



# PLANETLAB

An open, shared platform for developing,  
deploying, and accessing planetary-scale services

Timothy Roscoe

# Table of Contents

- What is PlanetLab?
- Architectural principles
- Current status
- What has this to do with Programmable networks?
- PlanetLab as a disruptive technology
- Opportunities / questions

# PlanetLab is...

- Large collection of machines spread around the world for distributed systems research
- Focus/catalyst for networking and systems community
- Intel project  $\Rightarrow$  consortium of companies and universities

# The value proposition

- Institutions provide ~3 nodes
  - IA32 architecture servers
  - Hosted outside the firewall
  - Provide power, cooling, & bandwidth
- In exchange, researchers get to use a small slice of machines worldwide.

# What PlanetLab is about

- Open platform for inventing next generation of “planetary scale” services.
- The foundation for the next Internet
  - Beyond TCP/UDP/IP/DNS/BGP/OSPF...
  - “the next internet will be created as an overlay on the current one”
- *More a platform than a testbed*
  - not a collection of pipes and giga-pops
  - not a distributed supercomputer
  - geographically distributed network services
- Focus and mobilize research community to define the emerging Internet

# Focussing the community

- Think of PlanetLab as a **shared artefact**
- Provides many diverse, overlapping projects around the world with a **stable place to stand** to change things
- Forum for **exchange** and **composition** of services and applications
- **Selection environment** based on real deployment and use
- **Bottom-up approach** to changing the world



PLANETLAB

Timothy Roscoe, 16 December 2003

intel®

Intel **Research**  
Berkeley

# Long-term aims

- PlanetLab incubates the next Internet
- New networks deployed as overlays over existing Internet
- *Service-oriented* network becomes the norm
- Computation as a localizable network resource

# Lots of work being done in widely distributed systems...

- Researchers had no vehicle to try out their next *n* great ideas in this space
  - Lots of architectures
  - Lots of simulations
  - Lots of emulation on large clusters
  - Lots of folks calling their 17 friends before the next deadline
- but *not* the surprises and frustrations of experience at scale to drive innovation

# Origins and progress

- “Underground” meeting March 2002
- Intel seeds
  - First 100 nodes
  - Operational support
- First node up July 2002
- By SOSP (deadline March 2003) 25% of accepted papers refer to PlanetLab
- Large presence at SIGCOMM
- 11 out of 27 papers in NSDI 2004

# Architectural principles

- *Slices* as fundamental resource unit
- Distributed Resource Control
- Application-Centric Interfaces
- Unbundled Management
- Self-obsolescence
  - everything initially built should eventually be replaced **by the community**
  - initial centralized services only bootstrap distributed ones

# Slice-ability

- Each *service* runs in a *slice* of PlanetLab
  - distributed set of resources (network of VM)
  - allows services to run continuously
- VM monitor on each node enforces slices
  - limits fraction of node resources consumed
  - limits portion of name spaces consumed
- Challenges
  - global resource discovery
  - allocation and management
  - enforcing virtualization
  - security

# VM options for slices

- VMs for complete desktop environment
  - e.g., VMware
  - extremely complete, poor scaling
- VM sandboxes widely used for web hosting
  - Ensim, BSD Jail, Linux VServers, UML,
  - limited `/bin`, no `/dev`, many VMs per  $\Phi$ M
  - *limit the API for security*
- Scalable Isolation kernels (VMMs)
  - host multiple OS's on cleaner VM
  - Xen, Denali,
  - Simple enough to make secure

# Distributed Resource Allocation Challenges

- Balance site and global policies
- Support multiple mechanisms
  - *Simultaneously*
  - SHARP, Dslice, Emulab, etc.

⇒ Provide flexible per-node, low-level multiplexer/reservation system



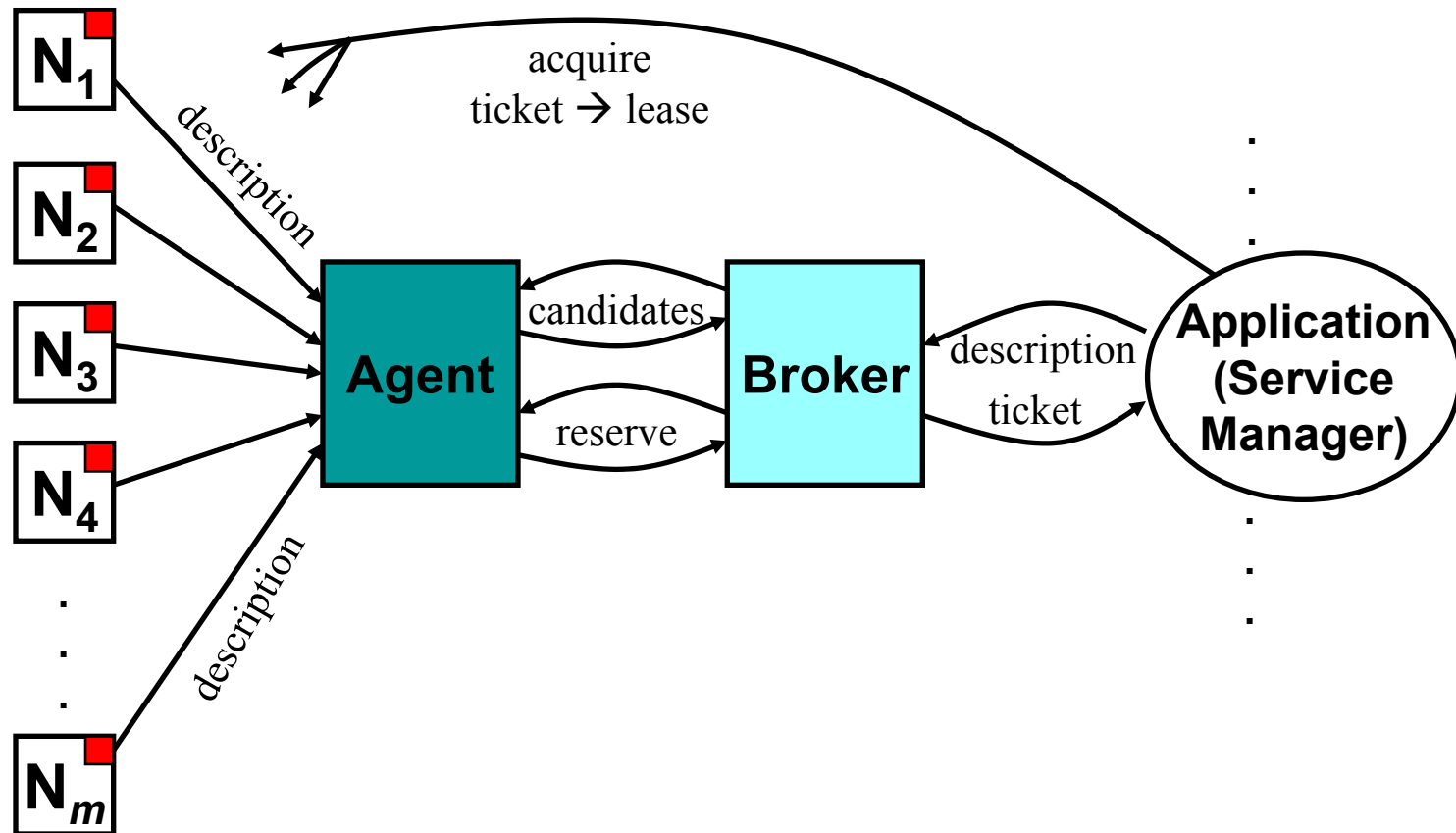
PLANETLAB

Timothy Roscoe, 16 December 2003

intel®

Intel **Research**  
Berkeley

# Dynamic slice creation



# Application-Centric Interfaces

- Inherent problems
  - stable platform versus research into platforms
  - writing applications for temporary testbeds
  - integrating testbeds with desktop machines
- Approach
  - take popular API (Linux), evolve implementation
  - later separate *isolation & application* interfaces
  - provide generic “shim” library for desktops

# Unbundled Management

- Partition mgmt into orthogonal services
  - resource discovery
  - monitoring system health
  - topology management
  - manage user accounts and credentials
  - software distribution and updates
- Approach
  - management services run in their own slice
  - allow competing alternatives
  - engineer for innovation (minimal interfaces)

# PlanetLab today



- About 340 nodes, 125 sites, 30 countries, 5 continents
- Universities, Labs, POPs, CoLos, DSL lines
- Large presence in systems research conferences



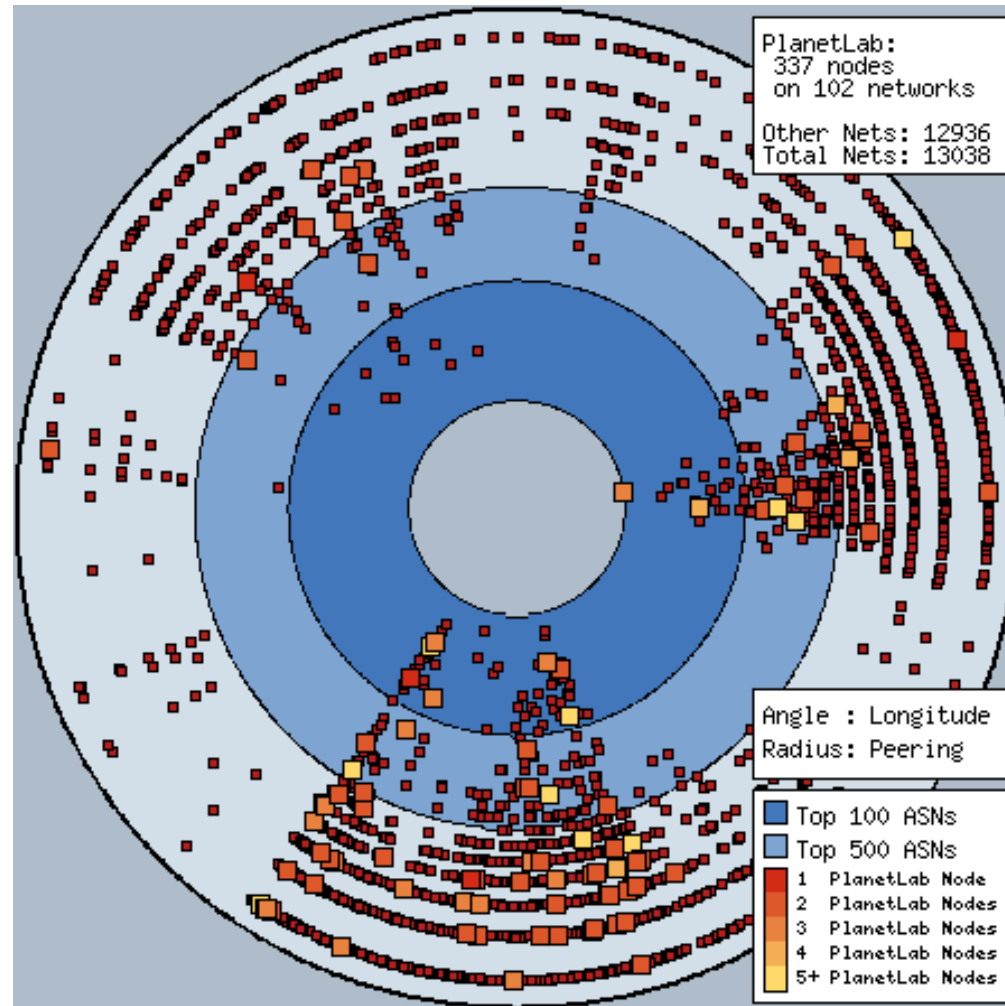
PLANETLAB

intel®

Timothy Roscoe, 16 December 2003

Intel **Research**  
Berkeley

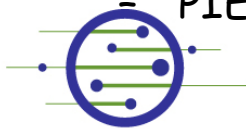
# Network coverage



# What do people use it for?

## (the ones we know about)

- Overlay Networks
  - RON, ROM++, ESM, XBone, ABone, etc.
- Network measurement
  - Scriptroute, \*Probe, I3, etc.
- Application-level multicast
  - ESM, Scribe, TACT, etc.
- Wide-area distributed storage
  - Oceanstore, SFS, CFS, Palimpsest, IBP
- Resource allocation
  - Sharp, Slices, XenoCorp, Automated contracts
- Distributed query processing
  - PIER, IrisLog, Sophia, etc.
- Content Dist. Networks
  - CoDeeN, ESM, UltraPeer emulation, Gnutella mapping
- Management and Monitoring
  - Ganglia, InfoSpect, Scout Monitor, BGP Sensors, etc.
- Distributed Hash Tables
  - Chord, Tapestry, Pastry, Bamboo, etc.
- Virtualization and Isolation
  - Denali, VServers, SILK, Mgmt VMs, etc.
- Router Design implications
  - NetBind, Scout, NewArch, Icarus, etc.
- Testbed Federation
  - NetBed, RON, XenoServers
- Etc., etc., etc.



# Where is it going?

- Just beginning
  - towards representative sample of the Internet
- Working Groups
  - Virtualization, Dynamic slices, Monitoring, etc.
- Building the consortium
  - Industrial partners, testbeds, gov't funding, etc.
- Hands-on experience is mothering tremendous innovation
  - nothing "just works" in the wide-area at scale
- Rich set of research challenges ahead

# What has this to do with Programmable Networks?

- PlanetLab addresses the related problems of:
  - *Deploying* programmable networking architectures
  - *Evaluating* competing approaches in a realistic setting
  - *Evolving* the "real" network architecture towards a more programmable model
- PlanetLab seems to be quite successful at doing this.



PLANETLAB

Timothy Roscoe, 16 December 2003

intel®

Intel  
**Research**  
Berkeley

# Why is it successful?

- Community model
  - "network effects"
  - Lots of benefit from small entry fee
- Sliceability
  - Enables multiple approaches
  - Sharing of the platform
- Virtual machine interface
  - Emphasis on *multiplexing the machine*
  - *Isolation* left to the VMM



# Also...

- The central problem in building the PlanetLab platform (and many applications) is *distributed management*
- There would seem to be parallels with Programmable Networks

# What is PlanetLab doing to the Internet?

- PlanetLab appears to function as a disruptive technology
  - Applications use the network differently
  - The network sometimes reacts in a hostile manner
  - Leads to new requirements on infrastructure

# Operational experience

- The Internet is extraordinarily brittle
  - Innovation resembles Denial-of-Service
  - IDSes default to "attack warning"
  - "Common cases" burned into routers
- PlanetLab now supports *full packet auditing* to end-users

# The "end-to-end" argument...

- PlanetLab changes what constitutes an "end-system"
    - Slices share physical nodes
    - Slices are distributed
    - Responsibility needs to shift to slices
    - TCP is no longer end-to-end
- ⇒ Tension between applications and "Internet architecture"

# Operational experience

- UDP replaces TCP
  - N-to-N applications are different
  - Short-term: File Descriptor limits
  - Long-term: Remove abstraction barriers
    - Aggressive application timeouts
    - Detailed information about network perf.
  - Still congestion controlled
  - DCCP not available



# Questions

The opportunity exists to seriously change the network by deploying research artefacts.

- What can the ProgNet community offer PlanetLab to help build the next network?
- What can PlanetLab provide the ProgNet community as a deployment platform?

# Credits...

- Tom Anderson, Andy Bavier, Mic Bowman, Paul Brett, Brent Chun, Jon Crowcroft, David Culler, Frans Kaashoek, Scott Karlin, Aaron Klingaman, Vivek Pai, Larry Peterson, Amin Vahdat, Mike Wawrzoniak, and many, many researchers worldwide...

Thanks - please join us!

<http://www.planet-lab.org/>

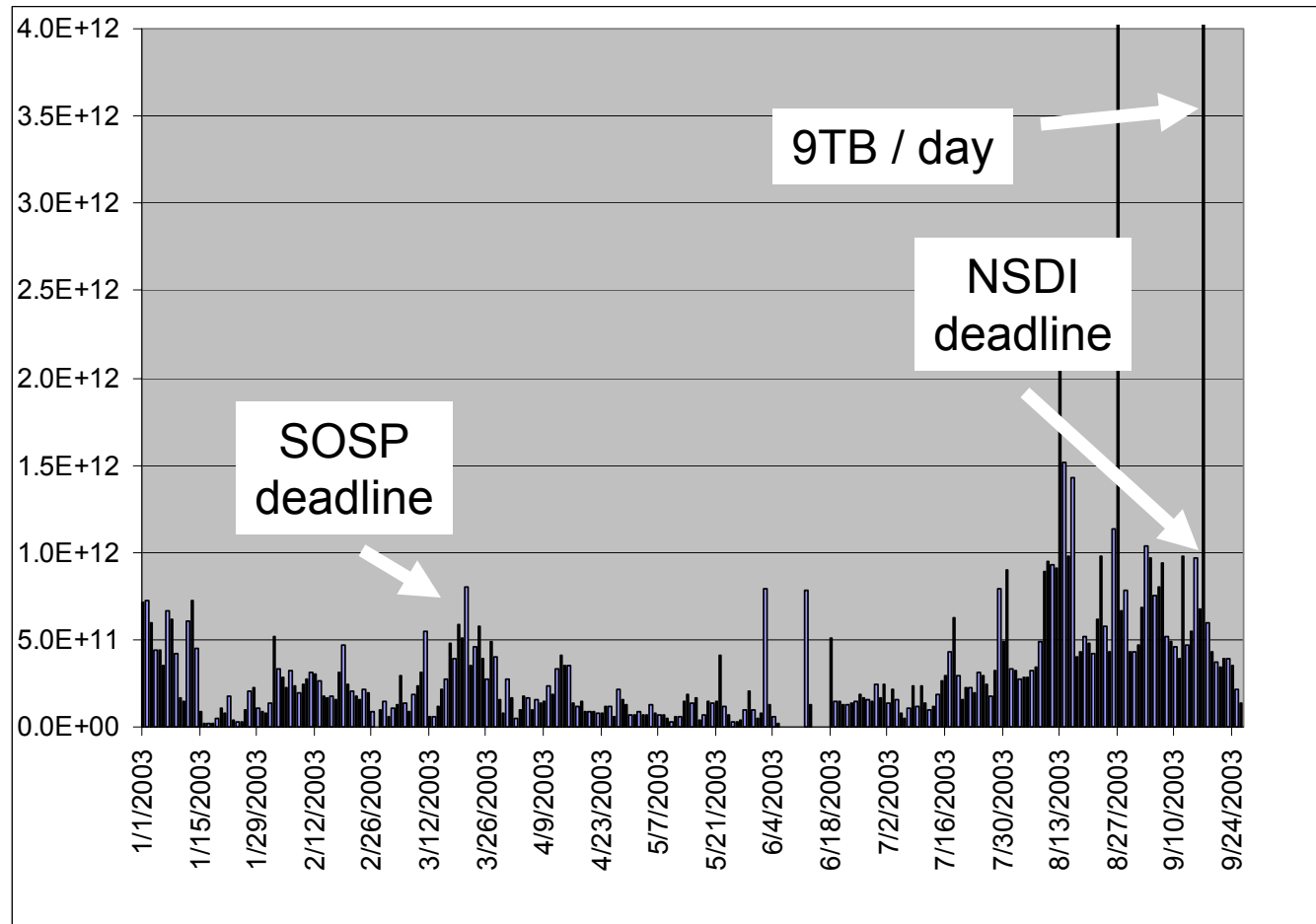
# Backup slides



Timothy Roscoe, 16 December 2003



# Aggregate bandwidth usage



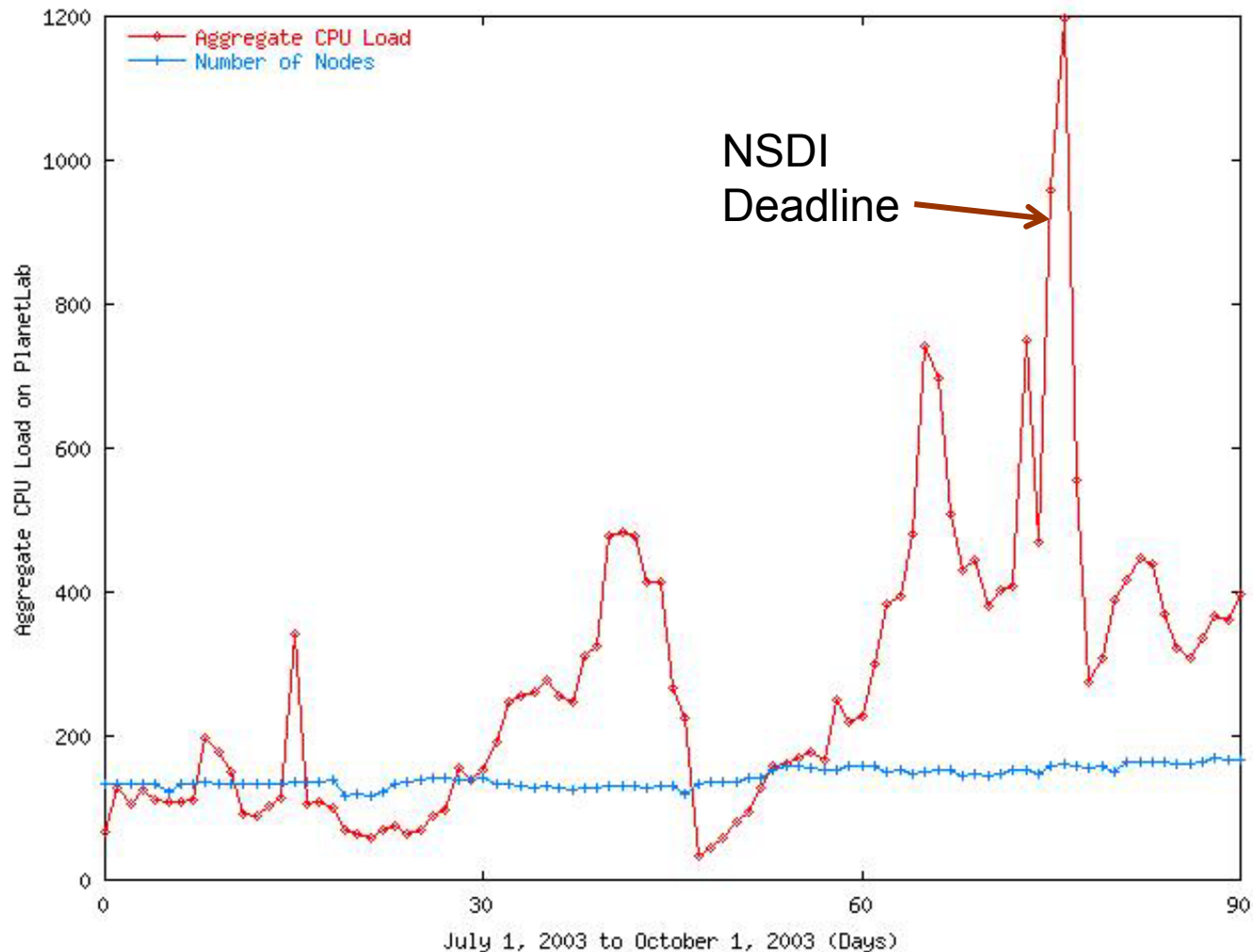
PLANETLAB

Timothy Roscoe, 16 December 2003

Intel Research  
Berkeley

intel®

# Aggregate CPU load



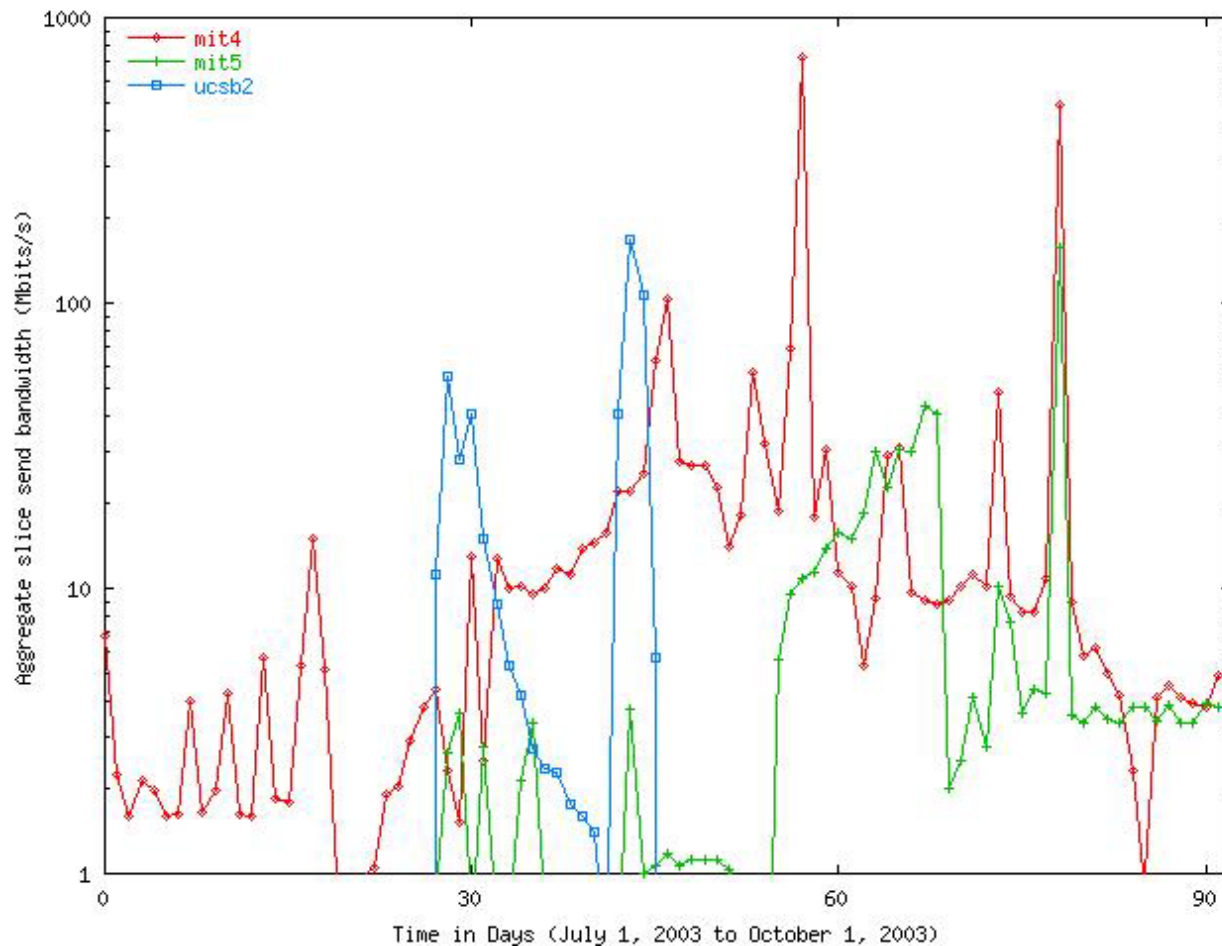
PLANETLAB

Timothy Roscoe, 16 December 2003

Intel Research  
Berkeley

intel®

# Top 3 sites (network b/w)



PLANETLAB

Timothy Roscoe, 16 December 2003

Intel Research  
Berkeley

intel®