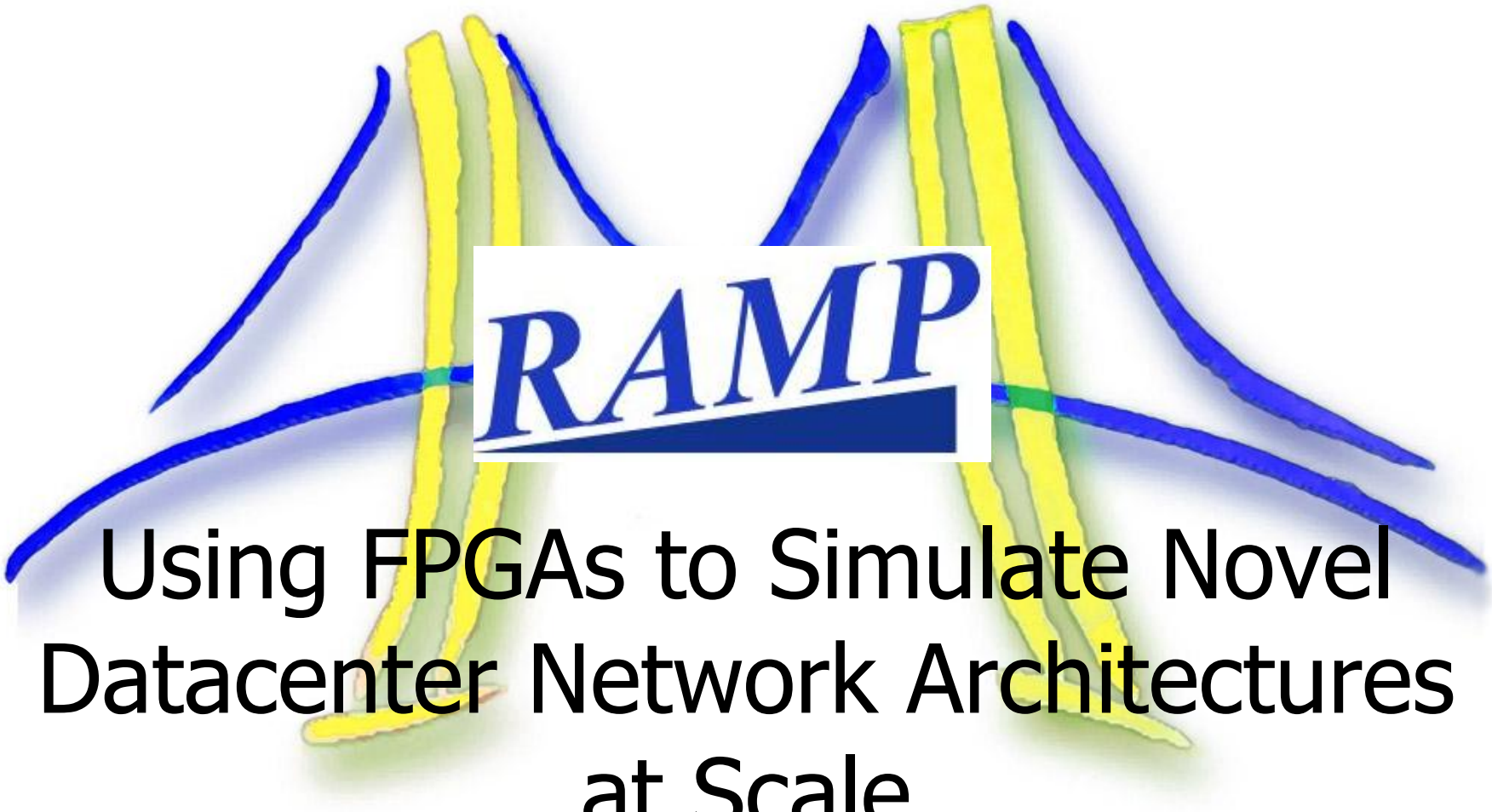


Using FPGAs to Simulate Novel Datacenter Network Architectures at Scale

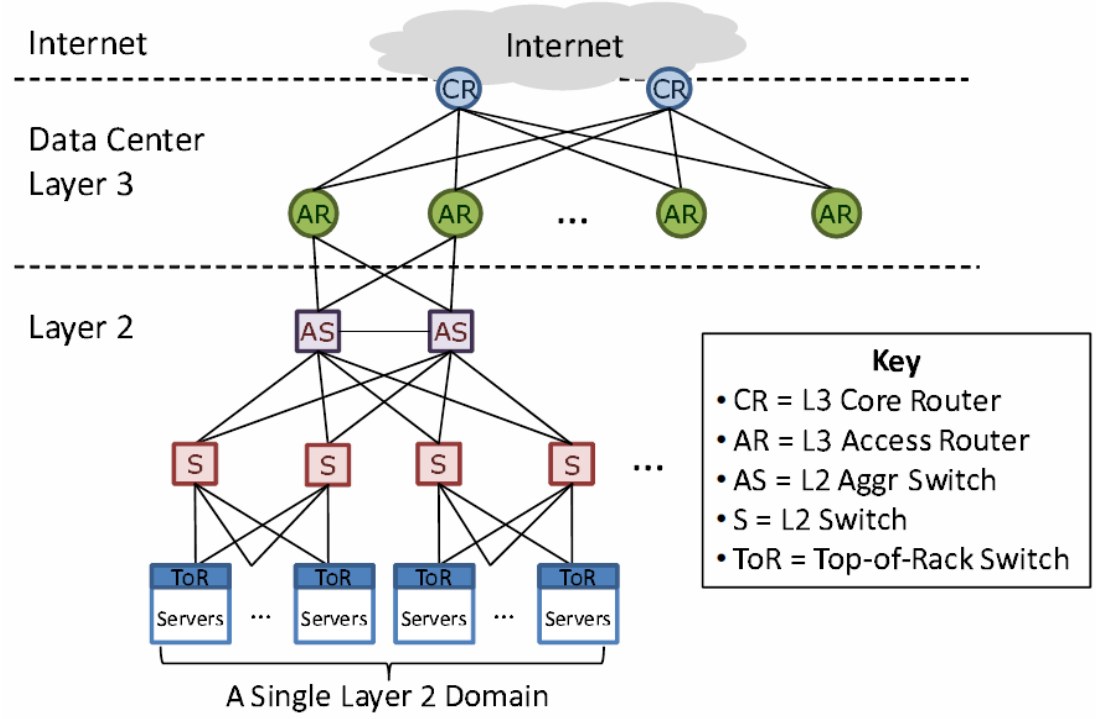
Zhangxi Tan, Krste Asanovic, David Patterson
UC Berkeley
March 2010

Outline

- Datacenter Network Overview
- RAMP Gold
- Modeling Nodes and Network
- A Case Study: Simulating TCP Incast problem
- Related Work
- Conclusion and Questions

Datacenter Network Architecture Overview

- Conventional datacenter network (Cisco's perspective)

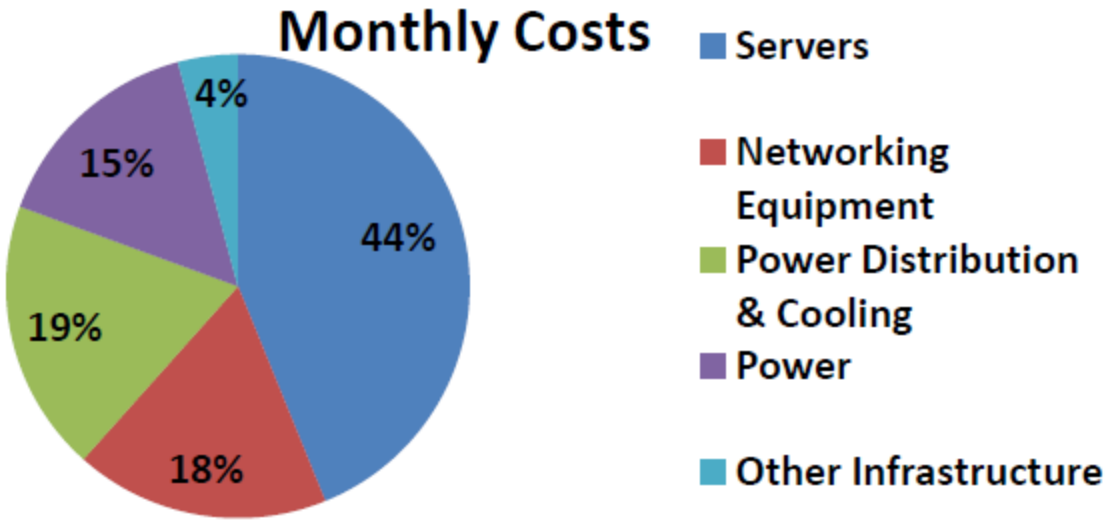


- Modern modular datacenter (Microsoft Chicago center) : 40~80 machines/rack, 1,800~2,500 servers per container, 150-220 containers on the first floor, ~50 rack switches per container

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

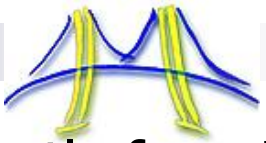
Observations

- 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



3yr server & 15 yr infrastructure amortization

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



- 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 : low-latency cut-through switch (Arista Network), Infiniband switch for Datacenter (Sun)

- Different observations lead to many distinct design features
  - Switch designs
    - Store-forward vs. Cut-through
    - Input only buffering vs. Input and output buffering
    - Low radix vs. High radix
  - Network designs
    - Hyper-cube vs. Fat-tree
    - State vs. Stateless core
  - Application and protocols
    - MPI vs. TCP/IP

Comments on the SIGCOMM website

- DCell
  - Are there any implementations or tests of DCells available?
- VL2
  - At the end of section 5.2, it is mentioned that this method is used because of the big gap of the speed between server line card and core network links. 10x gap is a big gap, it is possible that for other design, the gap is smaller, like 5x, or smaller, if so, does random split also perform well? Though it is mentioned that, when this gap is small, instead of random split, sub-flow split maybe used, does this have effect on the performance of VL2?
- Portland
  - The evaluation is limited to a small testbed, which is understandable, but some of the results obtained may change significantly in a large testbed.

Problem

- The methodology to evaluate new datacenter network architectures has been largely ignored
  - Scale is **way smaller** than real datacenter network
    - <100 nodes, most of testbeds < 20 nodes
  - Synthetic programs and benchmarks
    - Datacenter Programs: Web search, email, map/reduce
  - Off-the-shelf switches architectural details are NDA
    - Limited architectural design space configurations: E.g. change link delays, buffer size and etc.

How to enable network architecture innovation at scale without first building a large datacenter?

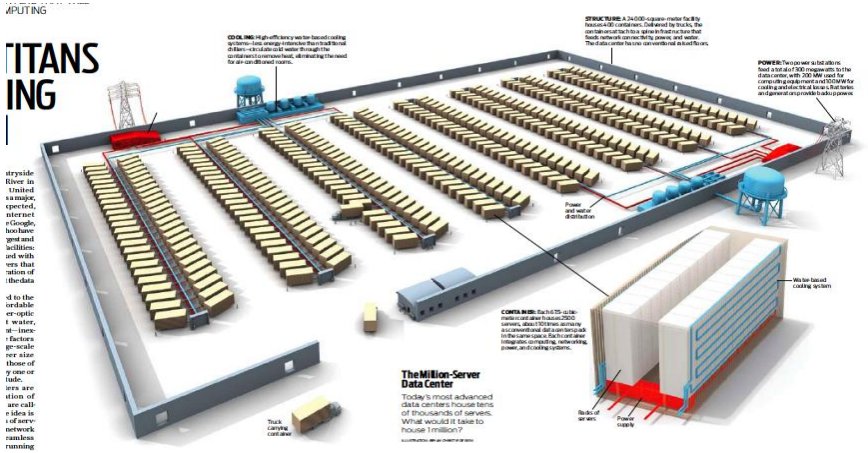
My Observation

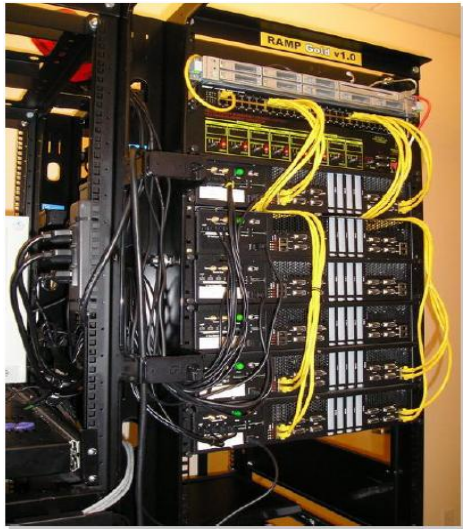
- Datacenters are computer systems
  - Simple and low latency switch designs:
    - Arista 10Gbps cut-through switch: 600ns port-port latency
    - Sun Infiniband switch: 300ns port-port latency
  - Tightly-coupled supercomputing like interconnect

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

My Approach

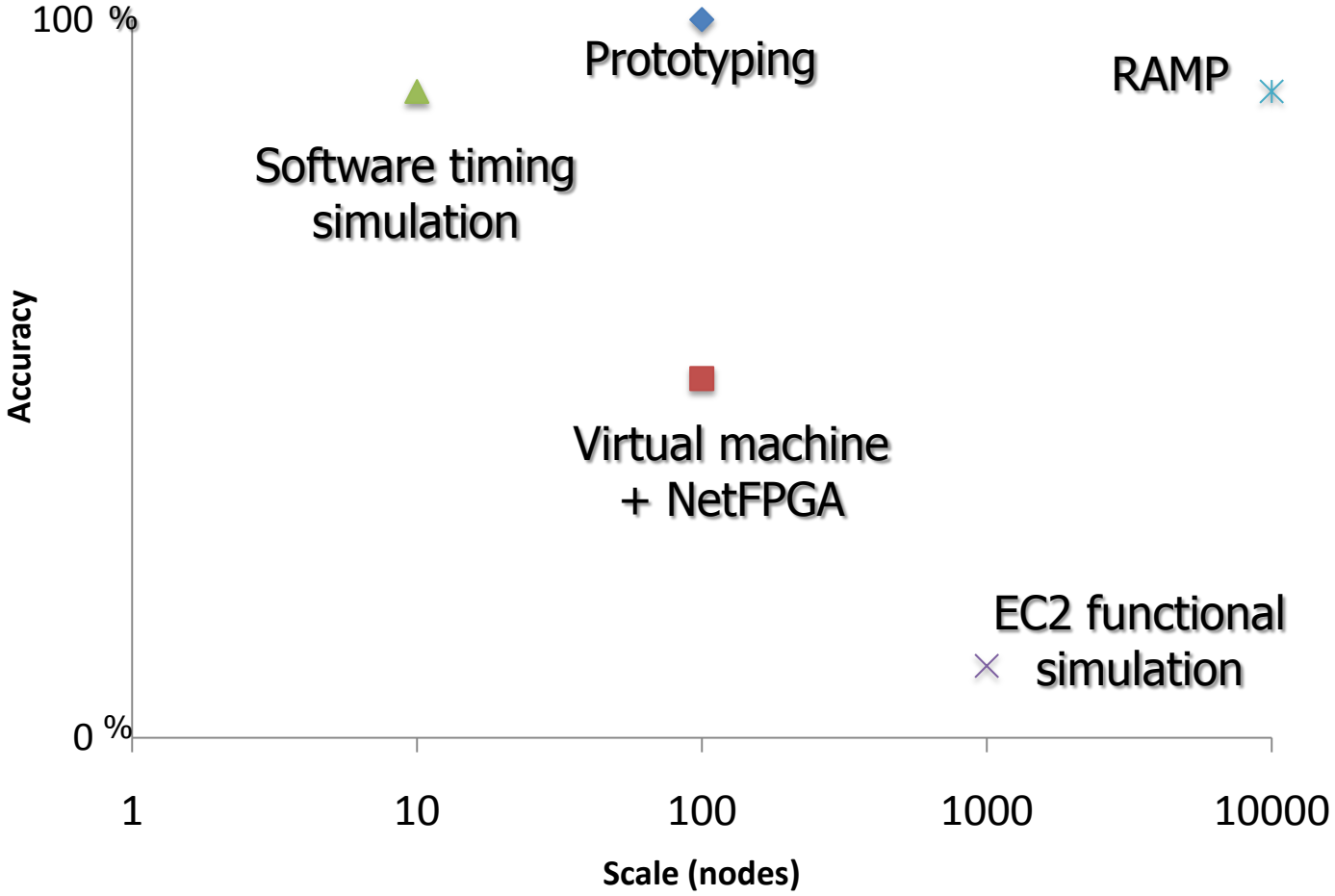
- Build a "wind tunnel" for datacenter network using FPGAs
  - Simulate O(10,000) nodes: each is capable of running real software
  - Simulate O(1,000) datacenter switches (all levels) with detail and accurate timing
  - Runtime configurable architectural parameters (link speed/latency, host speed)
  - Build on top of RAMP Gold: A full-system FPGA simulator for manycore systems
  - Prototyping with a rack of BEE3 boards





Photos from wikipedia, datacenterknowledge.com and Prof John Wawrzynek

Compare to existing approaches



- RAMP: Simulate O(100) seconds with reasonable amount of time

Research Goals

- Simulate node software with datacenter hardware at O(10,000) scale
  - Help design space exploration in new datacenter designs
- Use the tool to compare and verify existing network designs

Outline

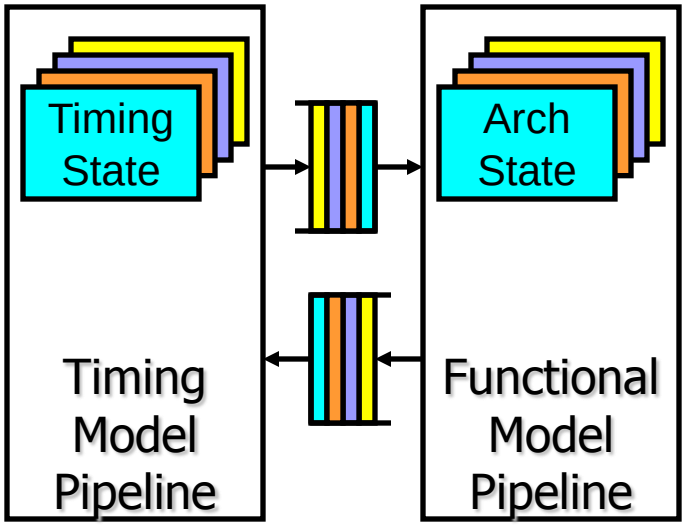
- Datacenter Network Overview
- RAMP Gold
- Modeling Nodes and Network
- A Case Study: Simulating TCP Incast problem
- Related Work
- Conclusion and Questions

RAMP Gold : A full-system manycore emulator

- Leverage RAMP FPGA emulation infrastructure to build prototypes of proposed architectural features
  - Full 32-bit SPARC v8 ISA support, including FP, traps and MMU.
  - Use abstract models with enough detail, but fast enough to run real apps/OS
  - Provide cycle-level accuracy
  - Cost-efficient: hundreds of nodes plus switches on a single FPGA

- Simulation Terminology in RAMP Gold
  - Target vs. Host
    - Target: The system/architecture simulated by RAMP Gold, e.g. servers and switches
    - Host : The platform on which the simulator itself runs, e.g. FPGAs
  - Functional model and timing model
    - Functional: compute instruction result, forward/route packet
    - Timing: CPI, packet processing and routing time

RAMP Gold Key Features

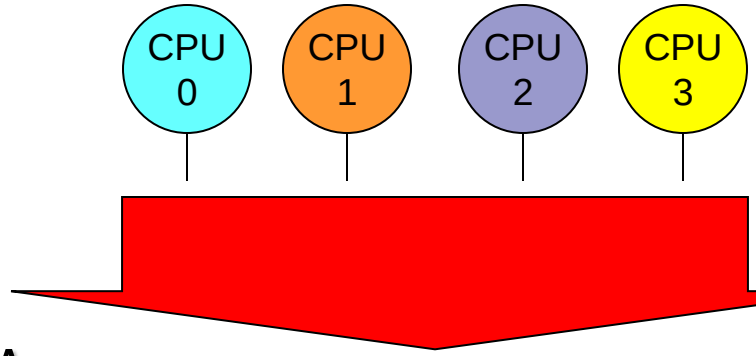


- Abstract RTL not full implementation
- Decoupled functional/timing model, both in hardware
  - Enables many FPGA fabric friendly optimizations
  - Increase modeling efficiency and module reuse
    - E.g. Use the same functional model for 10 Gbps/100 Gbps switches
- Host multithreading of both functional and timing models
  - Hide emulation latencies
  - Time multiplexed effect patched by the timing model

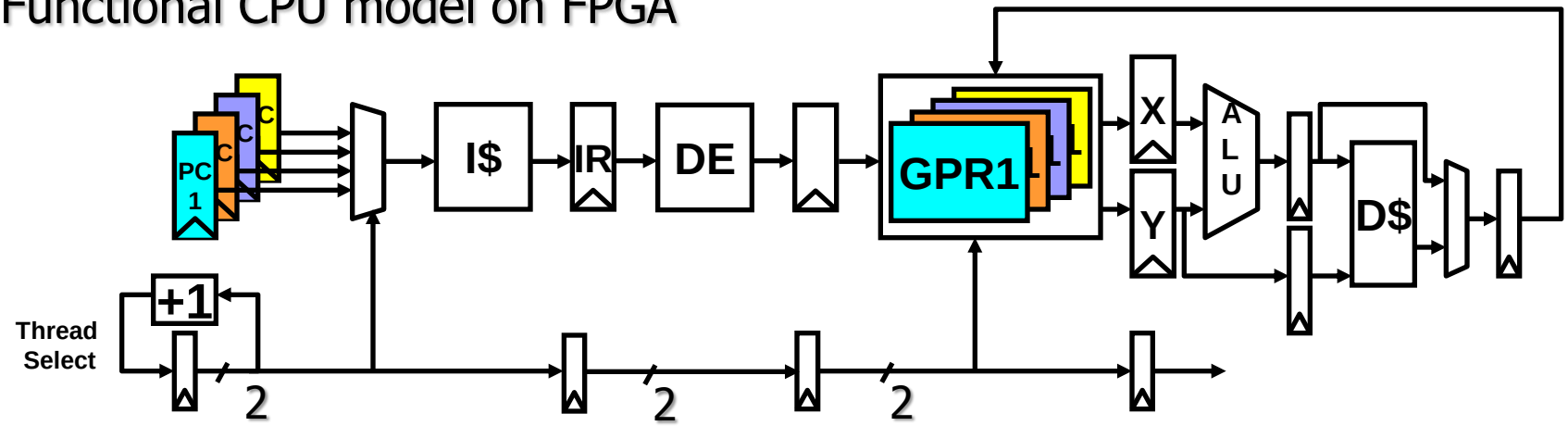
Host multithreading

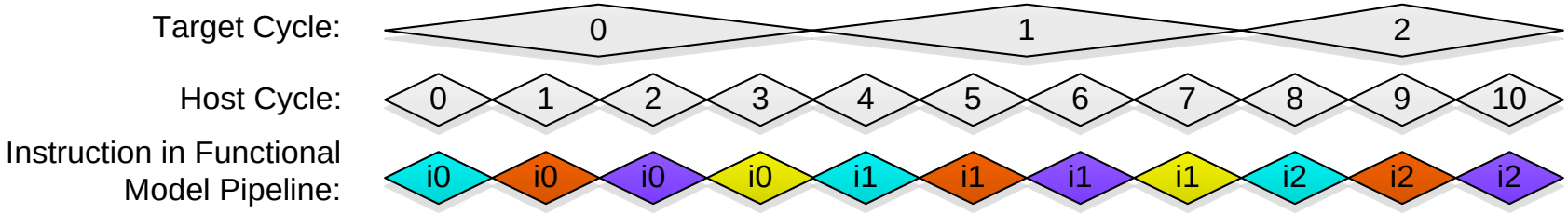
- Example: simulating four independent CPUs

Target Model



Functional CPU model on FPGA





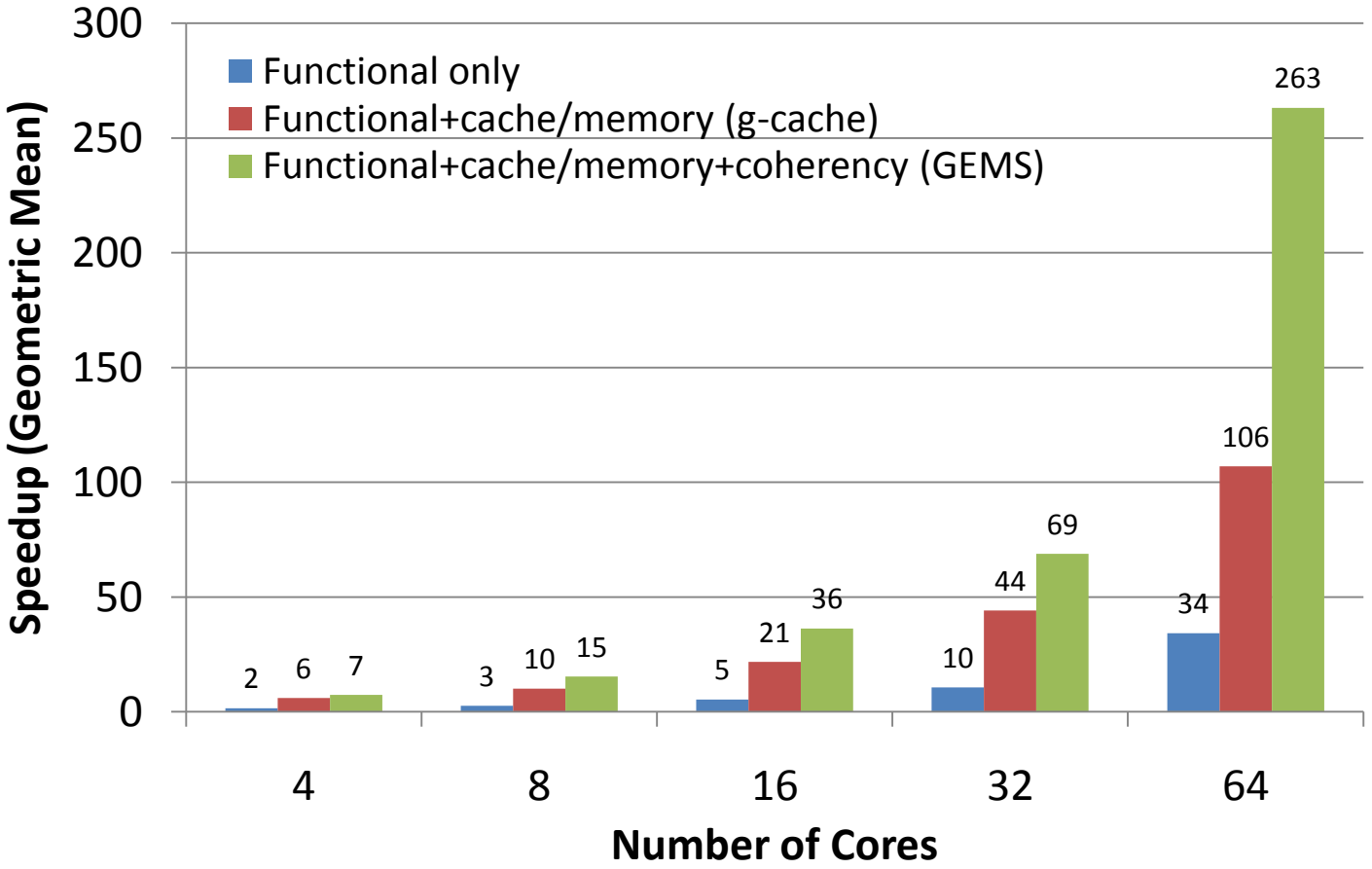
RAMP Gold Implementation

- Single FPGA Implementation (current)
  - $750 Xilinx XUP V5 board
  - 64 cores (single pipeline), 2GB DDR2, FP, processor timing model, ~1M target cycles/second
  - Boot Linux 2.6.21 and Research OS

- Multi-FPGA Implementation for datacenter simulation
  - BEE3 : 4 Xilinx Virtex 5 LX155T
  - ~1.5K cores, 64GB DDR2, FP, timing model
    - Higher emulation capacity and memory bandwidth

RAMP Gold Performance vs Simics

- PARSEC parallel benchmarks running on a research OS
- >250x faster than full system simulator for a 64-core multiprocessor target



Outline

- Datacenter network overview
- RAMP Gold
- Modeling Nodes and Network
- A Case Study: Simulating TCP Incast problem
- Related Work
- Conclusion and Questions

Modeling Servers

- Server model - SPARC v8 ISA with a simple CPU timing model
  - Similar to simulating multiprocessors
    - One functional/timing pipeline simulate up to 64 machines (one rack); fewer threads to improve single thread performance.
    - True concurrency among servers
    - Adjustable core frequency (scaling node performance)
  - Adjustable simulation accuracy
    - Fixed CPI at 1 with a perfect memory hierarchy (default)
    - Can add detailed CPU/memory timing models for points of interest

- Scaling on Virtex 5 LX155T (BEE3 FPGA)
  - ~6 pipelines, 384 servers on one FPGA, 1,536 on one BEE3 board
  - Host memory bandwidth is not a problem
    - <15% peak bandwidth per pipeline
    - dual memory controllers on BEE3

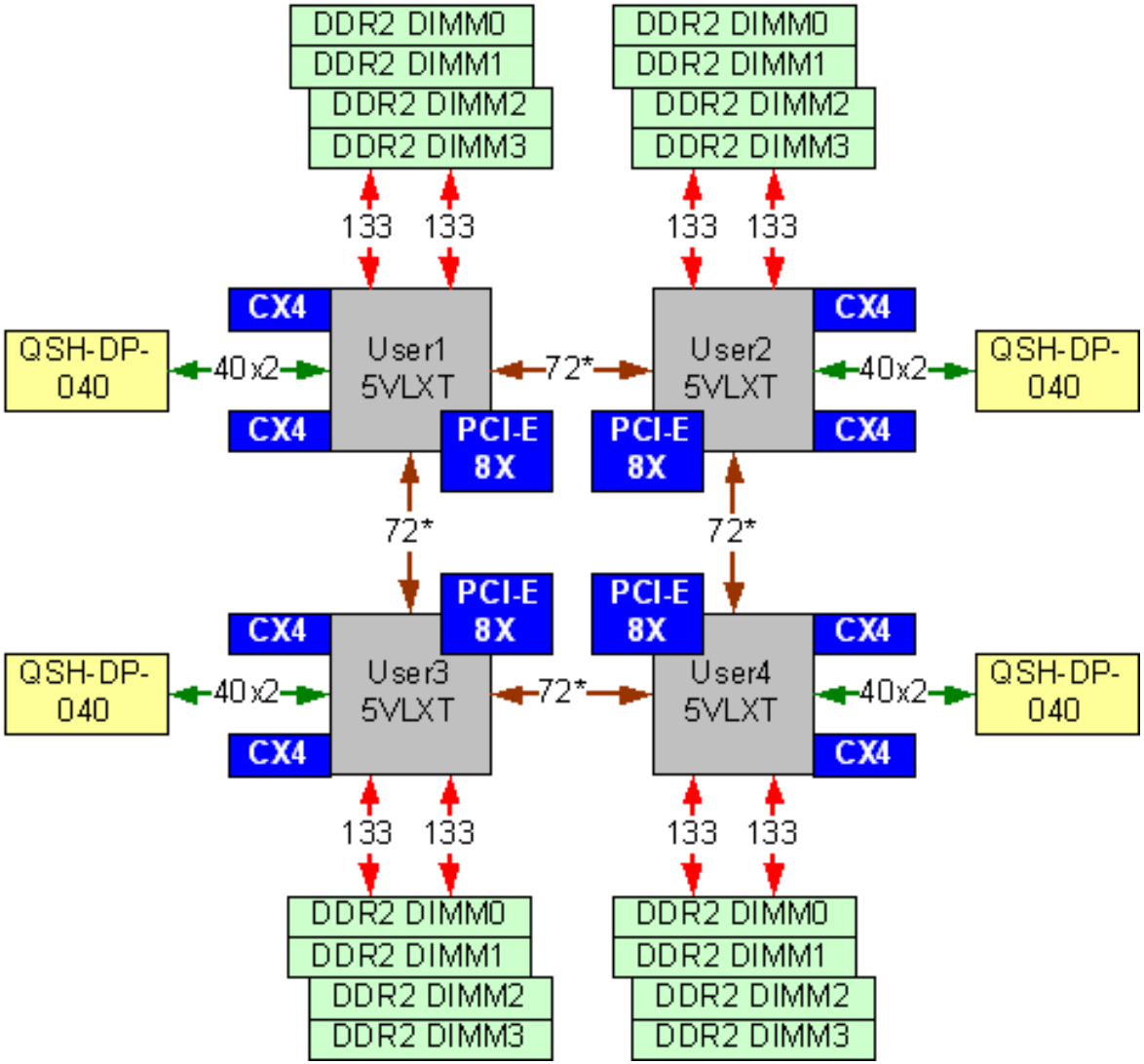
Node Software

- System software per simulated server
  - Debian Linux + Kernel 2.6.21 per node
  - Hadoop on OpenJDK (binary from Debian)
  - LAMP (Linux, Apache, Mysql, PHP)
- Map-reduce Benchmarks (Hadoop Gridmix)
  - Pipelined jobs : common in many user workloads
  - Large sort : processing large dataset
  - Reference select : sampling from a large data set
  - Indirect Read : simulating an interactive job
- Web 2.0 benchmarks, e.g. Cloudstone
- Some research code, e.g. Nexus

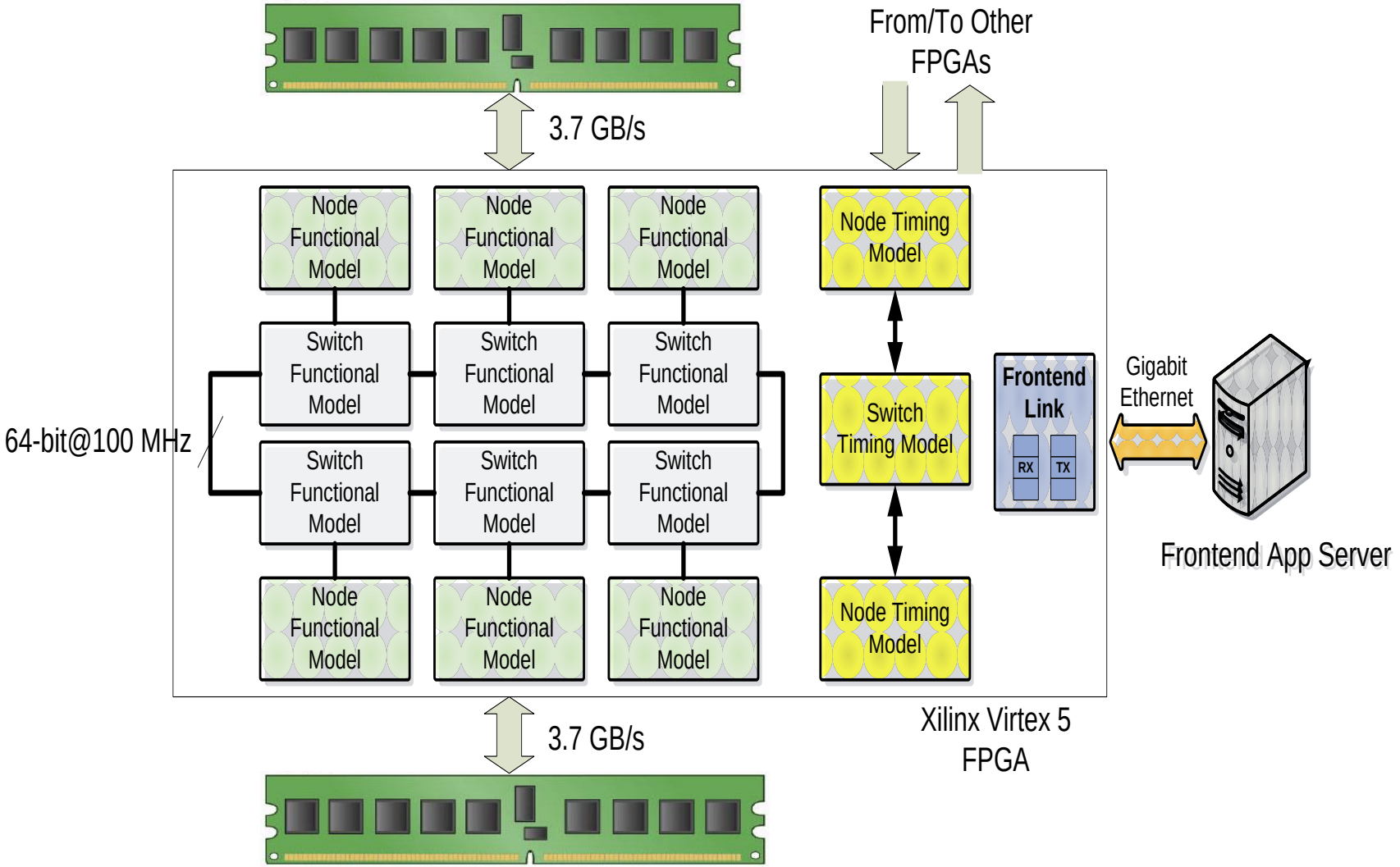
Modeling Network

- Modeling switches and network topology
  - Switch models are also threaded with timing/functional decoupled
  - Start with simple input buffered source-routed switch, then conventional designs
  - Use all-to-all interconnect to simulate arbitrary target topology within one FPGA
  - Runtime configurable parameters without resynthesis
    - Link bandwidth
    - Link delay
    - Switch buffer size
- Estimated switch resource consumption
  - Datacenter switches are "small and simple", e.g.. <10% resource utilization for a real implementation (Farrington HotI'09),
  - abstract model < 1,000 LUTs per switch
  - Using DRAM to simulate switch buffers.

BEE3 : Host Platform



Put everything together



Predicted Performance

- Median map-reduce job length at Facebook (600 machines) and Yahoo!
  - Small and short jobs dominate, 58% at facebook
  - More map tasks than reduce tasks

	Map Task	Reduce Task
Facebook	19 sec	231 sec
Yahoo!	26 sec	76 sec

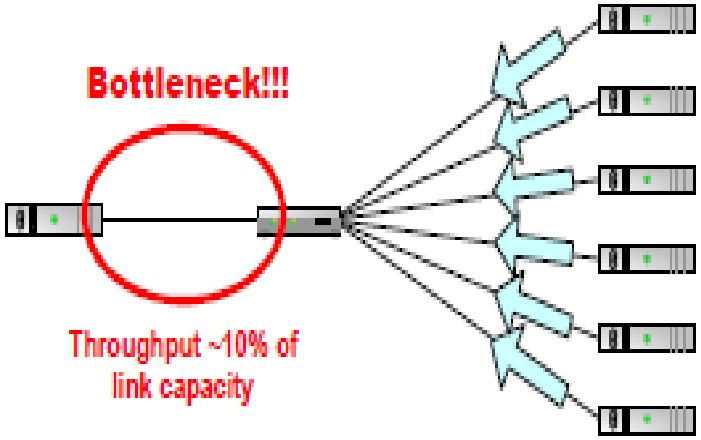
- Simulation time of the median tasks till completion on RAMP Gold

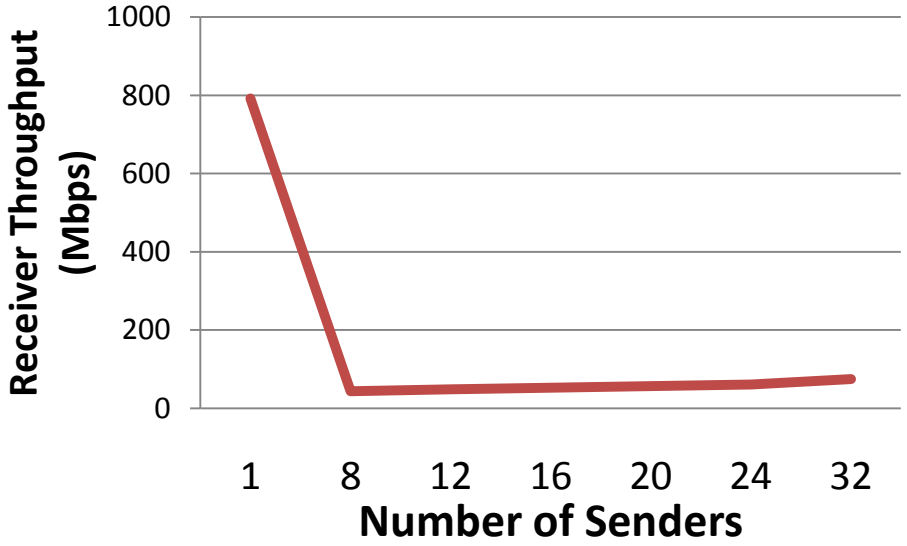
	Map Task	Reduce Task
Facebook (64 threads /pipeline)	5 h	64 h
Yahoo! (64 threads /pipeline)	7 h	21 h
Facebook (16 threads /pipeline)	1 h	16 h
Yahoo! (16 threads /pipeline)	2 h	5 h

Outline

- Datacenter Network Overview
- RAMP Gold
- Modeling Nodes and Network
- A Case Study: Simulating TCP Incast problem
- Related Work
- Conclusion and Questions

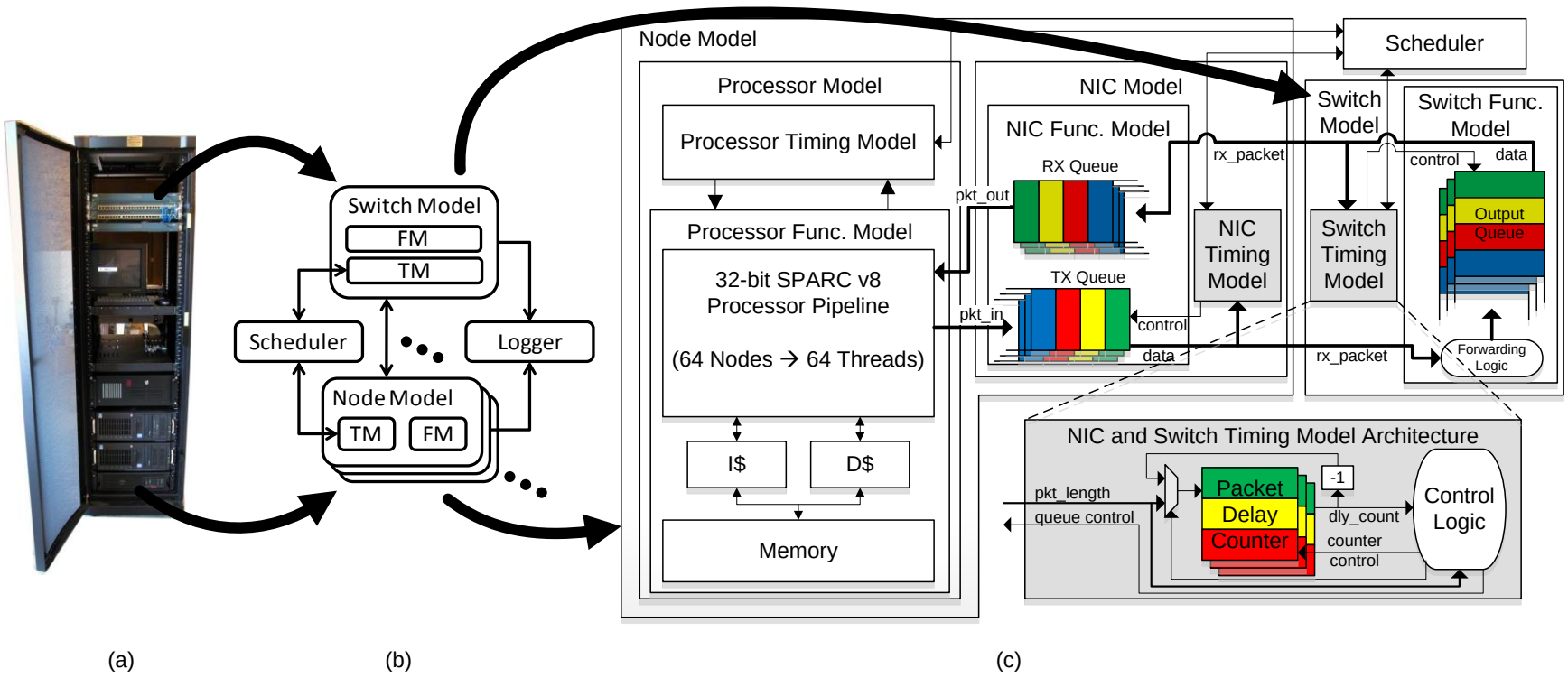
Case study: Reproduce the TCP Incast problem





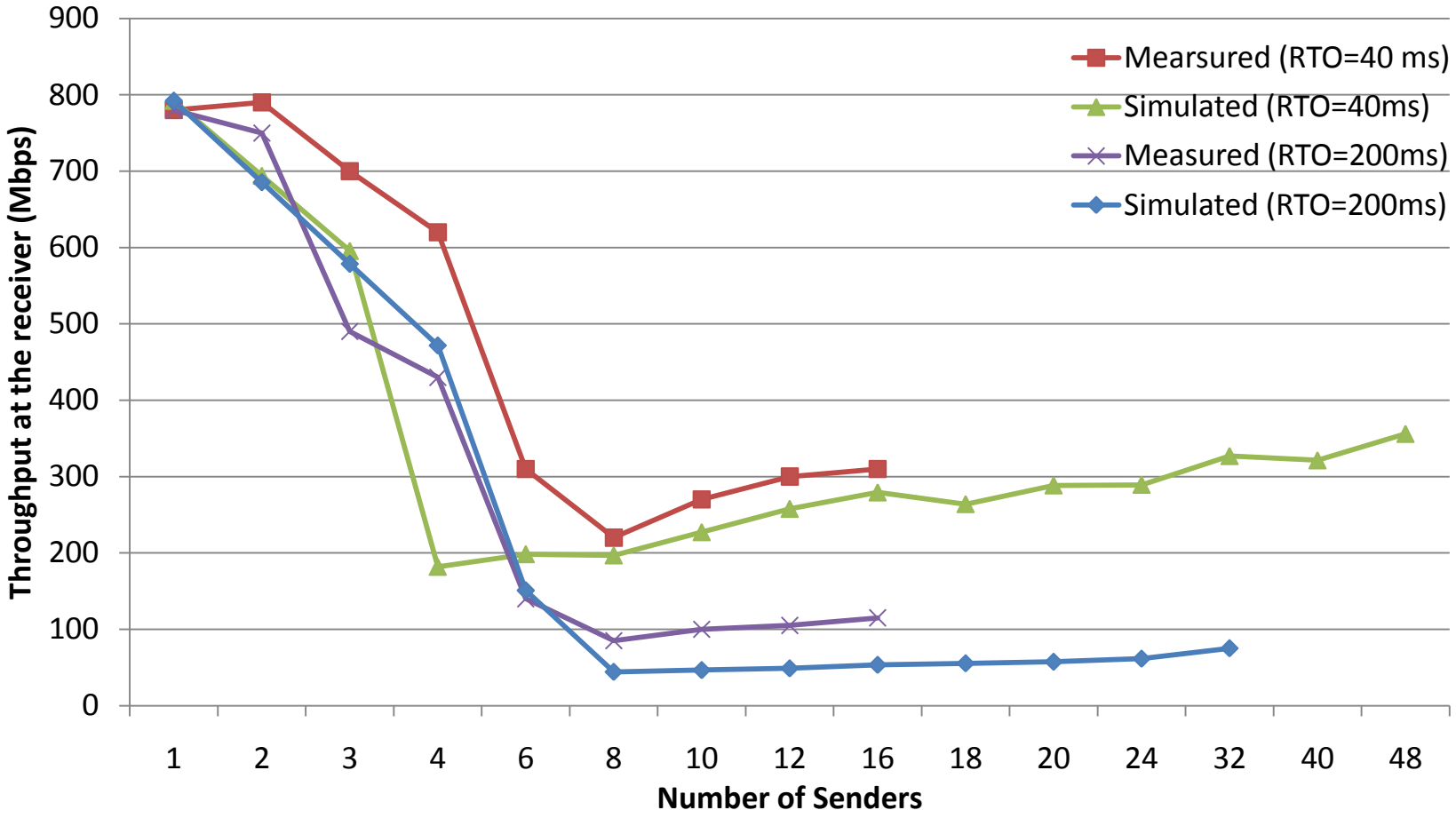
- A TCP throughput collapse that occurs as the number of servers sending data to a client increases past the ability of an Ethernet switch to buffer packets.
  - Safe and Effective Fine-grained TCP Retransmissions for Datacenter Communication. V. Vasudevan. and et al. SIGCOMM'09
  - Original experiments on NS/2 and small scale clusters (<20 machines)

Mapping to RAMP Gold



- Single rack with 64 machines
- One 64-port rack simple output-buffered switch with configurable buffer size (abstract model)

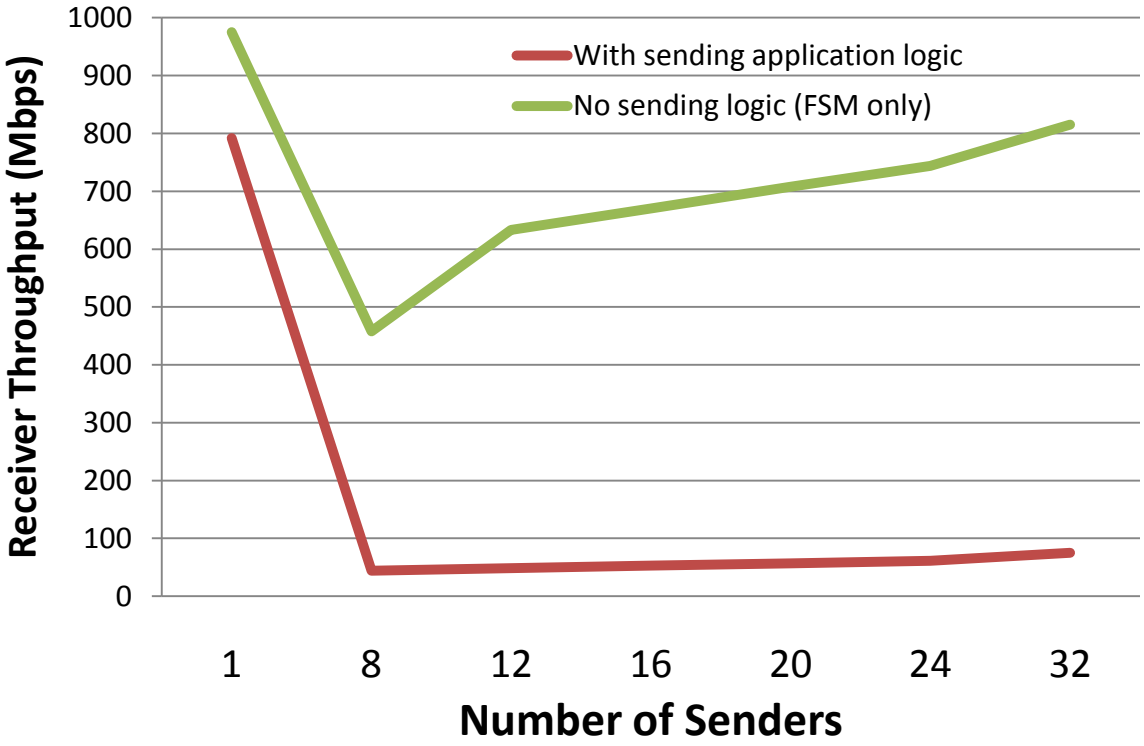
Result: Simulation vs Measurement



- Different in absolute values, but similar curve shapes
  - Off-the-shelf switches are "black-boxes", but abstract switch models work reasonably well

Measured result from, Y. Chen and et al "Understanding TCP Incast Throughput Collapse in Datacenter Networks", Workshop on Research in Enterprise Networking (WREN) 2009, co-located with SIGCOMM 2009

Importance of Node Software



- Simulation configuration: 200ms RTO, 256 KB buffer size
- Node software and application logic may lead to a different result
  - No throughput collapse observed with more FSM senders
  - Different curve shapes, absolute difference : 5-8x

Outline

- Datacenter Network Overview
- RAMP Gold
- Modeling Nodes and Network
- A Case Study: Simulating TCP Incast problem
- Related Work
- Conclusion and Questions

Novel Datacenter Network Architecture

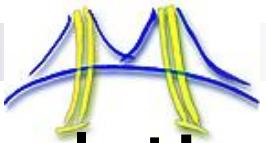
- R. N. Mysore and et al. "PortLand: A Scalable Fault-Tolerant Layer 2 Data Center Network Fabric", SIGCOMM 2009, Barcelona, Spain
- A. Greenberg and et al. "VL2: A Scalable and Flexible Data Center Network", SIGCOMM 2009, Barcelona, Spain
- C. Guo and et al. "BCube: A High Performance, Server-centric Network Architecture for Modular Data Centers", SIGCOMM 2009, Barcelona, Spain
- A. Tavakoli and et al, "Applying NOX to the Datacenter", HotNets-VIII, Oct 2009
- D. Joseph, A. Tavakoli, I. Stoica, A Policy-Aware Switching Layer for Data Centers, *SIGCOMM 2008* Seattle, WA
- M. Al-Fares, A. Loukissas, A. Vahdat, A Scalable, "Commodity Data Center Network Architecture", *SIGCOMM 2008* Seattle, WA
- C. Guo and et al., "DCell: A Scalable and Fault-Tolerant Network Structure for Data Centers", *SIGCOMM 2008* Seattle, WA
- The OpenFlow Switch Consortium, www.openflowswitch.org
- N. Farrington and et al. "Data Center Switch Architecture in the Age of Merchant Silicon", *IEEE Symposium on Hot Interconnects, 2009*

Software Architecture Simulation

- Full-system software simulators for timing simulation
  - N. L. Binkert and et al., "The M5 Simulator: Modeling Networked Systems", IEEE Micro, vol. 26, no. 4, July/August, 2006
  - P. S. Magnusson, "Simics: A Full System Simulation Platform. IEEE Computer", 35, 2002.
- Multiprocessor parallel software architecture simulators
  - S. K. Reinhardt and et al. "The Wisconsin Wind Tunnel: virtual prototyping of parallel computers. SIGMETRICS Perform. Eval. Rev., 21(1):48–60, 1993"
  - J. E. Miller and et al. "Graphite: A Distributed Parallel Simulator for Multicores", HPCA-10, 2010
- Other network and system simulators
  - The Network Simulator - ns-2, www.isi.edu/nsnam/ns/
  - D. Gupta and et al. "DieCast: Testing Distributed Systems with an Accurate Scale Model", NSDI'08, San Francisco, CA 2008
  - D. Gupta and et al. "To Infinity and Beyond: Time-Warped Network Emulation" , NSDI'06, San Jose, CA 2006

RAMP related simulators for multiprocessors

- Multithreaded functional simulation
  - E. S. Chung and et al. "ProtoFlex: Towards Scalable, Full-System Multiprocessor Simulations Using FPGAs", ACM Trans. Reconfigurable Technol. Syst., 2009
- Decoupled functional/timing model
  - D. Chiou and et al. "FPGA-Accelerated Simulation Technologies (FAST): Fast, Full-System, Cycle-Accurate Simulators", MICRO'07
  - N. Dave and et al. "Implementing a Functional/Timing Partitioned Microprocessor Simulator with an FPGA", WARP workshop '06.
- FPGA prototyping with limited timing parameters
  - A. Krasnov and et al. "RAMP Blue: A Message-Passing Manycore System In FPGAs", Field Programmable Logic and Applications (FPL), 2007
  - M. Wehner and et al. "Towards Ultra-High Resolution Models of Climate and Weather", International Journal of High Performance Computing Applications, 22(2), 2008



RAMP Gold and Datacenter Simulation

- Zhangxi Tan, Andrew Waterman, Henry Cook, Sarah Bird, Krste Asanović, David Patterson, "A Case for FAME: FPGA Architecture Model Execution", To appear, International Symposium on Computer Architecture (ISCA-2010), Saint-Malo, France, June 2010.
- Zhangxi Tan, Andrew Waterman, Rimas Avizienis, Yunsup Lee, Henry Cook, Krste Asanović, David Patterson, "RAMP Gold: An FPGA-based Architecture Simulator for Multiprocessors" To appear, Design Automation Conference (DAC-2010), Anaheim, CA, June 2010
- Zhangxi Tan, Krste Asanović, and David Patterson, "An FPGA-Based Simulator for Datacenter Networks", The Exascale Evaluation and Research Techniques Workshop (EXERT 2010), at the 15th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2010), Pittsburgh, PA, March 2010.

Conclusion

- Simulating datacenter network architecture is not only a networking problem
  - Real node software significantly affects the result even at the rack level
  - RAMP enables running real node software: Hadoop, LAMP
- RAMP will improve the scale of evaluation and accuracy
  - Will be promising for container-level experiments
  - FPGAs scale as the Moore's law
    - Fit ~30 pipelines on the largest 45nm Virtex 6
  - Help to evaluate protocol/software at scale

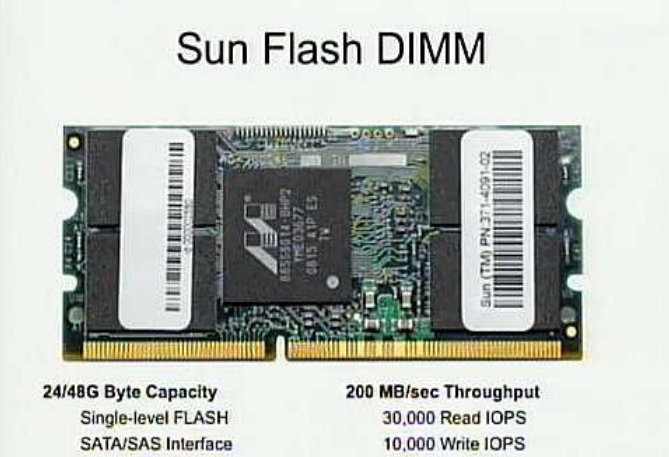
Questions I'd like to ask

- Modeling multicore effect in detail for sending large payload at 10 Gbps?
  - Alternatively, scale up single core performance by lowering CPI?
    - E.g. One single core 4 GHz CPU model to replace dual-core 2 GHz CPU model
- Performance and memory capacity requirement if using the simulator for cluster application development
  - What is the maximum acceptable slowdown?
  - What is the typical memory requirement?
- Is O(1,000) interesting enough (single BEE3 board)? Is O(10,000) a must-have feature?

Backup Slides

Memory capacity

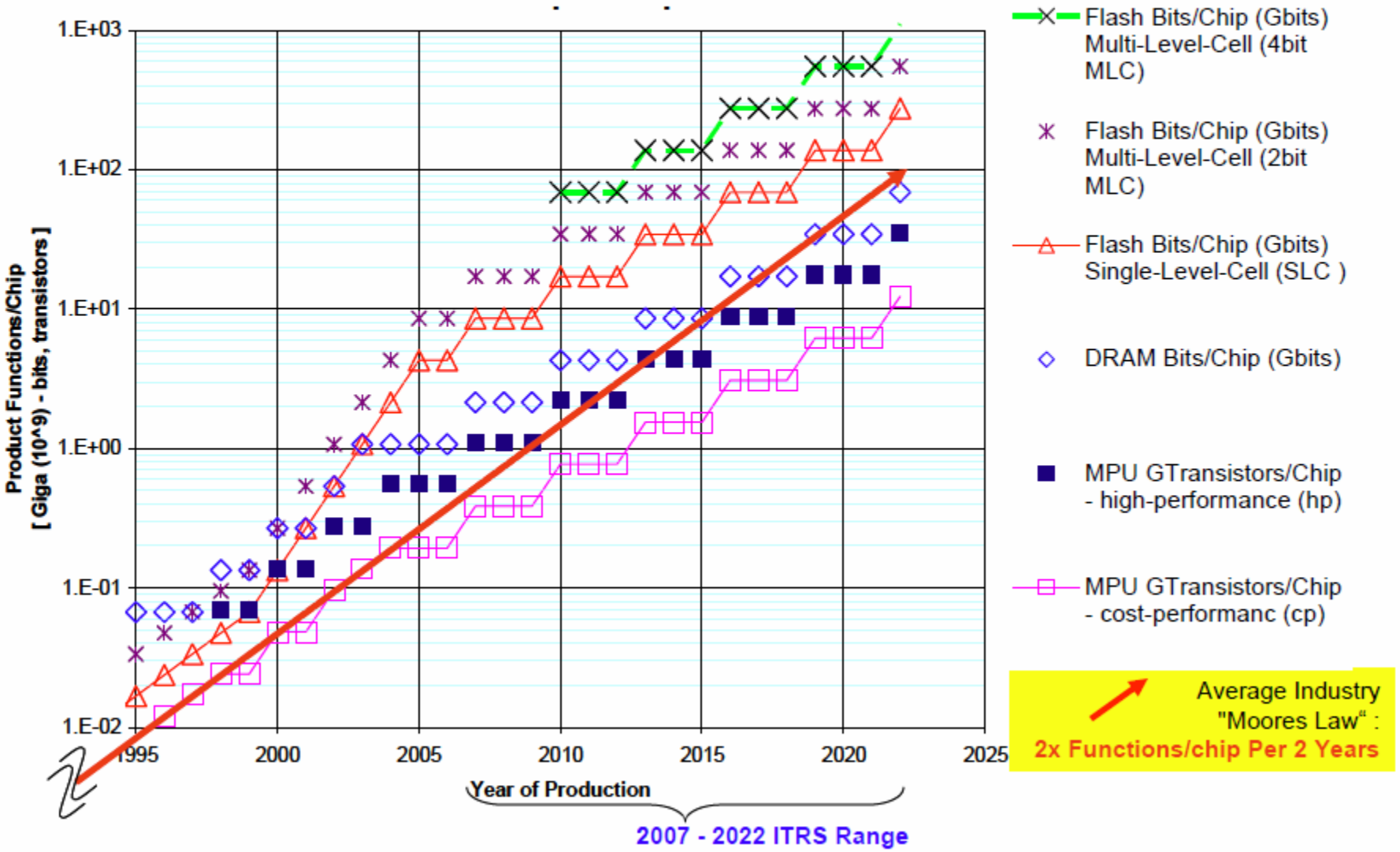
- Hadoop benchmark memory footprint
  - Typical configuration: JVM allocate ~200 MB per node
- Share memory pages across simulated servers



BEE3 Flash DIMM

- Use Flash DIMMs as extended memory
  - Sun Flash DIMM : 50 GB
  - BEE3 Flash DIMM : 32 GB DRAM Cache + 256 GB SLC Flash
- Use SSD as extended memory
  - 1 TB @ $2000, 200us write latency

Semiconductor Technology Roadmap



OpenFlow switch support

- The key is to simulate 1K-4K TCAM flow tables/port
  - Fully associative search in one cycle, similar to TLB simulation in multiprocessors
  - Functional/timing split simplifies the functionality : On-chip SRAM $ + DRAM hash tables
  - Flow tables are in DRAM, so can be easily updated using either HW or SW

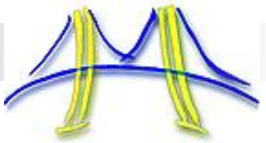
- Emulation capacity (if we only want switches)
  - Single 64-port switch with 4K TCAM /port and 256KB port buffer
    - Requires 24 MB DRAM
  - ~100 switches/one BEE3 FPGA, ~400 switches/board
    - Limited by the SRAM $ of TCAM flow tables

How did people do evaluation recently?

Novel Architectures	Evaluation	Scale/simulation time	latency	Application
Policy-away switching layer	Click software router	Single switch	Software	Microbenchmark
DCell	Testbed with exising HW	~20 nodes / 4500 sec	1 Gbps	Synthetic workload
Portland (v1)	Click software router + exiting switch HW	20 switches and 16 end hosts (36 VMs on 10 physical machines)	1 Gbps	Microbenchmark
Portland (v2)	Testbed with exising HW + NetFPGA	20 Openflow switches and 16 end-hosts / 50 sec	1 Gbps	Synthetic workload + VM migration
BCube	Testbed with exising HW + NetFPGA	16 hosts + 8 switches /350 sec	1 Gbps	Microbenchmark
VL2	Testbed with existing HW	80 hosts + 10 switches / 600 sec	1 Gbps + 10 Gbps	Microbenchmark
Chuck Thack's Container network	Prototyping with BEE3	-	1 Gbps + 10 Gbps	Traces

Additional Plan

- Scale to O(10,000) with 10 BEE3 boards
- Add a storage timing model
- Add switch power models
- Modeling multicore effects
- Improve per-node memory capacity (DRAM caching + FLASH)
- Make it faster ☺



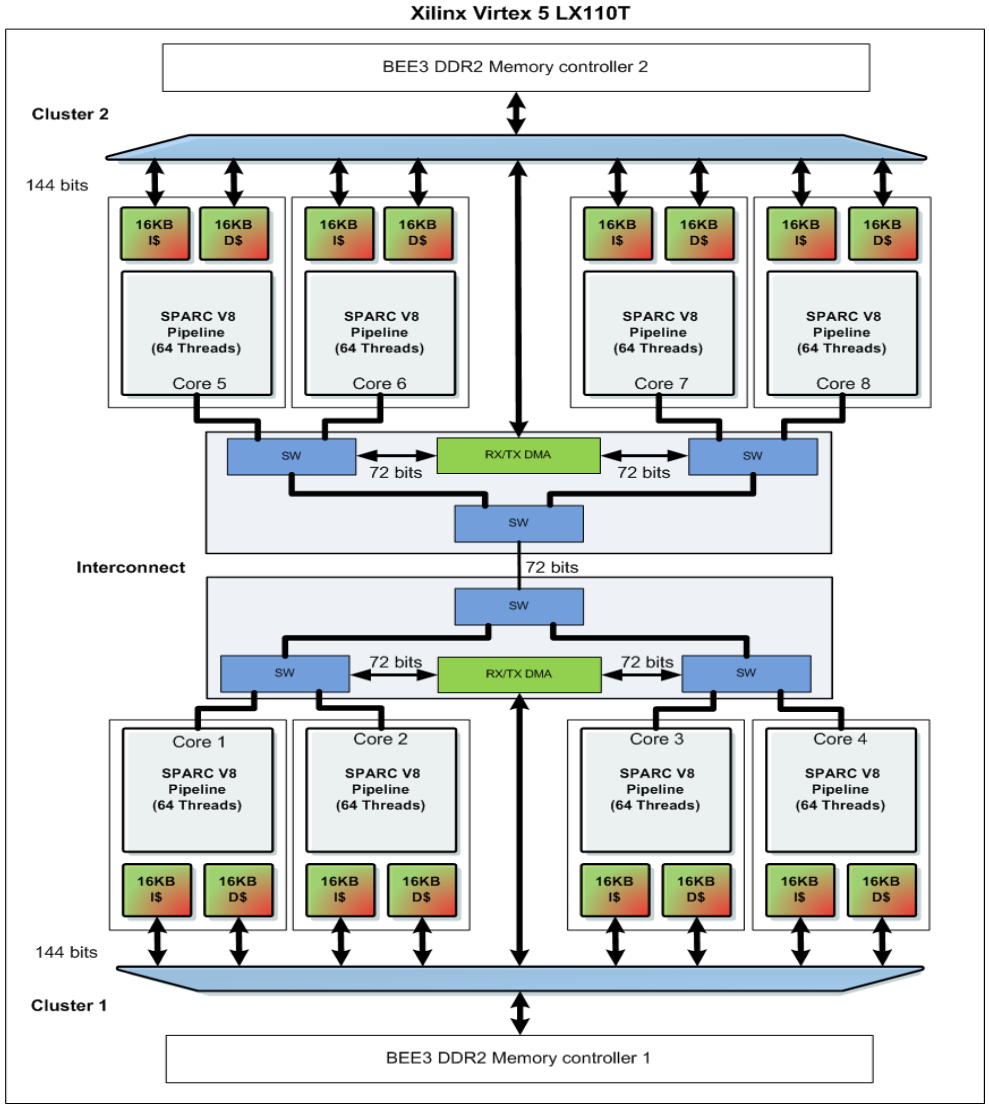
CPU Functional Model (1)

- 64 HW threads, full 32-bit SPARC v8 CPU
  - The same binary runs on both SUN boxes and RAMP
  - Optimized for emulation throughput (MIPS/FPGA)
  - 1 cycle access latency for most of the instructions on host
  - Microcode operation for complex and new instructions
    - E.g. trap, active messages
- Design for FPGA fabric for optimal performance
  - "Deep" pipeline : 11 physical stages, no bypassing network
  - DSP based ALU
  - ECC/parity protected RAM/cache lines and etc.
  - Double clocked BRAM/LUTRAM
  - Fine-tuned FPGA resource mapping

State storage

- Complete 32-bit SPARC v8 ISA w. traps/exceptions
- All CPU states (integer only) are stored in SRAMs on FPGA
  - Per context register file -- BRAM
    - 3 register windows stored in BRAM chunks of 64
      - 8 (global) + 3*16 (reg window) = 54
  - 6 special registers
    - pc/npc -- LUTRAM
    - PSR (Processor state register) -- LUTRAM
    - WIM (Register Window Mask) -- LUTRAM
    - Y (High 32-bit result for MUL/DIV) -- LUTRAM
    - TBR (Trap based registers) -- BRAM (packed with regfile)
- Buffers for host multithreading (LUTRAM)
- Maximum 64 threads per pipeline on Xilinx Virtex5
  - Bounded by LUTRAM depth (6-input LUTs)

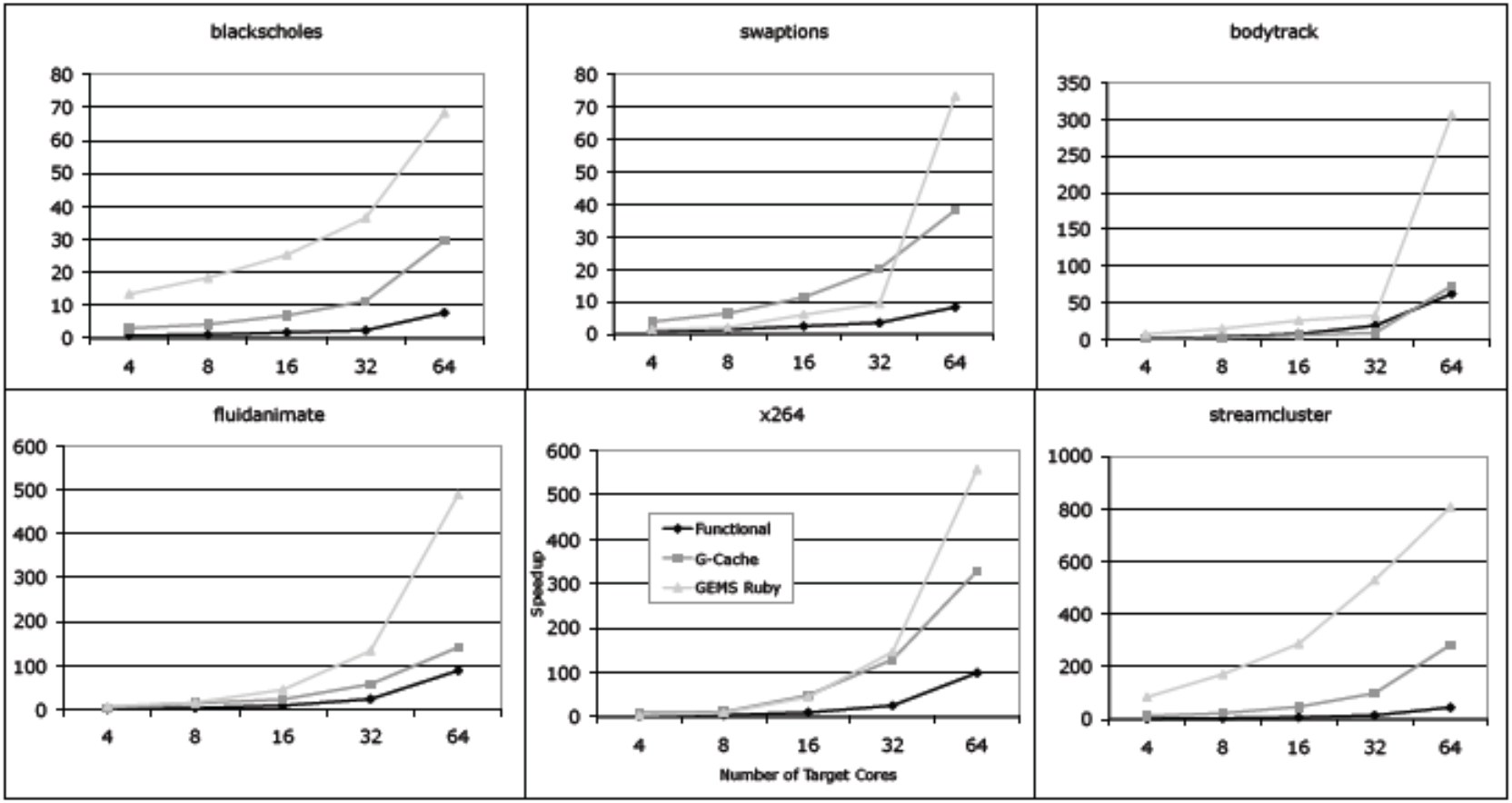
Example: A distributed memory non-cache coherent system



- Eight multithreaded SPARC v8 pipelines in two clusters
  - Each thread emulates one independent node in target system
  - 512 nodes/FPGA
  - Predicted emulation performance:
    - ~1 GIPS/FPGA (10% I$ miss, 30% D$ miss, 30% LD/ST)
    - x2 compared to naïve manycore implementation
- Memory subsystem
  - Total memory capacity 16 GB, 32MB/node (512 nodes)
  - One DDR2 memory controller per cluster
  - Per FPGA bandwidth: 7.2 GB/s
  - Memory space is partitioned to emulate distributed memory system
  - 144-bit wide credit-based memory network
- Inter-node communication (under development)
  - Two-level tree network with DMA to provide all-to-all communication

RAMP Gold Performance vs Simics

- PARSEC parallel benchmarks running on a research OS
- 269x faster than full system simulator@ 64-core configuration



General Datacenter Network Research

- Chuck Thacker, "Rethinking data centers", Oct 2007
- James Hamilton, "Data Center Networks Are in my Way", Clean Slate CTO Summit, Stanford, CA Oct 2009.
- Albert Greenberg, James Hamilton, David A. Maltz, Parveen Patel, "The Cost of a Cloud: Research Problems in Data Center Networks", ACM SIGCOMM Computer Communications Review, Feb. 2009
- James Hamilton, "Internet-Scale Service Infrastructure Efficiency", Keynote, ISCA 2009, June, Austin, TX