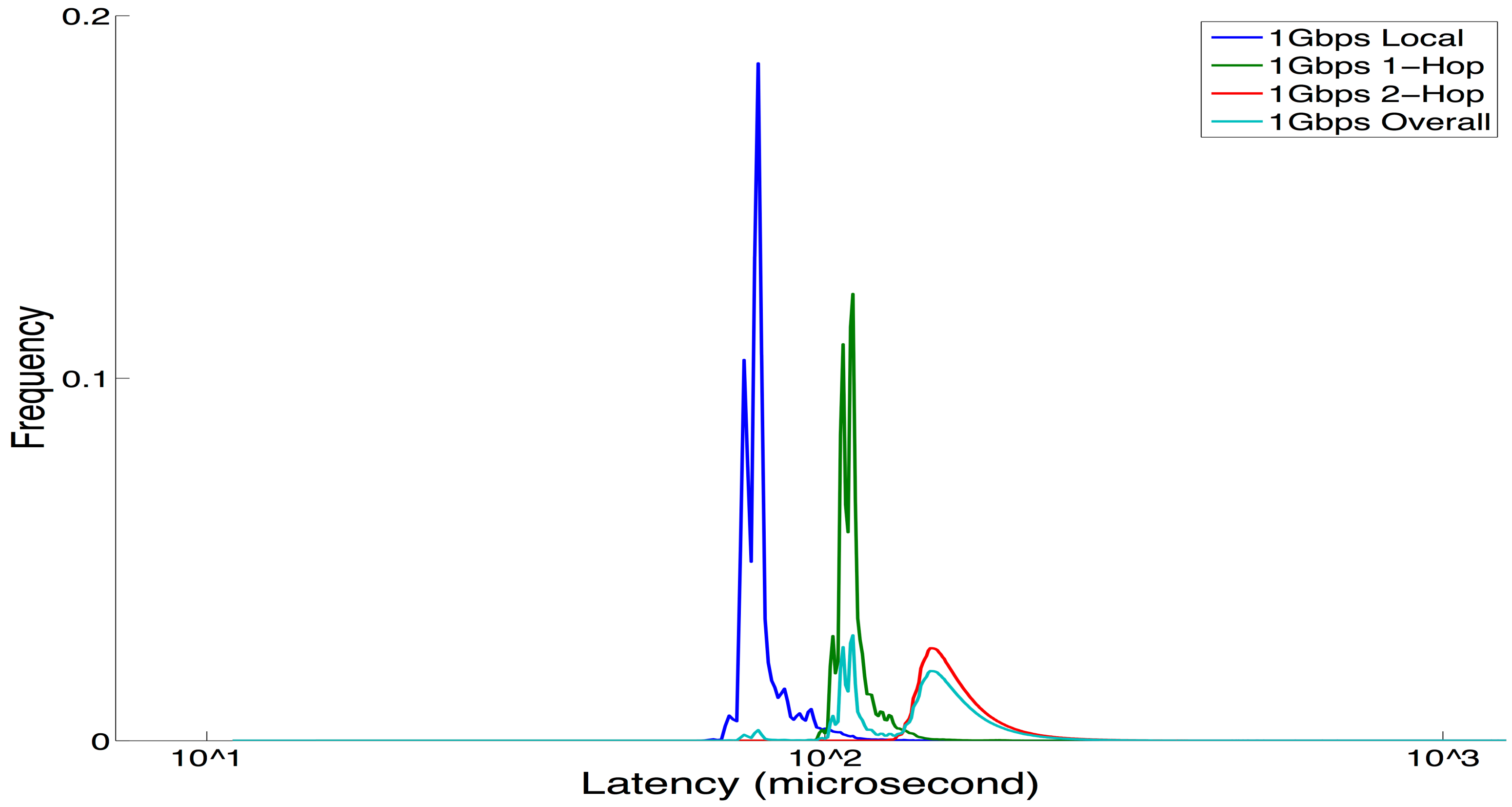
- Simulate up to the 2,000-node scale

- Two simulated interconnects
  - 1 Gbps interconnect : 1~1.5us port-port latency
  - 10 Gbps interconnect: 100~150ns port-port latency
    - 10x bandwidth, 10x shorter latency

- Large-scale questions to answer
  - Can we reproduce latency variations?
  - Will a 10x better interconnect help application performance by 10x?

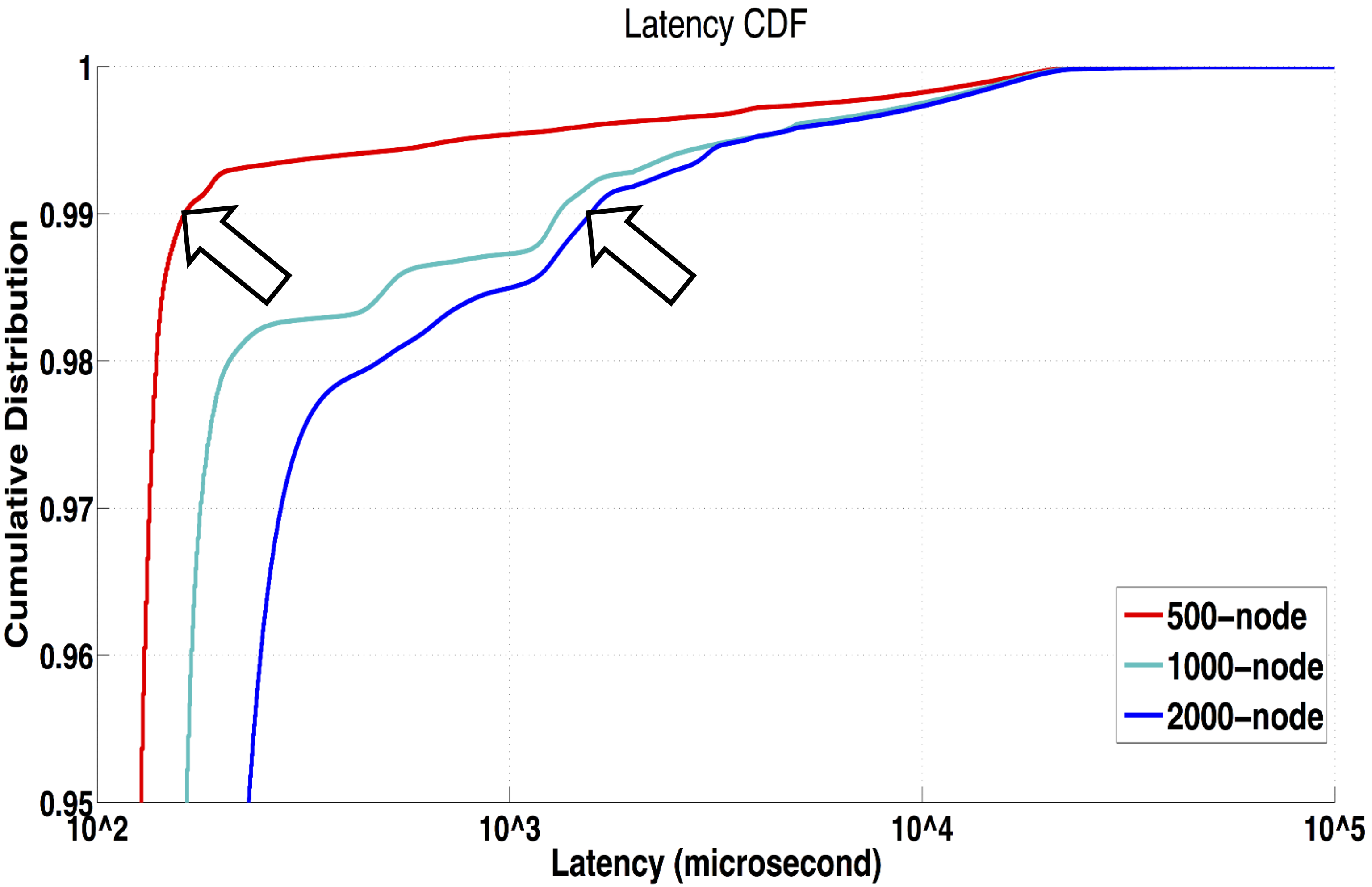
Reproducing the latency long tail at the 2,000-node scale

Luiz Barroso "entering the teenage decade in warehouse-scale computing" FCRC'11



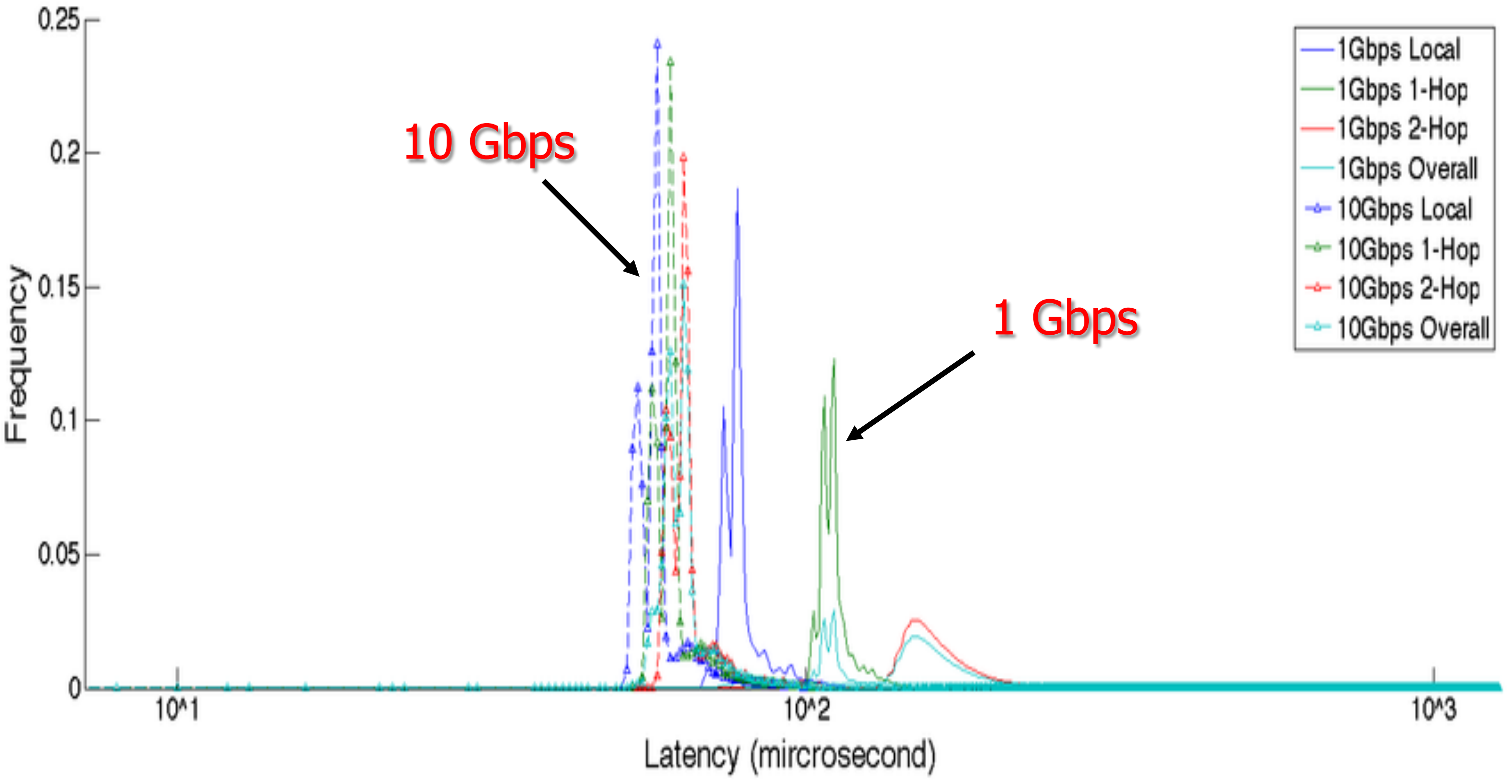
- Most requests finished ~100us, but some 2 orders of magnitude slower
- More switches -> greater latency variations

Impact of System Scale on the “long tail”



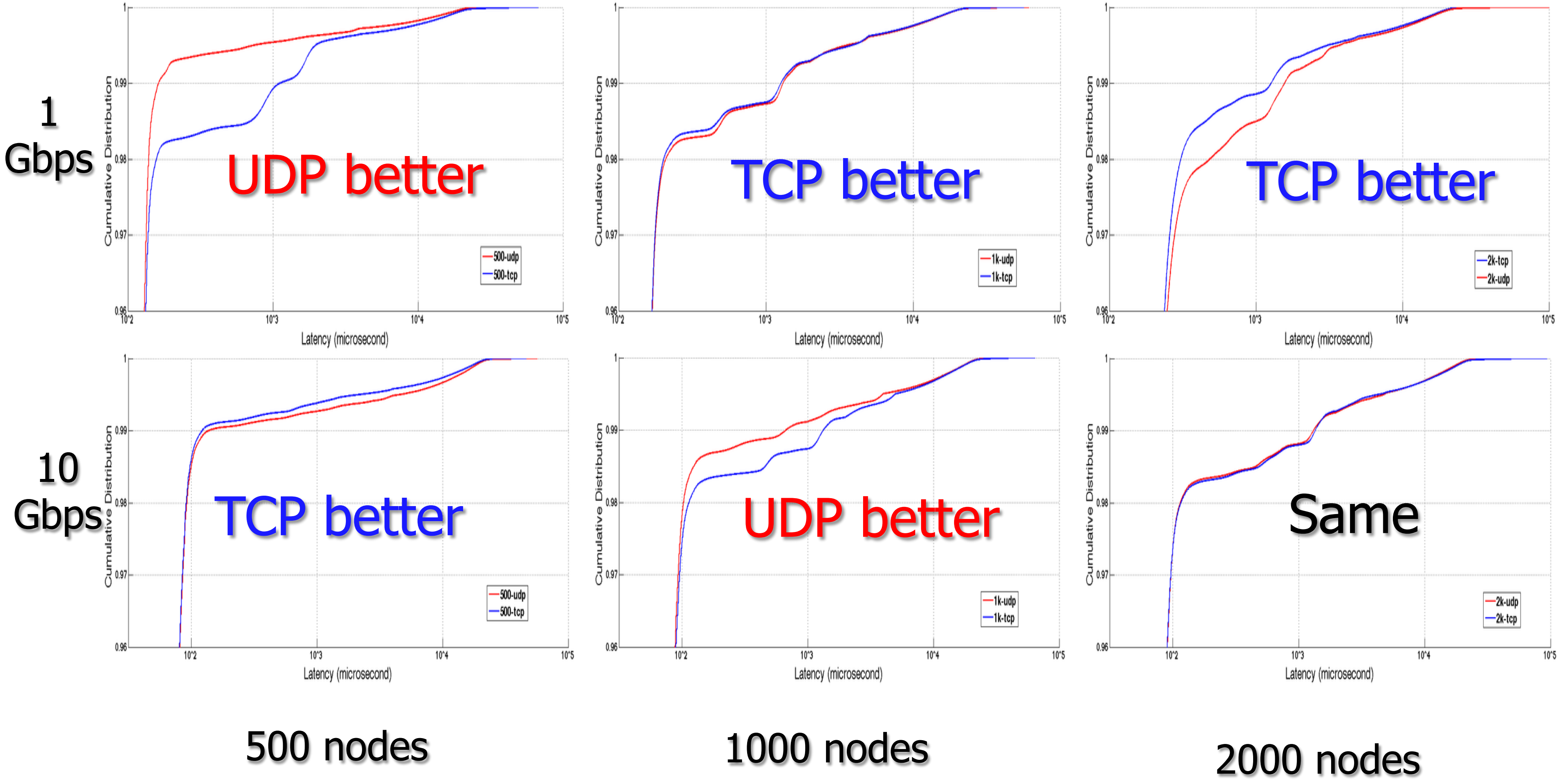
- The 99 percentile latency of 2,000-node is an order of magnitude worse than that of 500-node.

Improvement of a 10x Better Interconnect



- Low-latency 10Gbps switches improve access latency but only <2x
- The software stack dominates!

O(100) vs O(1000) at the “tail”



- Which network protocol is better at minimizing long tail?

Other issues at scale

- TCP does not consume more memory than UDP when server load is well-balanced

- Do we really need a fancy transport protocol?
  - □ Vanilla TCP might just perform fine
    - Don’t just focus on the protocol : CPU, NIC, OS and app logic
  - □ Too many queues/buffers in the current software stack

- Effects of changing interconnect hierarchy
  - □ Adding a datacenter-level switch affects server host DRAM usages

Experience and Lessons Learned

- Need massive simulation power even at rack level
  - DIABLO generates research data overnight with ~3,000 instances
  - FPGAs are slow, but not if you have ~3,000 instances
- Real software and kernel have bugs
  - Programmers do not follow hardware spec!
  - We modified DIABLO multiple times to support Linux hacks
- Massive-scale simulation have transient errors like real datacenter
  - E.g. Software crashes due to soft errors
- FPGAs are great, but we need better tools
  - FPGA Verilog/Systemverilog tools are not productive.
  - Chisel work at Berkeley

Conclusions

- Simulating the OS/Application crucial to understand network performance
- Can not generalize O(100) results to O(1,000)
- DIABLO is good enough to reproduce relative numbers
- A great tool for design-space exploration at scale

DIALBO is available at:

http://diablo.cs.berkeley.edu

Research partially supported by Microsoft (Award #024263) and Intel (Award #024894) funding and by matching funding by U.C. Discovery (Award #DIG07-10227). Additional support comes from Par Lab affiliates Nokia, NVIDIA, Oracle, and Samsung. Research also partially supported by DARPA Award Numbers FA8650-09-C-7907 and HR0011-12-2-0016.