Research Challenges for Modern Data-Center Networks

Jeff Mogul
HP Labs Fellow
jeff.mogul@hp.com

Keynote 3:
23rd International Teletraffic Congress
San Francisco, CA
8 September 2011

Compared to stuff like …

– Energy Savings in Cellular Access Networks
– Content-centric Networks
– Optical Burst Switching
– Million-node Wireless Sensor Networks

Data-center networks are boring

– right?

Right!

Right!

Except for

Right!

Except for

- Scale

Right!

Except for

- Scale
- Performance

Right!

Except for

- Scale
- Performance
- Power

Right!

Except for

- Scale
- Performance
- Power
- Reliability

Right!

Except for

- Scale
- Performance
- Power
- Reliability
- Security

Right!

Except for

- Scale
- Performance
- Power
- Reliability
- Security
- Cost

Right!

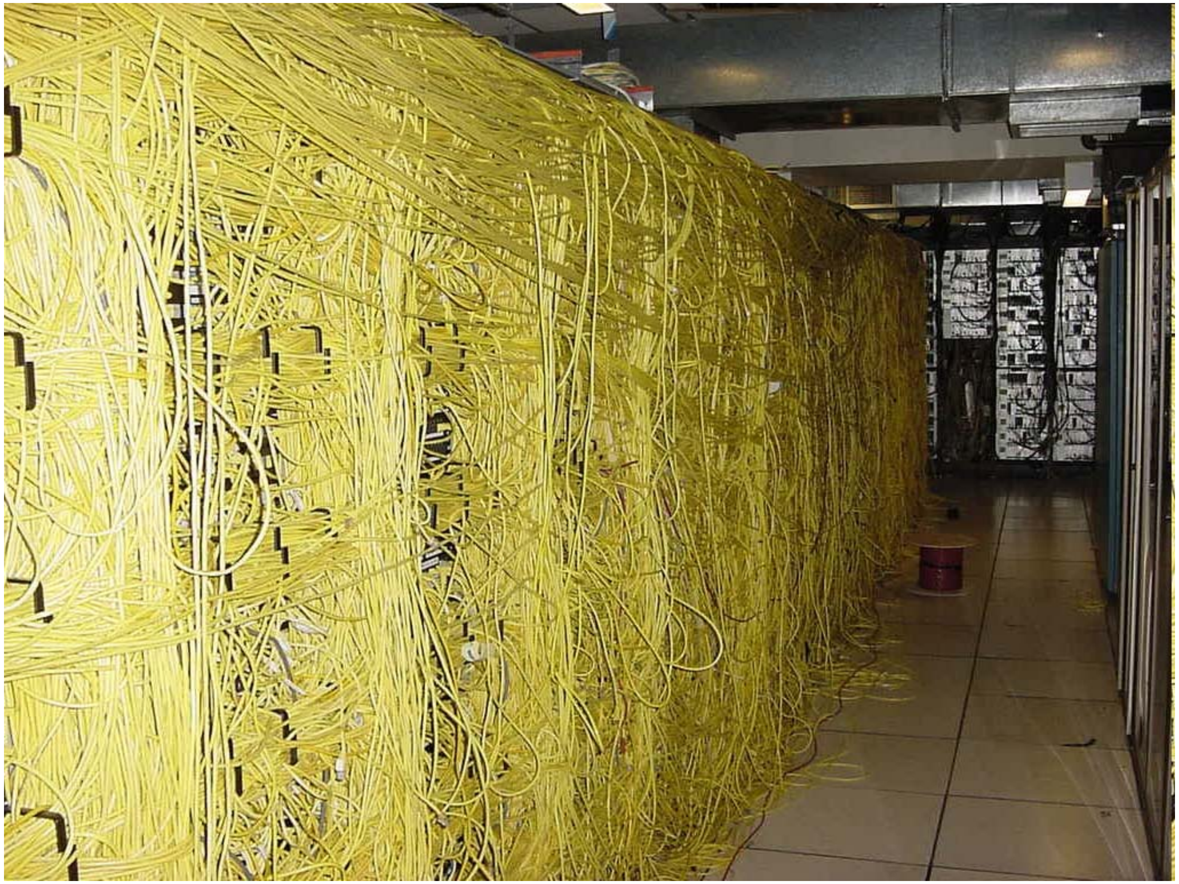
Except for

- Scale
- Performance
- Power
- Reliability
- Security
- Cost
- Management

Right!

Except for

- Scale
- Performance
- Power
- Reliability
- Security
- Cost
- Management
- and some other stuff



Right!

Except for

- Scale
- Performance
- Power
- Reliability
- Security
- Cost
- Management
- and some other stuff

There really aren't that many interesting research problems to solve for data-center networks

Right!

Except for

- Scale
- Performance
- Power
- Reliability
- Security
- Cost
- Management
- and some other stuff

There really aren't that many interesting research problems to solve for data-center networks

But there are enough to keep me talking for 45 minutes or so

If you're taking notes …

All references to published papers are hyperlinked

- So, no need to write down the titles/authors of papers
- just download the PDF of my slides, and click

Some specific challenges

- What do applications actually need?
- Scaling to 100K nodes or more
- Isolation in multi-tenant/cloud data-centers
- What's the right physical layer?
- Reducing network-related energy
- Simplifying network management
- Fault detection and diagnosis
- Challenges of future applications

- What do applications actually need?
- Scaling to 100K nodes or more
- Isolation in multi-tenant/cloud data-centers
- What's the right physical layer?
- Reducing network-related energy
- Simplifying network management
- Fault detection and diagnosis
- Challenges of future applications

What does a data center do, anyway?

- Legacy data centers:
  - Lots of relatively isolated apps,
    - such as 3-tier Web applications or large databases
  - Apps are typically tied to a specific set of servers
  - Storage traffic often goes over a non-Ethernet network

But these aren't of much interest for researchers

- New/future data centers:
  - Lots of interconnected (Web services) apps
  - Apps flex rapidly, and are not tied to any specific server
  - Storage traffic shares a "converged fabric" with data traffic
  - Scale matters: amortizing fixed costs over lots of uses

There's more than one kind of application

…and they have different needs.

Some examples

- Data analytics/Search
  - Needs high bandwidth, often all-to-all
  - also needs balanced bandwidth, or else you get stragglers
- High Performance Computing (HPC)
  - Needs really low latency (a few microseconds) + low jitter
- Web services/hosting
  - Needs lots of bandwidth to the Internet, + DDOS protection
  - Want to migrate VMs without having to change addresses

What do applications do with the network?

That is, *what kinds of communication patterns will we see in data-center networks*?

- Answer: we don't really know!
- It's hard to collect detailed traffic data at this scale
- People who have it don't like to share it
- Measurements from one DC might not apply to another

What we do know

Microsoft researchers have published some useful papers:

- VL2: A Scalable and Flexible DC Network (SIGCOMM '09)
  - Traffic matrices are variable and unpredictable
  - Most flows are short; most bytes are in the rare long flows
- The Nature of Datacenter Traffic (IMC '09)
  - Several patterns: *work-seeks-bandwidth* and *scatter-gather*
  - Even elephant flows don't last very long

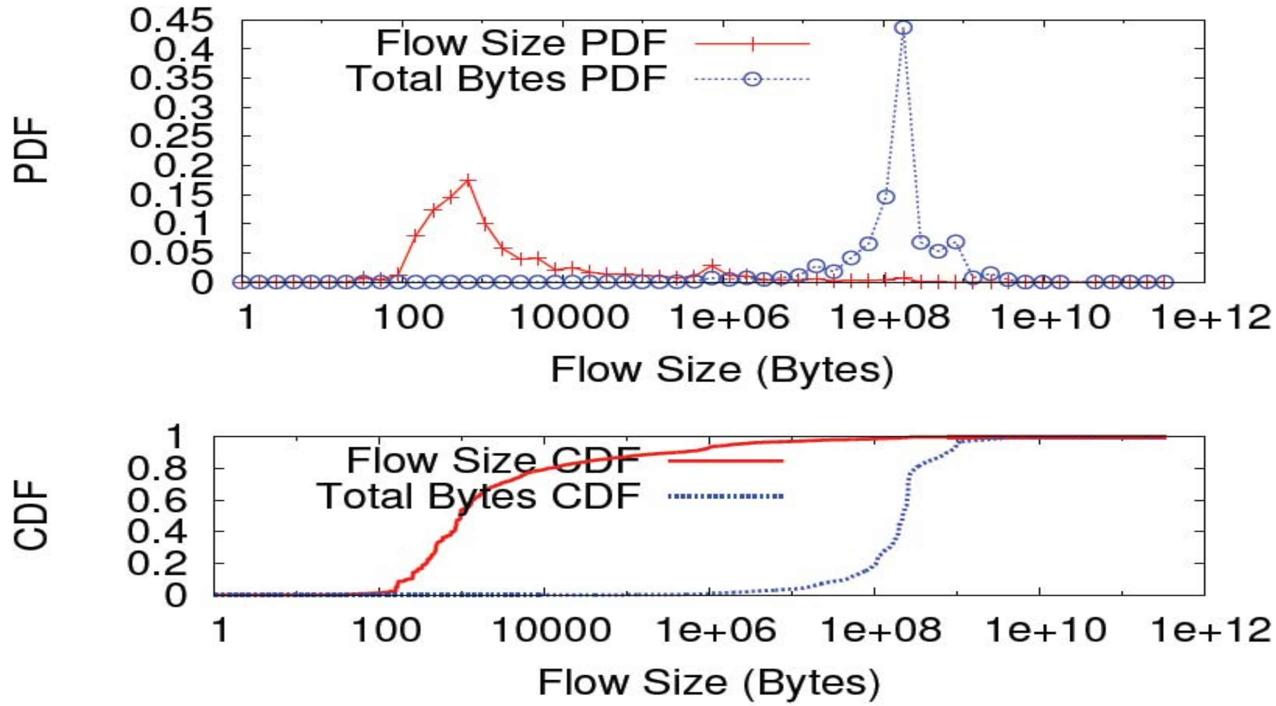


Figure 2: Mice are numerous; 99% of flows are smaller than 100 MB. However, more than 90% of bytes are in flows between 100 MB and 1 GB.

Reprinted from "**VL2: A Scalable and Flexible Data Center Network**," Albert Greenberg, James R. Hamilton, Navendu Jain, Srikanth Kandula, Changhoon Kim, Parantap Lahiri, David A. Maltz, Parveen Patel, Sudipta Sengupta, *Proc. SIGCOMM 2009*

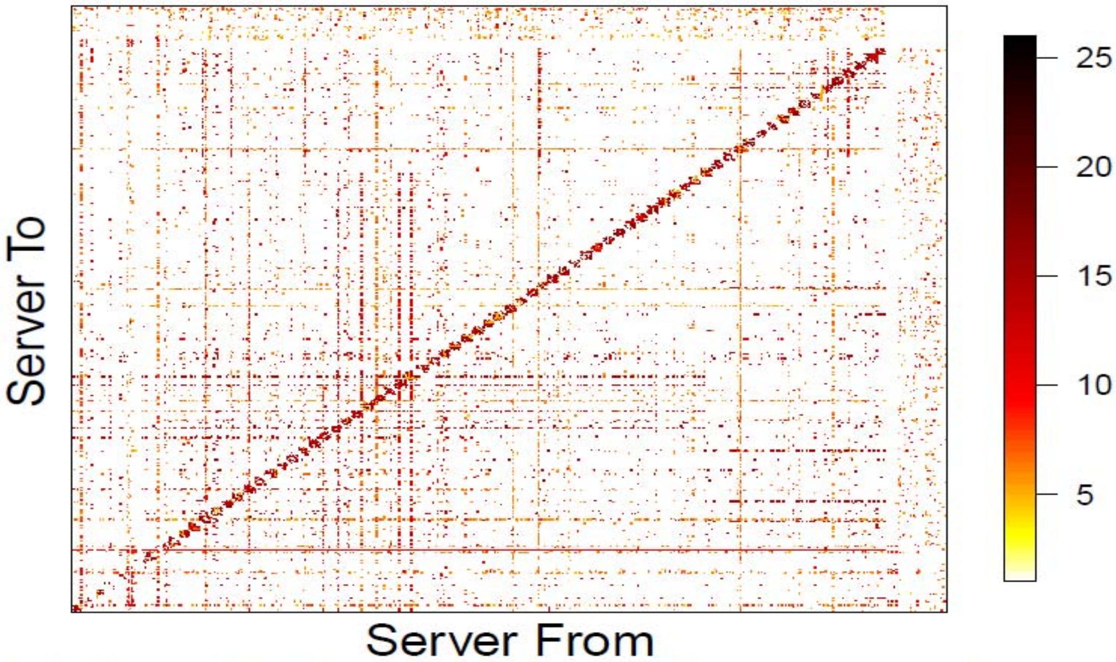


Figure 2: The *Work-Seeks-Bandwidth* **and** *Scatter-Gather* **patterns in datacenter traffic as seen in a matrix of** $\log_e$ **(Bytes) exchanged between server pairs in a representative 10s period. (See §4.1).**

Reprinted from "**The Nature of Datacenter Traffic: Measurements & Analysis**", Srikanth Kandula, Sudipta Sengupta, Albert Greenberg, Parveen Patel, Ronnie Chaiken, *Proc. IMC 2009*

So, what do applications really need?

Some combination of:

- High bandwidth, internally and externally
- Balanced bandwidth
- Predictable bandwidth
- All-to-all or tree-like flow patterns
- Low latency and jitter
- Flat L2 networks at large scale
  - How large? It depends …
- DDOS protection and firewalls

- What do applications actually need?
- Scaling to 100K nodes or more
- Isolation in multi-tenant/cloud data-centers
- What's the right physical layer?
- Reducing network-related energy
- Simplifying network management
- Fault detection and diagnosis
- Challenges of future applications

Scaling to really big networks

Some DC operators want L2 networks with 100K nodes or more (and lots of VMs per node)

- Others are OK with smaller "availability zones"

This creates pressure on:

- Ethernet learning tables
- Multicast group tables
- Access control lists in switches
- Broadcast/multicast load and storms
- End-to-end latency
- Power, cooling, and network administration

←This?

Or this?→

Scaling an L2 network while using cheap switches

- Problem: small Ethernet learning tables:
  - Either encapsulate or rewrite packets, to hide VM addresses from core switches
    - Encapsulate tenant packets: NetLord (SIGCOMM 2011)
    - Rewrite tenant packet headers: Diverter (WREN 2009)
  - Do this in hypervisor or edge switch?
  - Creates a need to distribute ARP-like data efficiently
- Problem: small ACL tables:
  - Not enough space to have one rule per tenant VM
  - NetLord's solution: expose Tenant-ID in encapsulation hdr

Scaling multicasts and broadcasts

Broadcasting and multicasting – the main limits to L2 scale?

- Per-node overhead scales with broadcast-domain size
- Broadcast-storm damage scales with domain size
- Switches have smallish IGMP tables
- Conventional wisdom: domain limit "a few hundred" nodes

What causes broadcasts/multicasts in data-centers?

- ARP and DHCP; Ethernet flooding
  - Solution: don't do it: e.g., SEATTLE (SIGCOMM '08), NetLord (SIGCOMM '11)
- Application-level multicasts
  - Don't allow these (e.g., EC2); or optimize multicast distribution (hp, in progress)

- What do applications actually need?
- Scaling to 100K nodes or more
- **Isolation in multi-tenant/cloud data-centers**
- What's the right physical layer?
- Reducing network-related energy
- Simplifying network management
- Fault detection and diagnosis
- Challenges of future applications

Isolation in multi-tenant/cloud data-centers

Functional isolation is simple; performance isolation is hard:

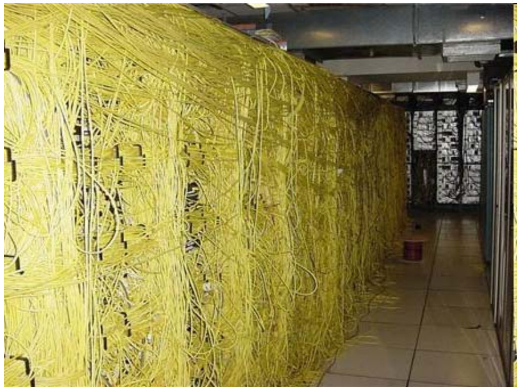
- What guarantees do tenants want?
- What can the network feasibly provide?
  - Considering: scale, cost, efficiency/offload, and security against cheating
- Several proposals for cloud tenant models:
  - Distributed Rate Limiting (SIGCOMM '07): global limit on sum of tenant's traffic
  - SeaWall (NSDI '11): limits on pair-wise inter-VM flows
  - Topology Switching (Hot-ICE '11): can spec. max # of link-sharers
- Massive scale makes this hard:
  - Not enough hardware rate limiters in switches
  - Economics forces resource multiplexing (conflicts with isolation/predictability)

- What do applications actually need?
- Scaling to 100K nodes or more
- Isolation in multi-tenant/cloud data-centers
- **What's the right physical layer?**
- Reducing network-related energy
- Simplifying network management
- Fault detection and diagnosis
- Challenges of future applications

Physical-layer challenges

- A few big switches, or lots of little switches?
  - Big switches: expensive, power-hungry, but easier to manage
  - Small switches: cheaper, cooler, but lots of wires and things to watch
- What's the wiring topology?
- What are the "wires?"
- How much do we want to do at L2?

Wiring topology challenges



You can get by with cheaper, cooler switches

- by creating multi-path topologies
- But then you need a lot of them, and a lot of wires
- What's the best arrangement?
  - Tree? Clos/fat-tree? HyperCube/BCube? HyperX? Scale-free?
- Things to worry about:
  - Wires are expensive, especially long ones
  - Wires consume power
  - Someone has to install them, and fix them if they are defective
  - How do you upgrade an already-wired data center?
  - How do you actually utilize the multiple paths?

Progress on wiring topologies

- Run the wires on backplanes:
  - Data Center Switch Architecture in the Age of Merchant Silicon (HotI '09)
  - although this is sort of back to building large switches
- Fully-loaded cost comparison of various alternatives
  - A Cost Comparison of Data Center Network Architectures (CoNEXT '09)
  - Lots more work to do here
- Automatic optimization of wiring topologies
  - Taming the Flying Cable Monster (USENIX ATC '11)
  - Multiple NP-hard optimization problems … more work here, too.
- Optimizing the choice among multiple paths
  - SPAIN (NSDI '10): finding the best path for a flow
  - MPTCP (SIGCOMM '11): using multiple paths for one TCP connection

Interesting kinds of wires

- Copper wire is cheap, but:
  - Takes lots of power, cannot carry 10Gb more than 10—15M
  - Ethernet is ubiquitous … but InfiniBand still has better Gbps/$
- All-optical "wires"?
  - Assessment of Optical Switching in Data Center Networks (NFOEC '10)
    - lots more work needed before this is practical
  - MEMS: OK for circuit switching, too slow for packet switching
  - Hybrids might work: c-Through or Helios (SIGCOMM '10)
- Wireless data center networks?
  - Augmenting DC Nets with Multi-Gigabit Wireless Links (SIGCOMM '11)
  - use wireless to bypass sparse hotspots; get away with a cheaper core

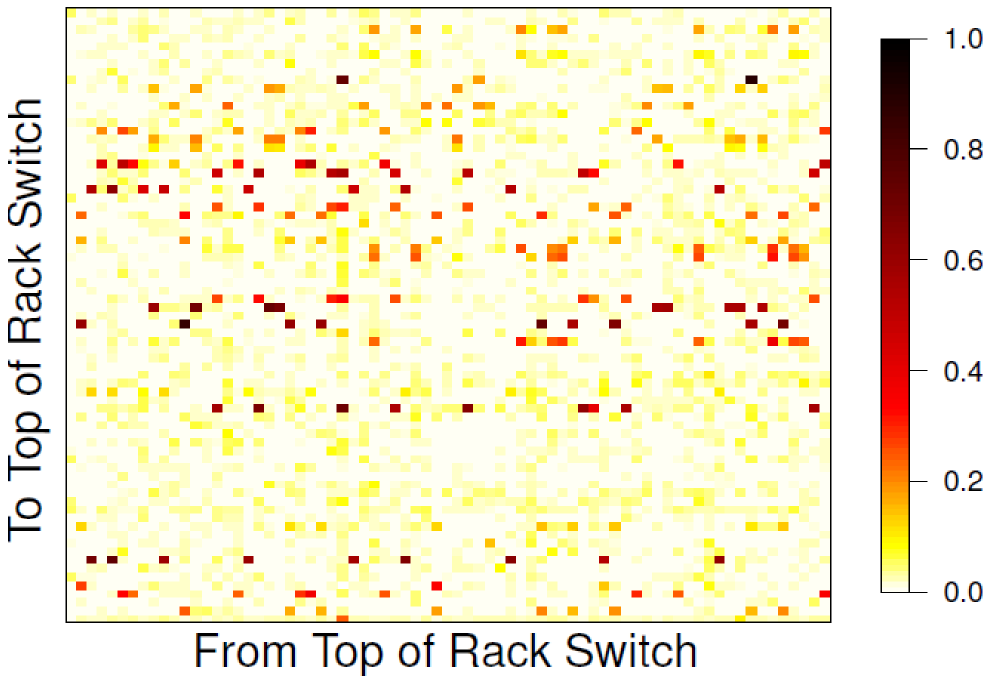


Figure 11: Traffic Demands (normalized) between ToR Switches.

Reprinted from "[Augmenting Data Center Nets with Multi-Gigabit Wireless Links]", Daniel Halperin, Srikanth Kandula, Jitendra Padhye, Paramvir Bahl, and David Wetherall, *Proc. SIGCOMM 2011*

Do we really want TCP on the bottom?

Historically, flow control and congestion avoidance belonged in the transport layer:

- Reflects the "end-to-end" & "narrow waist" arguments
- Keeps the MAC layer and switches simple

Data Center Bridging (DCB, aka DCE aka CEE)

– Puts flow control (PFC) + cong. control (QCN) in L2 HW
– "Required" by storage convergence, especially FCoE
– Is this really a good idea? or are we better off with iSCSI?
– DCTCP (SIGCOMM '10) solves similar problems, but in L4

- What do applications actually need?
- Scaling to 100K nodes or more
- Isolation in multi-tenant/cloud data-centers
- What's the right physical layer?
- **Reducing network-related energy**
- Simplifying network management
- Fault detection and diagnosis
- Challenges of future applications

Reducing energy consumption for data-center networking?

How important is this?

- 5--15% is enough to worry about
- and it hasn't been at all "proportional"

Approaches:

- Reducing worst-case power:
  - Use internal optics; Avoid using TCAMs
- Improving proportionality:
  - Idle-traffic sleeping: not a good idea for latency-sensitive apps?
  - Elastic Tree (NSDI '10): re-route traffic, so switches can sleep
  - Networks of Tiny Switches (NoTS): lots of really low-power switches

- What do applications actually need?
- Scaling to 100K nodes or more
- Isolation in multi-tenant/cloud data-centers
- What's the right physical layer?
- Reducing network-related energy
- **Simplifying network management**
- Fault detection and diagnosis
- Challenges of future applications

How do you manage a 100K-node network?

Stuff to worry about:

- Failures and repairs, routing, ACLs, DDoS, placing and moving VMs, billing tenants; services like DNS
- Change control; avoiding misconfiguration
- Keep it cheap: today, 1 admin per ~20 switches
  - Probably, the cost is worse than linear in the number of switches
  - This is for managing changes; managing failures is easier

We need to manage the network, not the pieces

- OpenFlow helps … a little
- An un-sexy area for research, but we need it!

What about traffic flow scheduling?

Flow scheduling: explicitly choose where flows go, because:

- You can't afford full bisection bandwidth everywhere
- or, Your "full bandwidth" topology needs perfect routing
- and, You would rather not constrain VM/sever placement due to network bandwidth limits
- and, random (ECMP) is suboptimal because of elephants

Recent automated approaches based on SDNs:

- Hedera (NSDI '10): detect elephants at switches & re-route based on a predictive model
- Mahout (INFOCOM '11): detect elephants sooner, at end-hosts

- What do applications actually need?
- Scaling to 100K nodes or more
- Isolation in multi-tenant/cloud data-centers
- What's the right physical layer?
- Reducing network-related energy
- Simplifying network management
- Fault detection and diagnosis
- Challenges of future applications

Things will go wrong

Accurate and detailed failure data is useful for:

- Responding to failures in real time
- Detecting failure trends
- Designing fault-tolerant networks and components

But such data for DC networks has been hard to obtain

- Data-center operators don't often share
- Logs usually need a lot of cleanup heuristics
- "Faults" are one thing; failures with impact are another
- VL2: faults high in hierarchy are rare but high-impact
- Understanding Network Failures in DCs (SIGCOMM '11)

- What do applications actually need?
- Scaling to 100K nodes or more
- Isolation in multi-tenant/cloud data-centers
- What's the right physical layer?
- Reducing network-related energy
- Simplifying network management
- Fault detection and diagnosis
- Challenges of future applications

The next set of challenges

- Super-low latency
  - RAMClouds (OSR 12/09): all data in DRAM, 5-10 μsec RPC latency
- Security, privacy, and anti-malware – all at the same time
- Idiot-proof network management
- Disaggregating + recomposing computing and storage
  - Blur distinction between “CPU-memory interconnect” and “network”
- Packet-switching meets all-optical networks
- Applications with VMs in multiple data centers
- Useful benchmarks/synthetic workloads for testing
  - Necessary for shared progress in quantitative field; don’t hide bogus tradeoffs
  - Per-packet flows; OpenFlow controller loads; multicast loads; etc.

A few words about HP Labs

- HP has O(320,000) employees, worldwide
- HP Labs has O(450) researchers, worldwide
  - Eight broad themes: Cloud and Security, Information Analytics, Intelligent Infrastructure, Mobile and Immersive Experience, Networking and Communications, Printing and Content Delivery, Services and Sustainability.
- The Networking and Communications Lab (NCL)
  - 30 researchers in Palo Alto, Beijing, and Princeton
  - Major projects on Enterprise/Data-Center networks, Programmable Networks, Large-Scale Sensor Networks
  - We plan to hire both summer interns and full-time researchers in 2012

Questions?

THANK YOU

Backup slides