



# PLANETLAB

Timothy Roscoe  
Intel Research at Berkeley  
Friday, May 13th, 2005

# Overview

- Intro: history, goals, concepts
- Basics: accounts, slices, nodes
- Real work: developing & deploying  
*< break >*
- Tools and services
- Advanced network programming
- Methodology issues
- Where to go for more information



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# What this won't cover...

- Emphasis is on using the PlanetLab PLC web interface
- This tutorial doesn't talk about:
  - How institutions can join PlanetLab
  - Most of a PI's duties
  - PlanetLab's detailed architecture
  - PlanetLab's programmatic API
  - Future directions



Timothy Roscoe, 8 May 2005



# Part I: History, goals, and concepts

# PlanetLab is...

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

# The value proposition

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

# Origins: wide-area distributed systems research c.2002

- Researchers had no way to try out real systems
  - Architectures, simulations, emulation on large clusters, calling 17 friends before the next deadline...
- but *not* the surprises and frustrations of experience at scale to drive innovation
- How can research systems be validated?

# Origins: large-scale networking research c.2002

- Strong feeling the Internet had ossified
  - Intellectually, infrastructure, etc.
  - NRC "looking over fence at networks"
- New ideas abandoned as undeployable
  - Overlays as a way out of the impasse
  - Next internet emerges as overlay (again)
- How can researchers deploy overlays?



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Early timeline

- David Culler and Larry Peterson discuss initial idea early 2002
- "Underground" meeting March 2002
- Position paper (Anderson, Culler, Peterson, Roscoe) June 2002.
- Intel seeds project, core team, 100 nodes
- First node up July 2002
- By SOSIP (deadline March 2003) ~25% of accepted papers refer to PlanetLab
- Large presence at SIGCOMM
- 11 out of 27 papers in NSDI 2004



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# PlanetLab is...

- A testbed for experiments
  - Experiment at scale
  - Real-world conditions
  - Potential for real workloads & users
- A deployment platform
  - Continuously running services
  - Design, deploy, measure cycle
  - Long-term studies
- A microcosm of the next Internet
  - Evolve network to support overlays and slices
  - Make the network architecture more *computational*
- A shared artifact!



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# What is PlanetLab good for?

- *Planetary-Scale* applications:
  - **Low latency** to widely spread users
  - **Span boundaries:** jurisdictional and administrative
  - **Simultaneous viewpoints:** on the network or sensors
  - **Hardware deployment** is undesirable
- Deploy, Evaluate, Evolve the architecture



PLANETLAB

Timothy Roscoe, 8 May 2005

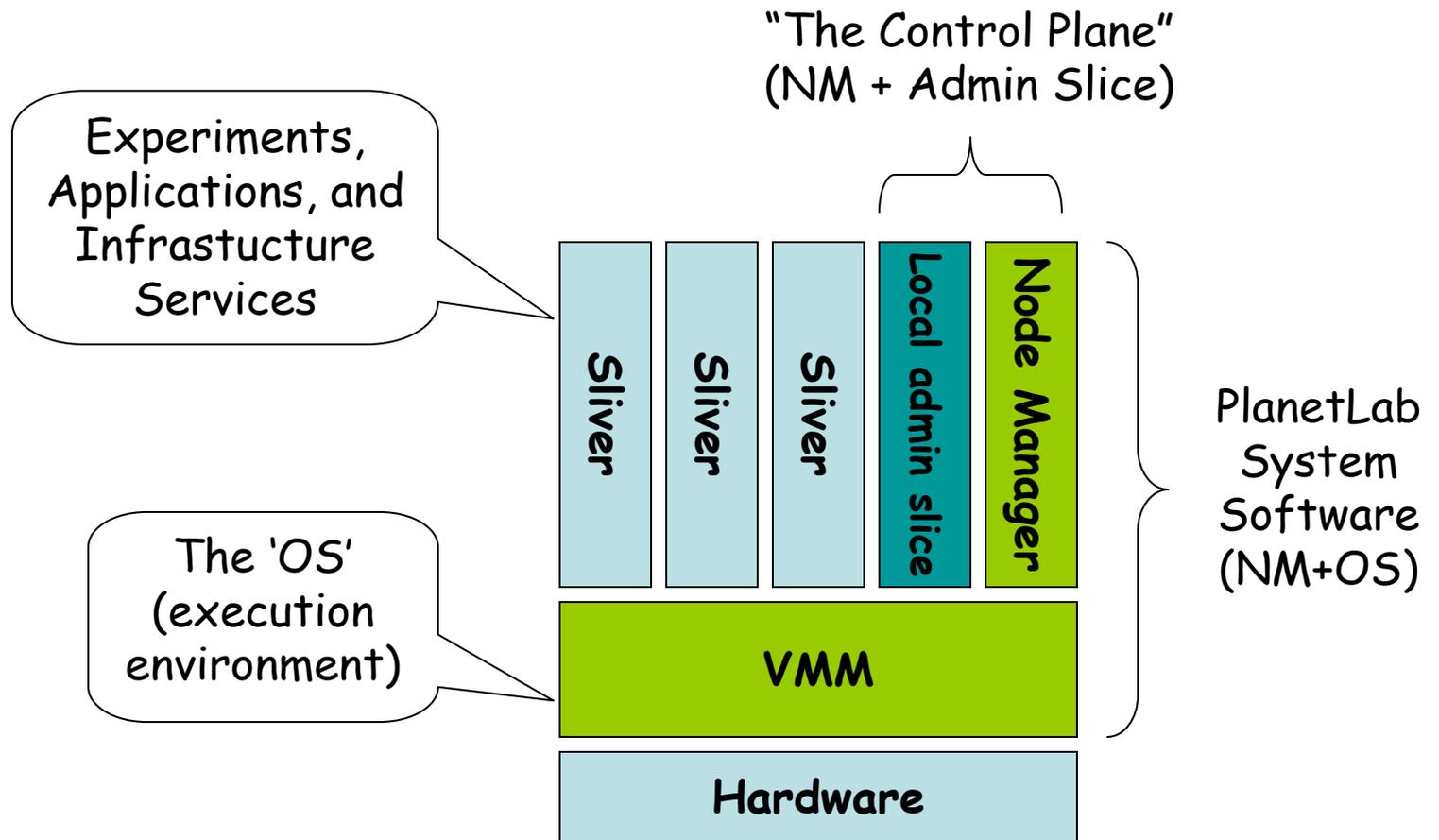
Intel  
Research  
Berkeley

intel®

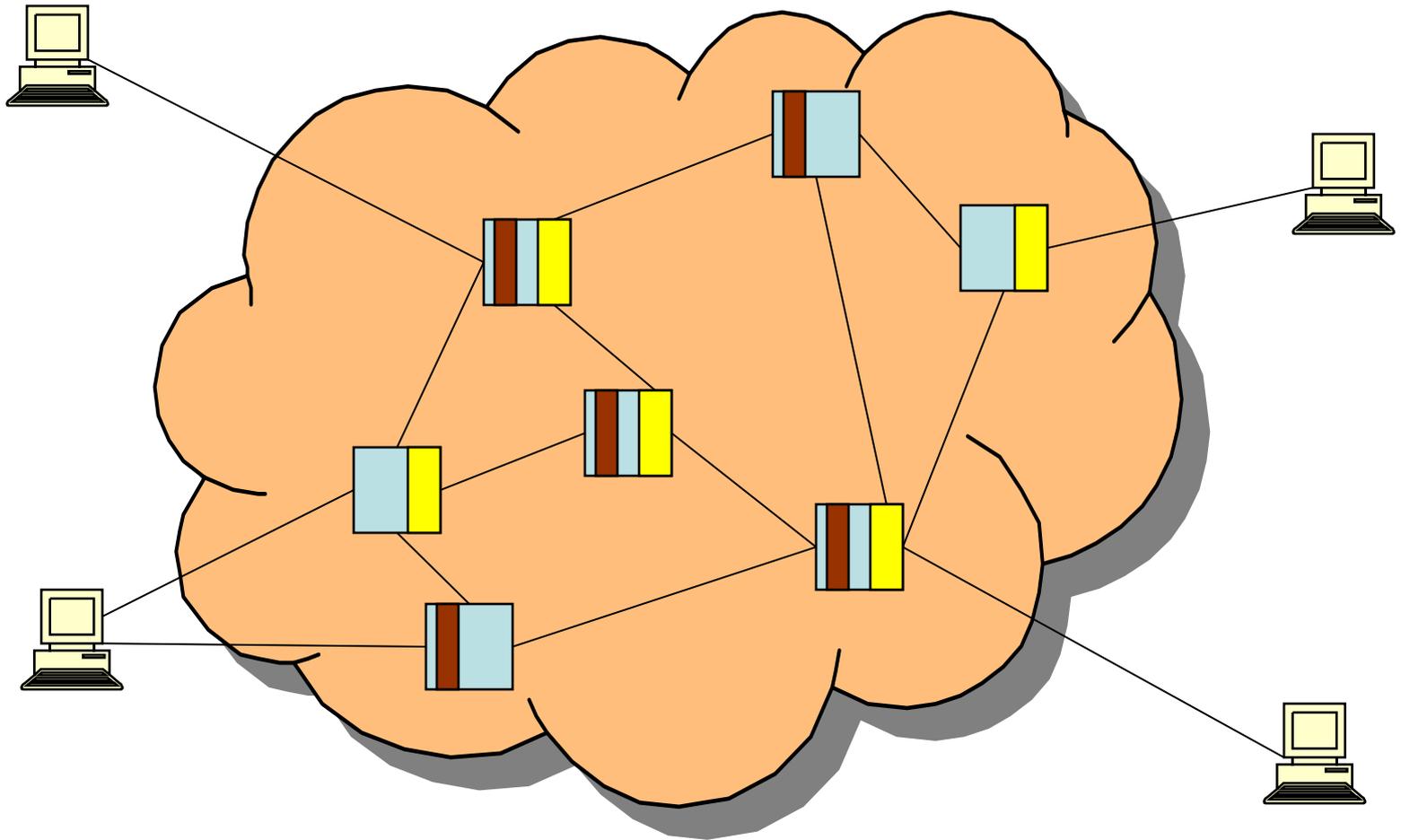
# Distributed Virtualization

- *Slices*
  - Basic unit of isolation
  - Distributed set of virtual machines (slivers)
  - Services & applications run "in" slices
- *Nodes*
  - Physical machines, grouped into *Sites*.
  - One node hosts many slivers
- **Infrastructure Services**
  - Provide functionality to developers or other services rather than users

# Node architecture



# Slices



# PlanetLab these days



About 550 nodes, 260 sites, 30 countries, 5 continents  
Universities, Labs, POPs, CoLos, DSL lines  
Huge presence in systems research conferences  
> 400 projects so far



Timothy Roscoe, 8 May 2005

Intel **Research**  
Berkeley

intel®

# What do people use it for?

## (a few we know about)

- Overlay Networks
  - RON++, Pluto, P2, etc.
- Network measurement
  - Scriptroute, \*Probe, etc.
- Application-level multicast
  - ESM, Scribe, TACT, etc.
- Wide-area storage
  - Oceanstore, SFS, SFS-RO, CFS, Ivy, Palimpsest, IBP
- Resource allocation
  - SHARP, Bellagio, Automated contracts
- Distributed query processing
  - PIER, SDIMS, Sophia, IrisLog, etc.
- Network architecture
  - Evolve, Detour, I3
- Content Dist. Networks
  - CoDeeN, ESM, UltraPeer emulation, Gnutella mapping
- Management and Monitoring
  - Ganglia, InfoSpect, Sword, BGP Sensors, etc.
- Distributed Hash Tables
  - Chord, Tapestry, Pastry, Bamboo, Kademia, etc.
- Virtualization and Isolation
  - Xen, VServers, SILK, Mgmt VMS, etc.
- Router Design implications
  - NetBind, Scout, Network capabilities, Icarus, etc.
- Testbed Federation
  - NetBed, RON, XenoServers
- Etc., etc., etc.



# The PlanetLab Consortium

- Modelled on the W3C
- Run by Universities
  - U. Washington, U.C. Berkeley, U. Cambridge, Princeton U.
  - Based in Princeton, NJ, USA.
- Funded by Industry and Govts.
  - NSF, EU, Cernet, etc.
  - Intel, HP, Google, AT&T, France Telecom...



Timothy Roscoe, 8 May 2005



# The PlanetLab Consortium

- Node resources provided by member institutions
- Small "support" team NOC in Princeton
  - Additional NOCs planned in Europe (Paris), China (Tsinghua)
- Steering Committee
  - University representatives
  - Top-level industrial sponsors



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Staff is small

- ~5 people in Princeton.
- Q. Who develops the software then?
- A. You do!
  - System software, tools, services, libraries, measurement data, etc. contributed by participating institutions (often researchers)
  - PlanetLab is as much a community as an artifact
- Irregular meetings debate technical direction
  - Recommendations in PlanetLab Design Notes (PDNs)
  - Announced on mailing lists and web site; all welcome
  - Modelled on early IETF

# Summary of Part I

- PlanetLab is a shared, communal infrastructure of machines ("nodes")
- Nodes are hosted by institutions and connected directly to the Internet
- Each node hosts many virtual machines ("slivers") on behalf of users
- Services, experiments, applications run in distributed collections of slivers called "slices".



Timothy Roscoe, 8 May 2005

Intel **Research**  
Berkeley

intel®

# Part II: Basics

## Accounts, slices, nodes

# First steps to using PlanetLab

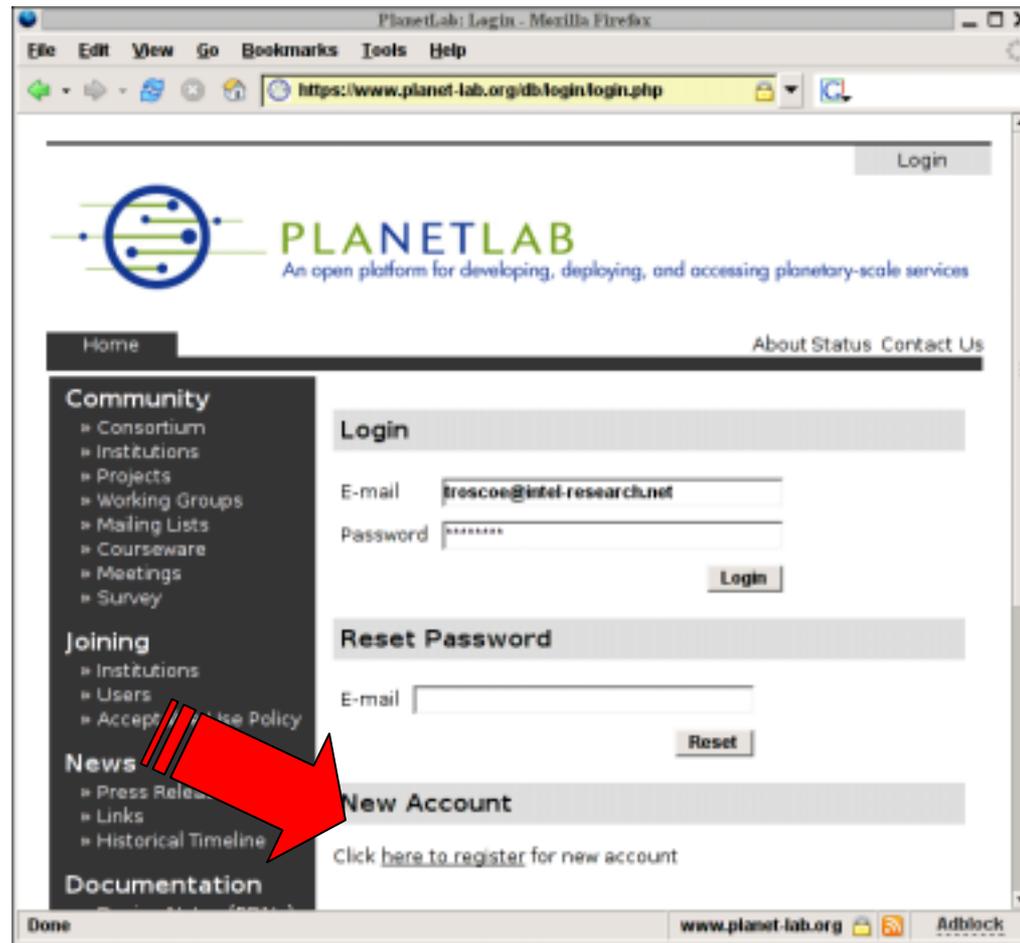
1. Register as a user
2. Create an ssh key
3. Create a slice
4. Add nodes to the slice
5. Describe the slice
6. Log into a sliver

# Getting started: accounts

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

The screenshot shows the PlanetLab website homepage. A red arrow points to the 'Login' button in the top right corner. The page features the PlanetLab logo, a navigation menu, a sidebar with links for Community, Joining, News, and Documentation, a world map showing node distribution, and an Announcements section.

# Step 1: Register



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel

# The Acceptable Use Policy

PlanetLab: Account Registration - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

https://www.planet-lab.org/ib/accounts/showaup.php

- No hacking attempts of the PlanetLab nodes. This includes "red team" (hacker test) experiments. All access is non-root.
- Avoid spin-wait for extended periods of time. If possible, do not spin-wait at all.

### Network Usage Rules

- Do not use your PlanetLab slice (account) to gain access to any hosting site resources that you did not already have.
- Do not use one or more PlanetLab nodes to flood a site with so much traffic as to interfere with its normal operation. Use congestion controlled flows for large transfers.
- Do not do systematic or random port or address block scans. Do not spoof or sniff traffic.

### Consequences

Violation of this AUP may result in any of the following:

- disabling the slice (account);
- removing the site from PlanetLab;
- informing the organization's administration.

To report a suspected violation of this policy, [click here](#).

I have read the Acceptable Use Policy, and will abide by its terms.

Copyright ©2004-2005 PlanetLab Consortium

Done www.planet-lab.org AdBlock

# What the AUP says...

(but read it carefully!)

- Research and educational use only
  - No commercial or illegal activities
- No suspicious network traffic
  - No portscans, DoS attacks, spoofing, repeated probing of routers, etc.
- Share resources responsibly
  - No tight loops, use congestion control, etc.
- Be nice and sensible!



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# What the AUP says (contd)

- Assume no privacy
  - Assume other users, and site sysadmins, can see all your traffic.
- Assume no reliability
  - Nodes may reboot at any time, without warning. They may not come back.
- Assume no durability
  - Disks may be wiped at any time, without warning.
- In practice, this rarely happens.  
But it *does* happen.



# Filling out the form

- Fill in contact details
- Select your site
  - E.g. "Universidade Federal de Minas Gerais"
- Are you a PI?
  - PIs are responsible for PlanetLab site participation, approve users, create slices
  - Most users are not PIs
- Are you a tech contact?
  - Probably not!
- Submit the form...



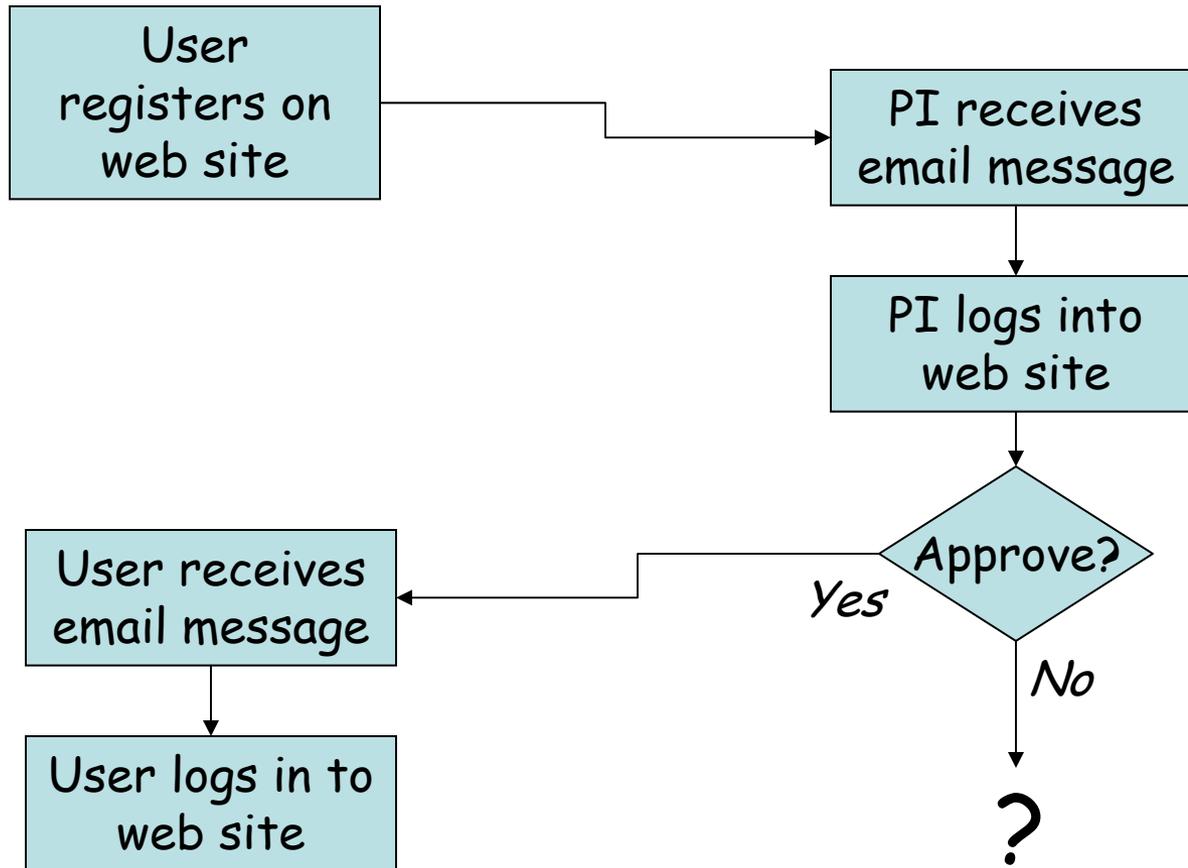
PLANETLAB

Timothy Roscoe, 8 May 2005

Intel **Research**  
Berkeley

intel®

# User registration



# Account approval (as PI)



PlanetLab: View My Users - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

https://www.planet-lab.org/db/accounts/

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

Home User **PI** Admin About Status Contact Us

**Sites**

- » View All Sites
- » View My Site
- » Update My Site
- » Switch Site

**Nodes**

- » View All Nodes
- » View Site Nodes
- » Other Node Groups
- » Control Site Nodes
- » Bandwidth Limits
- » Power Control Units

**Users**

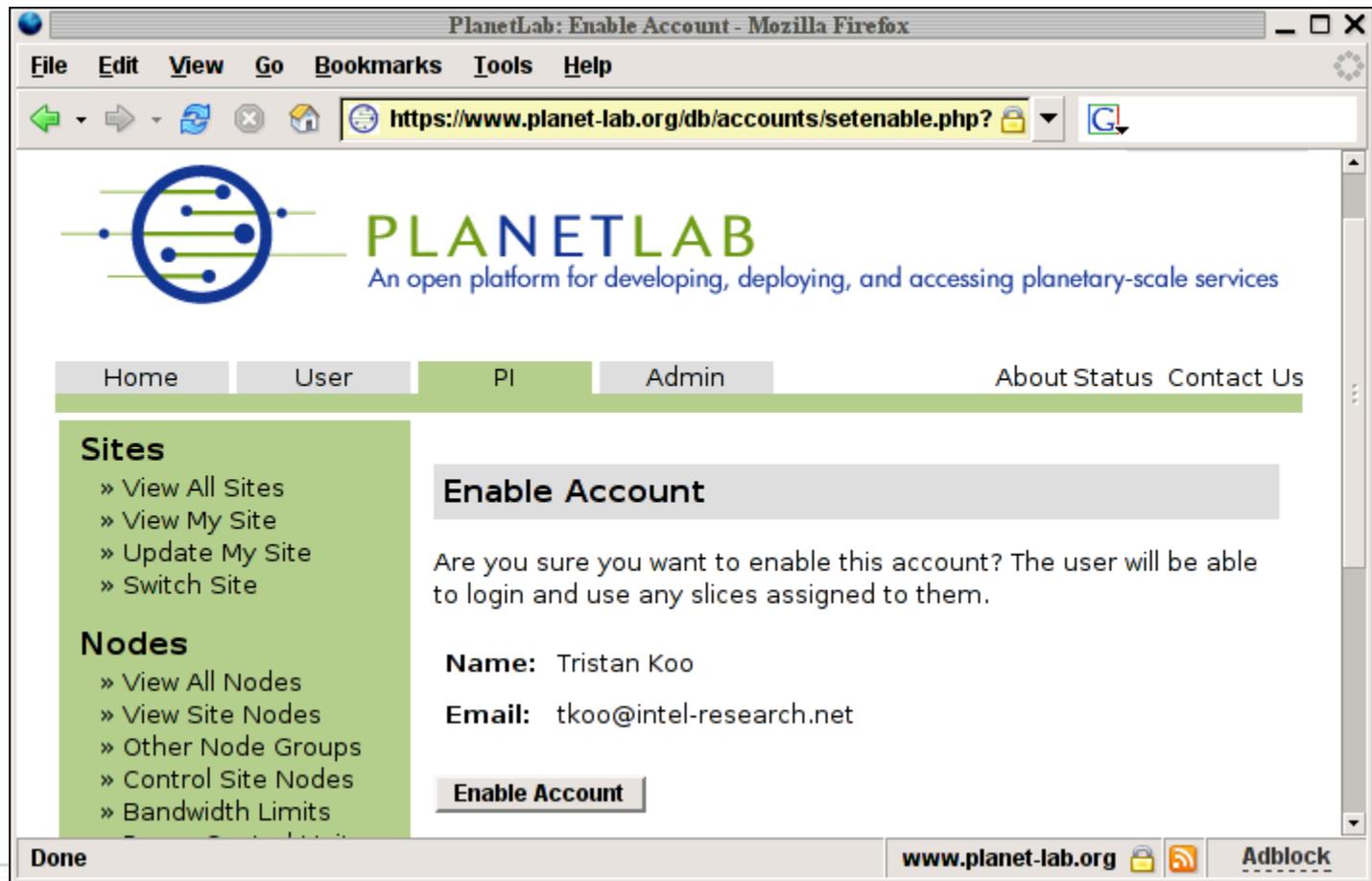
- » Manage Users
- » Manage My Account
- » Manage My Keys

**Accounts at Intel Research Berkeley**

<a href="#">Buonadonna, Philip</a>	buonadonna@acm.org	user
<a href="#">Chang, James</a>	jameschn@uclink.berkeley.edu	user
<a href="#">Chun (AdminAPI Account), Brent</a>	devnull@intel-research.net	user
<a href="#">Chun (Non-admin Account), Brent</a>	bnc@theether.org	user
<a href="#">Chun, Brent</a>	bnc@intel-research.net	admin user tech
<a href="#">Fall, Kevin</a>	kfall@cs.berkeley.edu	user
<a href="#">Fulton, Bryan</a>	bfulton@eecs.berkeley.edu	user

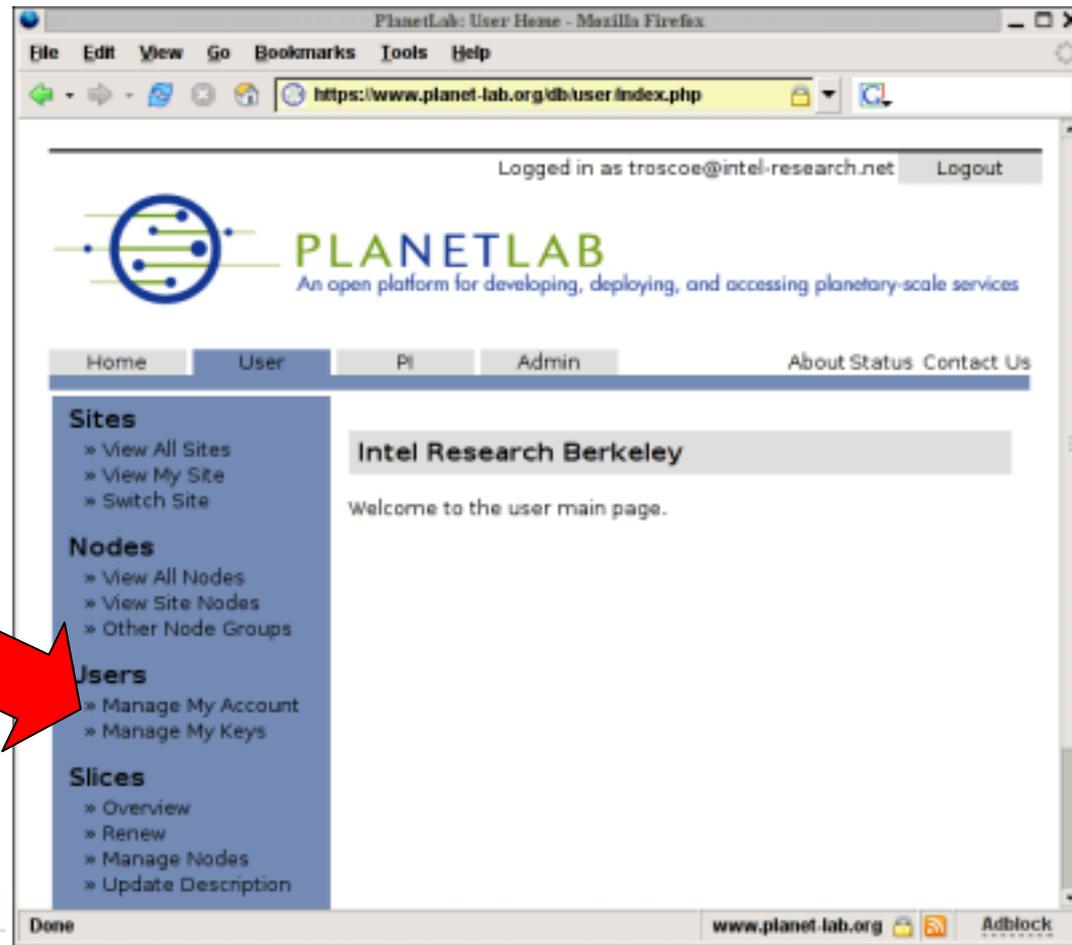
Done www.planet-lab.org Adblock

# Account approval (as PI)



The screenshot shows a Mozilla Firefox browser window with the title "PlanetLab: Enable Account - Mozilla Firefox". The address bar contains the URL "https://www.planet-lab.org/db/accounts/setenable.php?". The page header features the PlanetLab logo and the text "PLANETLAB An open platform for developing, deploying, and accessing planetary-scale services". A navigation menu includes "Home", "User", "PI", "Admin", "About Status", and "Contact Us". The "PI" tab is selected. On the left, there are two sections: "Sites" with links for "View All Sites", "View My Site", "Update My Site", and "Switch Site"; and "Nodes" with links for "View All Nodes", "View Site Nodes", "Other Node Groups", "Control Site Nodes", and "Bandwidth Limits". The main content area is titled "Enable Account" and contains the text: "Are you sure you want to enable this account? The user will be able to login and use any slices assigned to them." Below this, the user's details are listed: "Name: Tristan Koo" and "Email: tkoo@intel-research.net". At the bottom of the content area is an "Enable Account" button. The browser's status bar at the bottom shows "Done", the website URL "www.planet-lab.org", and an "Adblock" extension icon.

# You are now a user!



# Step 2: Generating an SSH key

- PlanetLab uses 1024-bit RSA keys for authentication on nodes
- Upload your key to the website
- Using OpenSSH:
  - `ssh-keygen -q -f /.ssh/id_planetlab -t rsa`
  - *Do use a secure passphrase*  
See later for using `ssh-agent` to reduce typing.
  - Upload `id_planetlab.pub` to web site

# Using PuTTY on Windows

- Use PuTTYgen to generate an ssh v2 1024-bit RSA key
- Copy text box to web form
  - Don't use more than 1-word comments
  - Should look a bit like this (except breaks):
  - `ssh-rsa  
AAAAB3NzaC1yc2EAAAABIwAAAIEArybtveURZwKjYY0j4ma9QQW  
cKD3iiSrVFhXrFs4YdSVRgnd6siYdGAJhLgsCentlQOOFhjqtS  
9AY/eIVx99ibmEDV7RrP6mk5NsPQseH/315oYbEkOmhrTuMZ2js  
WqiTzODQ+RCNF3iMxM+fPGmxVLRtDr2puIsXdlJqxTJYt8=  
troscoe@deleuze`
- Can also import OpenSSH key pairs

# Step 3: Creating a slice

- Actually, ask your PI to create a slice..

Slice names are of form:  
<site>\_<local name>

# Assign users to a slice (as PI)

PlanetLab: Dynamic Slice Overview - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

https://www.planet-lab.org/db/slices/

**PLANETLAB**  
An open platform for developing, deploying, and accessing planetary-scale

Home User **PI** Admin AboutStatus Contact Us

**Sites**

- » View All Sites
- » View My Site
- » Update My Site
- » Switch Site

**Nodes**

- » View All Nodes
- » View Site Nodes
- » Other Node Groups

**Dynamic Slice Overview**

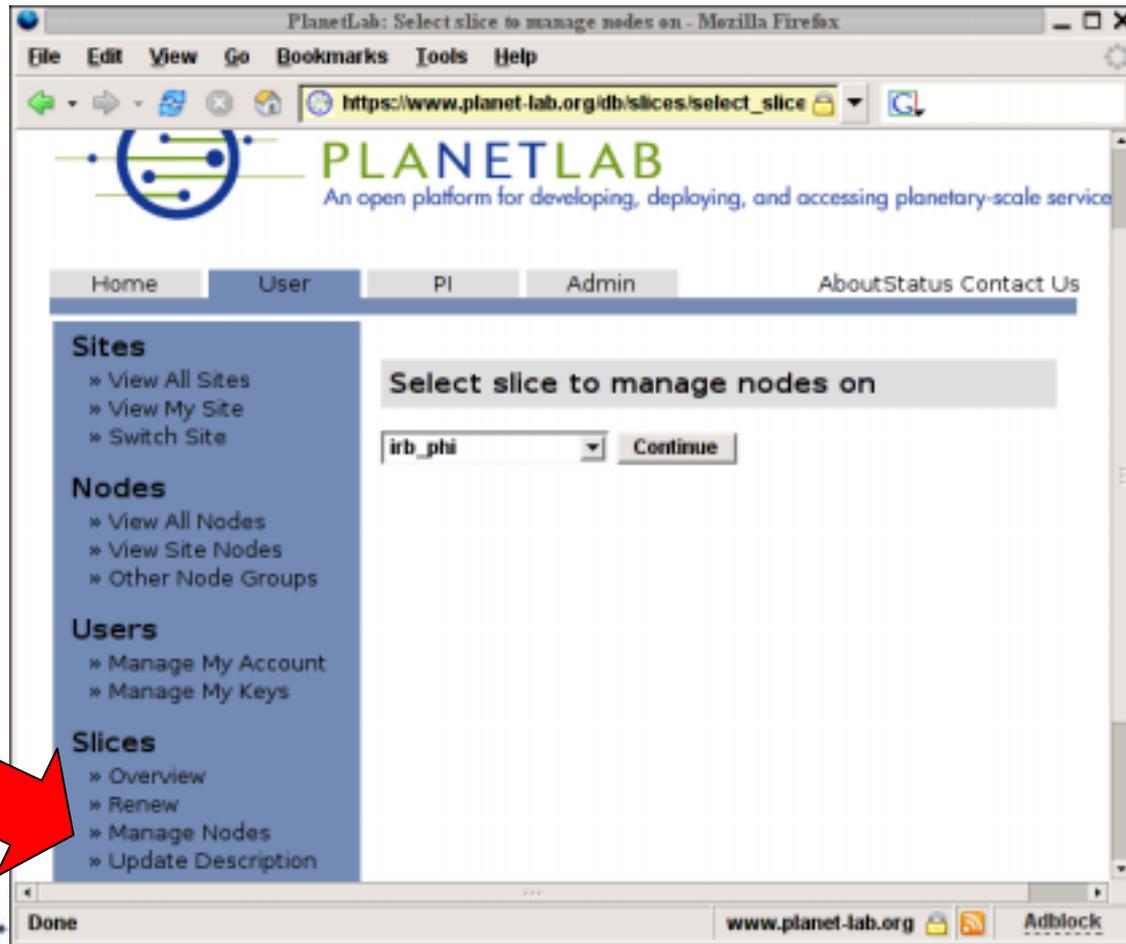
Slice Name/ State	Users	Expires
<a href="#">irb_tutorial</a> plc-instantiated	No users assigned	Friday 01-Jul-05 03:12:17 GMT

Done www.planet-lab.org Adblock

# By now...

- You have a user account
- Your PI has approved it
- Your PI has created a slice
- Your PI has assigned you as a user of the slice
- Next step: add nodes to the slice

# Step 4: Adding nodes



# Adding nodes

PlanetLab: Manage Node Assignments - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

https://www.planet-lab.org/db/slices/assign\_nodes

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

Home User PI Admin About Status Contact Us

**Sites**

- » View All Sites
- » View My Site
- » Switch Site

**Nodes**

- » View All Nodes
- » View Site Nodes
- » Other Node Groups

**Users**

- » Manage My Account
- » Manage My Keys

**Slices**

- » Overview
- » Renew
- » Manage Nodes
- » Update Description

[Back to slice details.](#)

**Manage Node Assignments**

Show:  
RNP - Ceara

Either all nodes from this site are already assigned, or no assignable nodes are available.

**All nodes currently assigned to slice irb\_phi:**

- 140.136.206.232
- 200-102-209-151.paemt7001.t.brasiltelecom.net.br
- 200-102-209-152.paemt7001.t.brasiltelecom.net.br
- alice.cs.princeton.edu
- arari.snu.ac.kr
- blast.uwaterloo.ca

Done www.planet-lab.org Adblock

# Adding nodes

- Page for adding/removing nodes
  - Select site to see nodes to add
  - Shown are all nodes currently in slice
  - Nodes can be removed
- For your first time, I suggest:
  - Add a couple of nodes, one nearby
- Note: Changes will take ~5 minutes to propagate!



PLANETLAB

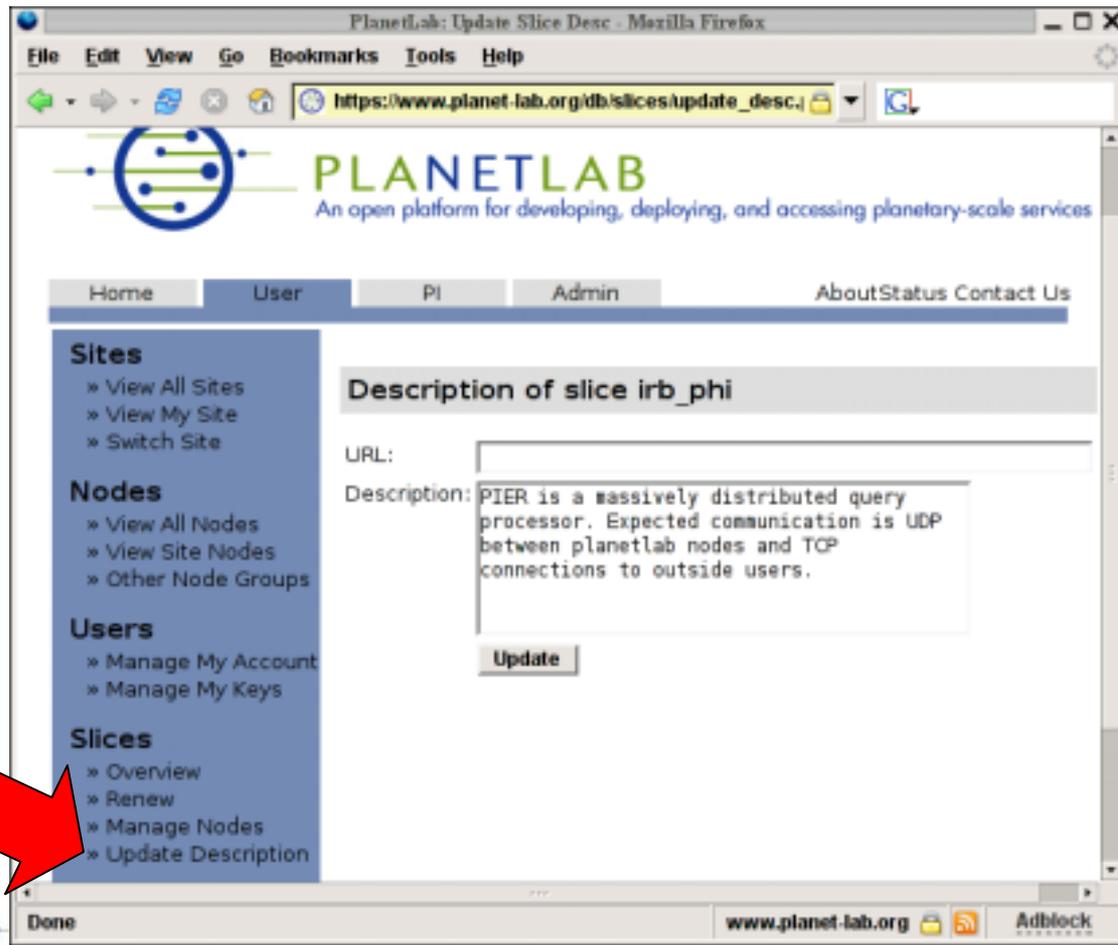
Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Step 5: Describe your slice

(while you're waiting for updates...)



The screenshot shows a Mozilla Firefox browser window with the URL [https://www.planet-lab.org/db/slices/update\\_desc.js](https://www.planet-lab.org/db/slices/update_desc.js). The page features the PlanetLab logo and navigation tabs for Home, User, PI, Admin, About, Status, and Contact Us. A left sidebar contains menu items for Sites, Nodes, Users, and Slices. The main content area is titled 'Description of slice irb\_phi' and includes a 'URL:' field, a 'Description:' text area containing the text 'PIER is a massively distributed query processor. Expected communication is UDP between planetlab nodes and TCP connections to outside users.', and an 'Update' button. A red arrow points from the 'Update Description' link in the sidebar to the 'Update' button.

# Step 6: Log in to a node (finally!)

- Your user name is your slice name
- E.g., with OpenSSH:
  - `ssh -l irb_phi \`  
    `-i ~/.ssh/id_planetlab \`  
    `planet1.berkeley.intel-research.net`

```
irb_phi@planet1:~  
$> ssh -l irb_phi -i .ssh/planetlab_rsa planet1.berkeley.intel-research.net  
The authenticity of host 'planet1.berkeley.intel-research.net (12.46.129.21)' ca  
n't be established.  
RSA key fingerprint is f4:b4:49:c0:fe:34:0c:f9:92:46:1c:2c:1b:ab:dc:19.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added 'planet1.berkeley.intel-research.net,12.46.129.21' (R  
SA) to the list of known hosts.  
Last login: Fri May 6 23:36:09 2005 from user-11fa28c.dsl.mindspring.com  
[irb_phi@planet1 irb_phi]$
```

# Logging in...

- Note that ssh asks you to accept the authenticity of the node
  - See later for how to avoid this
- If it works, congratulations!
- You should now have a Unix shell prompt
- Take a look around...

# What do you get on a node?

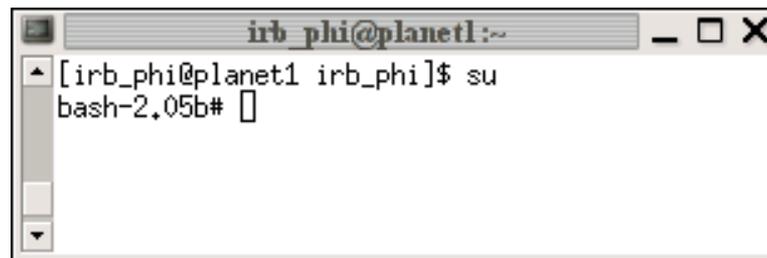
- Basic Linux Fedora Core 2 machine.
- Not many packages installed
  - Compilers, etc. missing
- Nobody logged in?

But look at the load average!



```
irb_phi@planet1:~  
[irb_phi@planet1 irb_phi]$ who  
[irb_phi@planet1 irb_phi]$ w  
19:52:41 up 44 days, 22:25, 0 users, load average: 15.06, 13.70, 12.27  
USER      TTY      FROM          LOGIN@  IDLE   JCPU   PCPU WHAT  
[irb_phi@planet1 irb_phi]$
```

- No root password?



```
irb_phi@planet1:~  
[irb_phi@planet1 irb_phi]$ su  
bash-2.05b#
```



# What's going on?

- You're in a *VServer*: a virtual Linux kernel
  - Each sliver is a VServer.
- Limited root capabilities, e.g.
  - Install software
  - Create new users
  - Open raw sockets
- Some resource sharing/scheduling



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Summary of Part II

- Registering, uploading keys, etc.
  - Done *once*.
- Creating a slice, assigning nodes
  - Typically done infrequently (once per project)
- Logging in
  - Environment is a *virtual Linux server*.

# Part III: Real Work Developing and Deploying

# Developing code: the challenge

- Writing code to run on a machine which is  $> 1000\text{km}$  away
- While libraries, etc. are needed on the remote machine?
- Copying files to remote machine...
- ...and for  $\sim 400$  other machines
- Keeping programs up-to-date on remote machines
- Debugging programs at a distance

# The good news:

- The environment is (mostly) Fedora Core 2 Linux
- Many tools have been developed to help with this problem
- Other users have provided *services* to help you
- Many researchers find this an interesting problem!



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Installing packages

- It's Fedora Core 2 so...
  - Yum, RPM, apt, etc. should work
  - As root - remember, you have root!
- But give Stork a try:
  - University of Arizona research project
  - Recently released (last week!)
  - Efficient shared package manager
  - Can also be used to install your own code!

# Stork



- For details, see
  - <http://www.cs.arizona.edu/stork/>
- Uses CoBlitz CDN for high-speed multicast of large binaries
- Shares packages between slices
- Can automatically keep packages up-to-date



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Developing code: "don't"s

- Don't compile on PlanetLab nodes
  - Requires you to install compilers, etc.
  - Debugging at a distance is harder
  - Takes valuable CPU from others!
- Don't run X11 clients on PlanetLab nodes
  - They're server machines
  - X11 doesn't work well over a WAN



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Developing code: "don't"s

- Don't treat a sliver as a workstation
  - Although it's Unix, it's intended for hosting long-running *services*, not for general use

# Developing code "do's"

- Do compile and test on your local desktop workstation
  - Using Fedora Core 2 is simplest
  - Can use other distros, but be careful with libraries
  - Static linking can often simplify things
  - Java usually portable, if you install the JRE on each node



# Developing code "do's"

- Do then copy binaries to the node to run
  - scp or rsync works for small numbers of nodes
  - Ensure library dependencies are satisfied

# OpenSSH hints and tips

- Authentication fails:
  - Try "ssh -v -2 ..."
- Make sure local permissions are correct:

```
chmod go-w $HOME $HOME/.ssh
chmod 600 $HOME/.ssh/authorized_keys
```
- If ssh hangs on exit:
  - redirect stdin/stdout/stderr to /dev/null
  - shopt -s huponexit in bash
- For more information, see:

<http://www.openssh.com/faq.html>



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Use authentication agents

- Use `ssh-agent` to avoid typing passphrase for your key
  - Note: "agent forwarding" doesn't work on nodes (yet)
  - For PuTTY users: Pageant
- Host key checking:
  - Set "strictHostchecking no" in `.ssh/config`
- Add all host keys from  
<http://www.planet-lab.org/xml/sites.xml>  
(see later!)

# Summary of Part III

- PlanetLab's execution environment is (today) Linux
- Develop on your local workstations, deploy on PlanetLab nodes
- For one (or few) nodes, simply a matter of copying
- BUT: how to deal with many (>400) nodes? See next section...



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Break!



Timothy Roscoe, 8 May 2005





# PLANETLAB

Timothy Roscoe  
Intel Research at Berkeley  
Friday, May 13th, 2005

# Part IV: Tools and Services

# Dealing with many nodes

- Deploying a large networked system requires:
  - *Discovery*: finding the nodes
  - *Deployment*: pushing the system out to the nodes
  - *Monitoring*: are the nodes up? Is the system up?
  - And much, much more.

# Management of Planetary-Scale Services

- Bad news:
  - This is a really hard problem
  - Perhaps the central problem in distributed systems research
- Good news:
  - Researchers are working on it
  - They are using PlanetLab
  - They make their tools available for you
- ... and, of course, you can write tools as well.



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

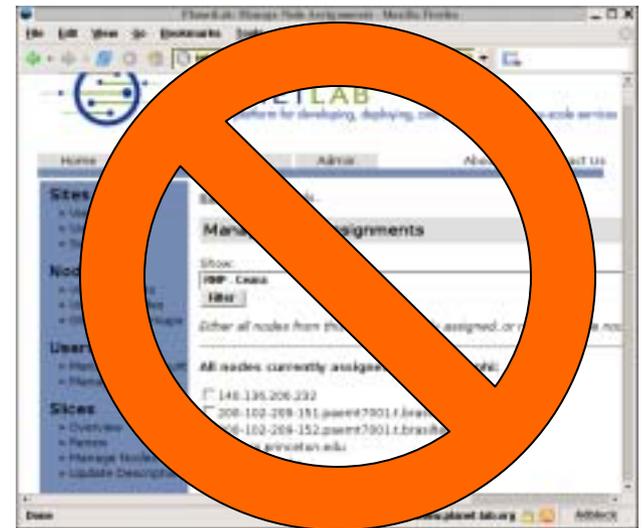
# Simple stuff: pssh

- Brent Chun's pssh package
- Provides parallel versions of:
  - ssh
  - scp
  - rsync
  - nuke (parallel "kill" with regexps)
- Simple way to control lots of slivers
- <http://www.theether.org/pssh/>



# Avoiding the web site

- Adding 300 nodes via web site is tedious (!)
- Solution: PlcApi
- XMLRPC interface
- All the web site's functionality
- Google for clients
- Or write your own!



# Deploying software

- AppManager (Ryan Huebsh, UCB)
  - Centralized monitoring and update of software
  - Uses cron job on each node, & polling
  - Simple, but highly effective!
  - <http://appmanager.berkeley.intel-research.net/>
- CoDeploy / CoBlitz
  - Use the Princeton CoDeeN CDN for efficient distribution of software to PlanetLab nodes
  - <http://codeen.cs.princeton.edu/codeploy/>
- Stork (U. Arizona)
  - Package management, uses CoDeploy for distribution

# Finding nodes

- PLCAPI (again)
  - Site, machine, model
  - Disk space, RSA keys,
  - Longitude, latitude,
  - Other config information...
- <http://www.planet-lab.org/xml/sites.xml>
  - Collated version of the above
  - Updated periodically

# Finding nodes

- CoMon (Princeton)
  - Extensive monitoring of node liveness
  - Also "cotop" and "cotest" CLI tools
- SWORD (U.C.Berkeley)
  - XML-based queries over nodes
  - Realtime resource state
  - <http://www.cs.berkeley.edu/~davidopp/sword/>

# Network measurement

- IPerf: network performance data  
<http://jabber.services.planet-lab.org/php/iperf/>
- All-pairs ping times between nodes
  - Run continuously by MIT
  - [http://www.pdos.lcs.mit.edu/~strib/pl\\_app/](http://www.pdos.lcs.mit.edu/~strib/pl_app/)
- Scriptroute (Neil Spring, UW)
  - Highly scriptable network measurement tool
  - <http://www.scriptroute.org/>



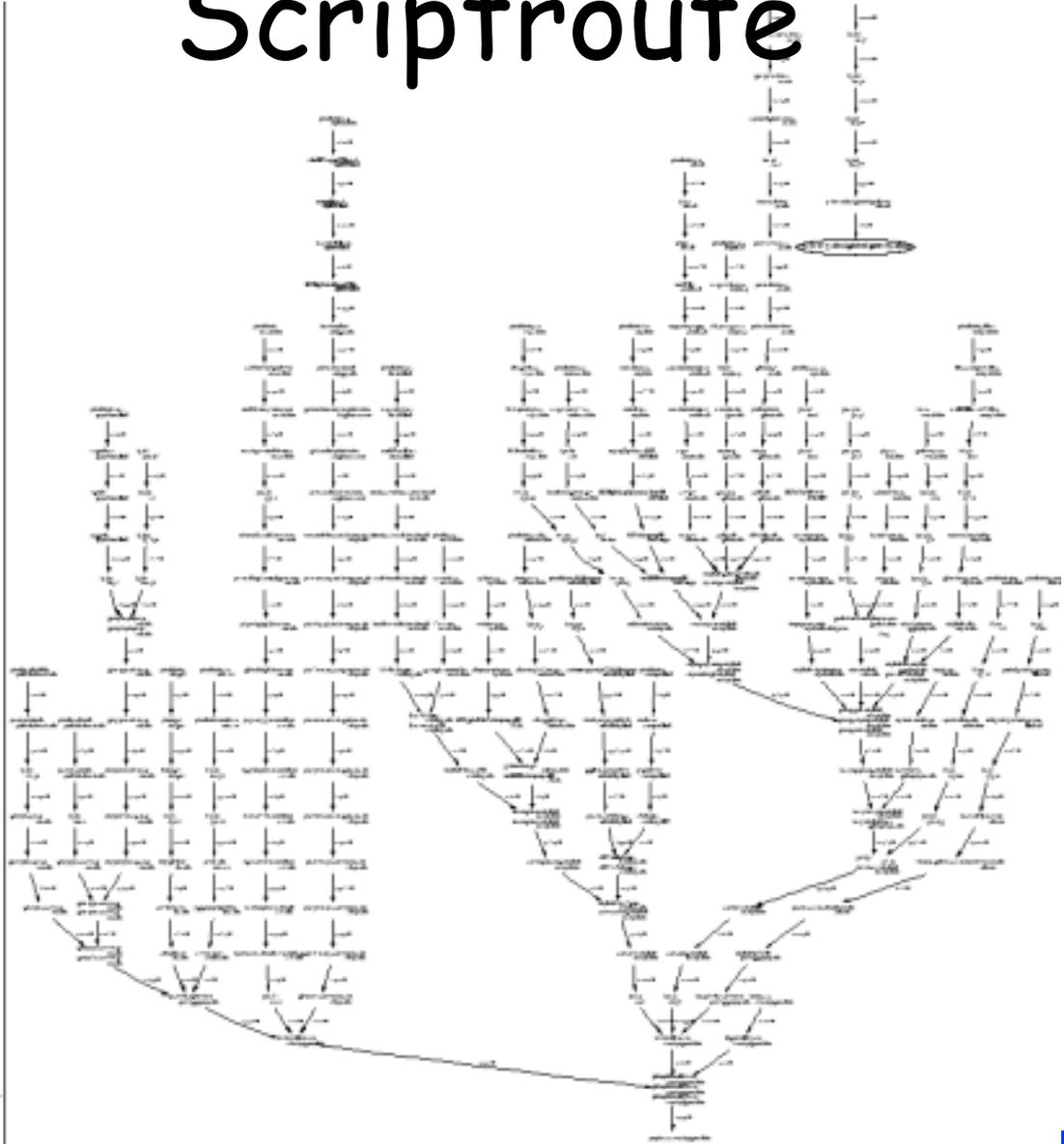
PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Scriptroute



# PlanetFlow

(point web browser at any node)



PlanetLab Node: planet2.berkeley.intel-research.net - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://planet2.berkeley.intel-research.net/slice.php resource discovery

Planetlab projects Pa... Downloads 3.3 Choosing Nodes PlanetLab Node: plan...

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

[Welcome](#) [Basic Search](#) [Advanced Search](#) [Browse By Slice](#) [Browse By Sou](#)

## Browse By Slice

Saturday, May 7, 2005

Click [Slice](#), [Flows](#), [MBytes](#), or [MBytes](#) to sort the table on that column, or to toggle the sorting order if the table is already sorted on that column. Click [Calculate...](#) to view the detailed statistics for a particular row if they are not already visible.

<a href="#">Slice</a>	<a href="#">Flows</a>	<a href="#">Percent</a>	<a href="#">MBytes</a>	<a href="#">Percent</a>	<a href="#">MBytes</a>	<a href="#">Percent</a>	<a href="#">Source IPs</a>
<a href="#">columbia_asherman</a>	226	1.11%	2.39	43.05%	191.28	36.52%	1
<a href="#">icmp-reply</a>	246	1.21%	1.90	34.24%	152.15	29.05%	1
<a href="#">ucb_bamboo</a>	384	1.89%	0.59	10.73%	110.88	21.17%	1
<a href="#">princeton_codeen</a>	6857	33.73%	0.21	3.85%	14.73	2.81%	1
<a href="#">ucb_srhea</a>	316	1.55%	0.12	2.13%	14.45	2.76%	1
<a href="#">irb_phi</a>	210	1.03%	0.15	2.65%	11.86	2.27%	1
<a href="#">rice_epost</a>	742	3.65%	0.02	0.28%	10.47	2.00%	1
...	...	...	...	...	...	...	...

Done AdBlock

# Resource Allocation

- PlanetLab has a rich resource allocation model
  - CPU, network, disk, etc.
  - Resources can be traded
- Several resource allocation services are emerging
- If you need more than the default, best-effort resources...



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel **Research**  
Berkeley

intel®

# Resource Allocation

- Bellagio (U.C. San Diego):
  - Combinatorial auctions for resources
  - <https://bellagio.ucsd.edu/>
- Sirius (U. Georgia):
  - Calendaring service: reserve hard resources for a limited time
  - <http://snowball.cs.uga.edu/~dkl/pslogin.php>

# Summary

- Brief tour of some tools and services for PlanetLab
- There are others: check the website, mailing lists, etc.
- More are on the way
- Please contribute your own!

# Part V: Advanced Networking

# Networking research on PlanetLab

- IP-level network research on PlanetLab is a bit different from on a dedicated machine
  - Each node is shared by many experiments
  - Sites limit bandwidth
  - Slices are restricted in what packets they can send



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel **Research**  
Berkeley

intel®

# Raw Sockets

- PlanetLab uses VNET for network isolation
- Good news:
  - you can open raw sockets as normal
- Bad news:
  - you won't see everyone's packets, just your own
  - you won't be able to send arbitrary packets, just ones you could have sent anyway from a socket.
- Still: you can ping, traceroute, run a user-space TCP stack, etc...



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Raw sockets part 2

- Slices with special privileges can open "true" raw sockets
  - Send / receive arbitrary packets
- "Proxy sockets" provide access to unused "dark" IP addresses
  - Used for network telescope experiments
- Capabilities handed out to "trusted" slices by the PlanetLab Consortium



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Well-known port numbers

- Each PlanetLab node has a single globally routable IP address
- Hundreds of services run simultaneously
- Q. Who allocates TCP/UDP port numbers across all nodes?
- A. Right now - informally via a Wiki...



PLANETLAB

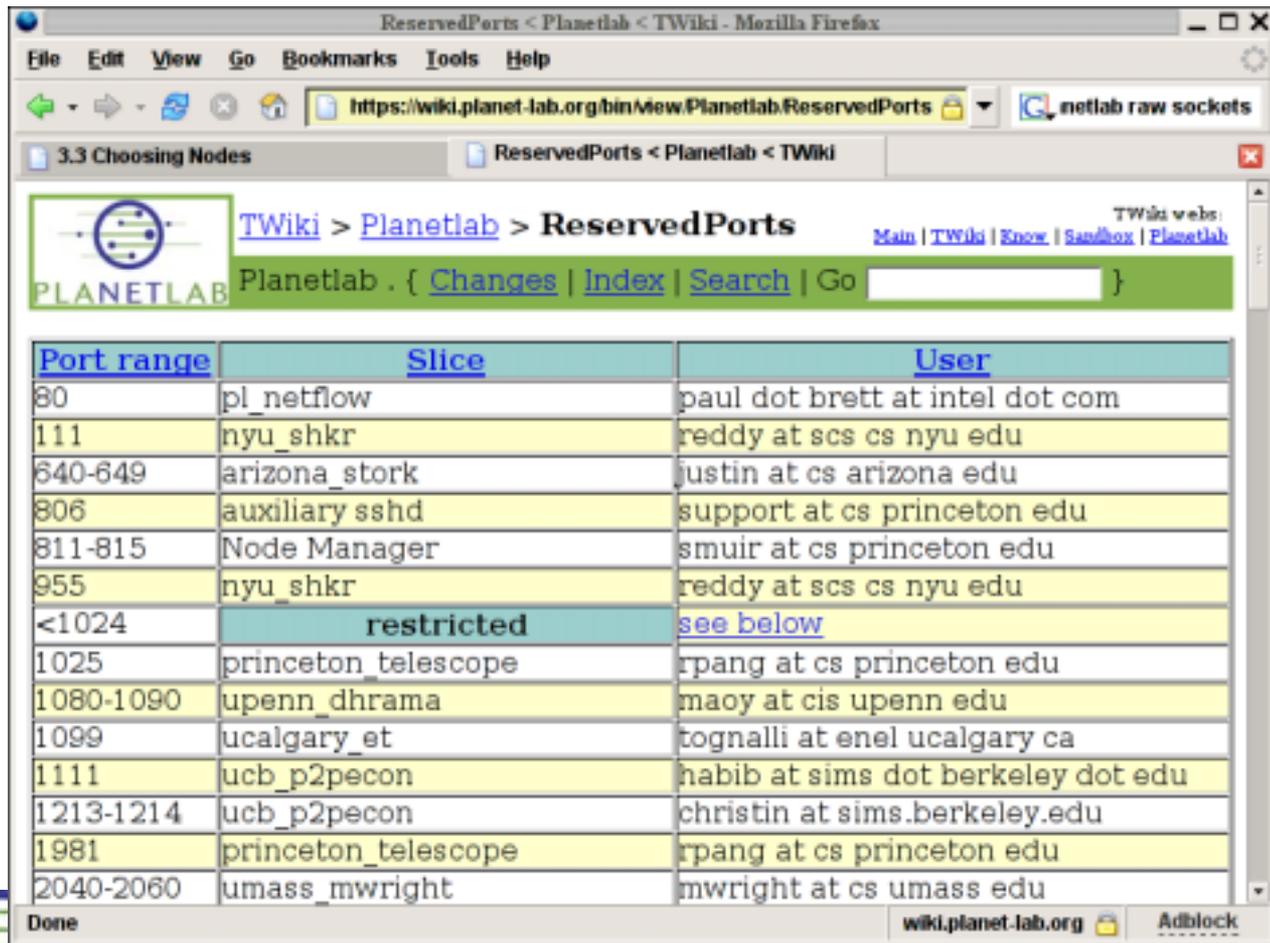
Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Reserved Port Numbers

sign up and claim a port number



Port range	Slice	User
80	pl_netflow	paul dot brett at intel dot com
111	nyu_shkr	reddy at scs cs nyu edu
640-649	arizona_stork	justin at cs arizona edu
806	auxiliary_sshd	support at cs princeton edu
811-815	Node Manager	smuir at cs princeton edu
955	nyu_shkr	reddy at scs cs nyu edu
<1024	restricted	<a href="#">see below</a>
1025	princeton_telescope	rpang at cs princeton edu
1080-1090	upenn_dhrama	maoy at cis upenn edu
1099	ucalgary_et	tognalli at enel ucalgary ca
1111	ucb_p2pecon	habib at sims dot berkeley dot edu
1213-1214	ucb_p2pecon	christin at sims.berkeley.edu
1981	princeton_telescope	rpang at cs princeton edu
2040-2060	umass_mwright	mwright at cs umass edu

# Network measurement

- Like all comedy, often a question of *timing*
- PlanetLab nodes use NTP
  - But sometimes off - see CoMon
- CPU intensive tasks make exact timestamping difficult
  - Libpcap at least stamps packets in kernel
  - May need to busywait to send e.g. packet pairs
- Use Scriptroute where possible
  - This is what it is designed for.



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Bandwidth is capped

- For many sites, bandwidth is by far the dominant cost of PlanetLab
- Per-node & per-slice b/w caps
- Note that you're sharing the link anyway
- Consider carefully what it means to measure the bandwidth!

# What is your bandwidth cap?

- If you really need to know:

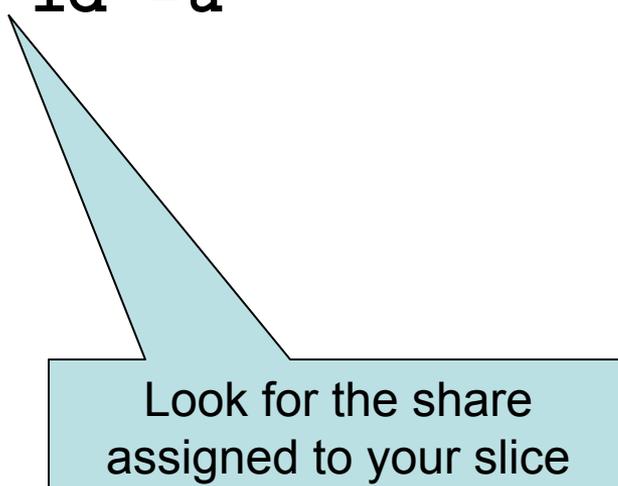
```
/sbin/tc -s -d class show dev eth0 \  
| grep 1: `id -u`
```

PlanetLab uses Linux  
Traffic Control (tc) for  
hierarchical fair queuing

# What is your bandwidth cap?

- If you really need to know:

```
/sbin/tc -s -d class show dev eth0 \  
| grep 1: `id -u`
```

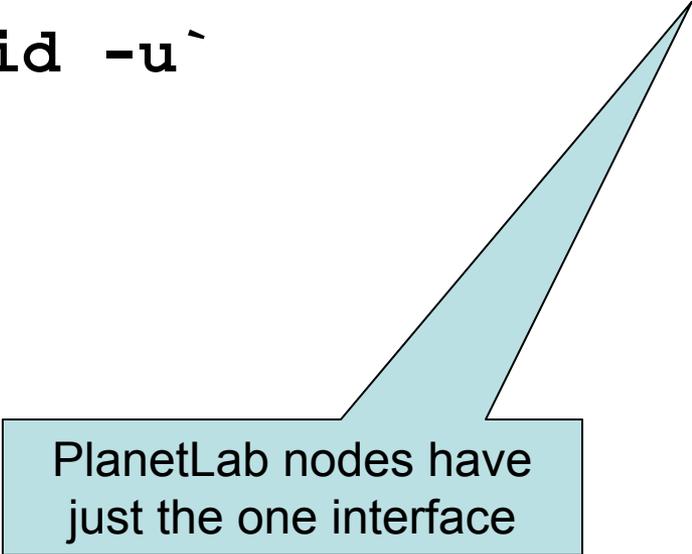


Look for the share  
assigned to your slice

# What is your bandwidth cap?

- If you really need to know:

```
/sbin/tc -s -d class show dev eth0 \  
| grep 1: `id -u`
```



PlanetLab nodes have  
just the one interface

# What is your bandwidth cap?

```
irb_boinc@planet1:~  
^ troscoe@gilles:~$ ssh irb_boinc@planet1.berkeley.intel-research.net  
Last login: Thu May 5 23:49:08 2005 from user-11fa28c.dsl.mindspring.com  
[irb_boinc@planet1 irb_boinc]$ /sbin/tc -s -d class show dev eth0 | grep 1: | `id -u`  
class htb 1:698 parent 1:1 leaf 698; prio 0 quantum 1540 rate 1Kbit ceil 10Mbit  
burst 1600b/8 mpu 0b cburst 30797b/8 mpu 0b level 0  
[irb_boinc@planet1 irb_boinc]$
```

This slice is capped at 10Mbit  
on this node

See 'man tc' for the other  
parameters

# Firewalls and NATs

- No PlanetLab nodes are NATed
  - All have global IPv4 addresses
- Sites are *requested* not to filter any external access to ports
  - In practice, many filter ICMP
  - Few filter TCP/UDP ports
- Exception: access to the local institution
  - Often: software or library licensing based on IP addresses ☹



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# What about IPv6?

- Wasn't a priority early on
- Also, US-centric
  - ⇒ not supported, little demand
- All that changed with China!
- Chinese team at Tsinghua University working on IPv6 support for all PlanetLab infrastructure
- Watch this space...



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Summary: Part V

- Socket programming is mostly conventional on PlanetLab
- But remember:
  - The node is shared with other researchers
  - Limits have been imposed on bandwidth

# Part VI: Methodology Issues

# PlanetLab as one of many research tools

- Network simulators
  - Ns2, SSF, OpNet,...
- Message-level simulators
  - Often ad-hoc, app-specific
- Cluster-based emulation
  - Emulab, Netbed
  - ModelNet
- Etc.



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel **Research**  
Berkeley

intel®

# PlanetLab: the new NS2?

- PlanetLab is slowly changing the publishing culture
  - "real" systems must now be deployed
  - This is good...
- The danger is:
  - PlanetLab becomes *the* requirement
  - PlanetLab comes to *define* networking reality

# PlanetLab Fallacies

- Symptom of its success and enthusiastic reception!
- No “methodology” consensus yet
  - Interesting area in itself
- Big difference between:
  - What PlanetLab can *teach*
  - What PlanetLab can *prove*
- Need a more critical approach to testbeds in research...



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Fallacy #1: PlanetLab is representative of the Internet

- PlanetLab is skewed towards:
  - The U.S.
  - The GREN
  - Well-connected commercial sites
- Measurement-related work on PlanetLab which extrapolates to the Internet must be **careful** in its claims!



PLANETLAB

Timothy Roscoe, 8 May 2005

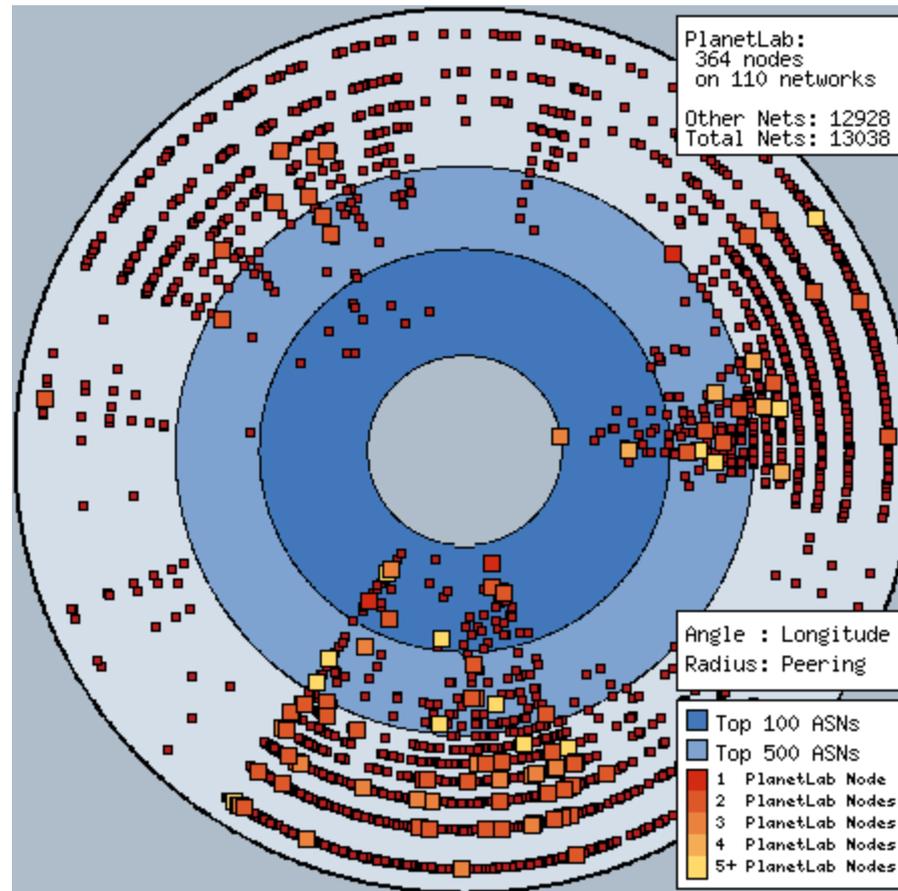
Intel  
Research  
Berkeley

intel®

# Of course, is the Internet itself representative?

- Enterprise networks are very different
- The (public) Internet is a very diverse environment anyway
- Hence:
  - What claims is the research making?
  - What do PlanetLab results do to substantiate those claims?

# PlanetLab presence (rather out of date)



# Fallacy #2: Quantitative results from PlanetLab are valid

- Experiments on PlanetLab are never fully repeatable:
  - Network conditions, machine load...
  - Comparative system measurements on PlanetLab are rarely credible
- Instead, more rigorous approach:
  - Simulation, emulation, and deployment *cross-validation*
  - ModelNet, Emulab
  - Long-term studies



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Fallacy #3. Quantitative results on PlanetLab are never valid

- Longitudinal studies of service behaviour over time
- Characterize the network conditions
- Run different approaches in parallel for long periods

# Reasons to be sceptical when reviewing papers:

- "As well as simulation results, we have run our system on PlanetLab"
  - Were the results commensurable?
  - What was learnt from this?
  - What difficulties were encountered?
  - Does PlanetLab match the motivating scenario?
  - Did it *really* work?



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# Reasons to be sceptical when reviewing papers:

- "We present results from running on  $n$  nodes of PlanetLab"
  - Why is  $n \ll 100$ ?
  - How were the nodes selected?
  - Were the nodes "cherry-picked"?
  - How is scalability beyond  $n$  being demonstrated?
  - Did it *really* work?



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# What PlanetLab can teach

- Discover new “issues” with system designs
  - E.g. DHT stability
  - Not easily found with simulations
- Highlight bad assumptions about the real network
- Derive well-grounded principles and abstractions for building real systems
- Attract real users, real workloads, and their challenges.



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
Research  
Berkeley

intel®

# PlanetLab *is...*

- Almost the only platform so far to:
  - Capture the systems challenges in wide-area distributed systems
  - Test what works and what doesn't in the wide area
  - Enable researchers to deploy long-running broad-coverage services and attract real users

# Part VII: Where to go for more information

# Basic documentation

- Web site: <http://www.planet-lab.org/>
  - FAQ
  - Wiki (contribute!)
  - Guides
  - API documentation
- Other sites:
  - Danny Bickson's guide:  
<http://www.cs.huji.ac.il/labs/danss/planetlab/PlanetlabProjectHowto.pdf>
  - Ian Wakeman's tutorial:  
<http://www.informatics.sussex.ac.uk/research/ngn/slides/planetlab05talks/tutorial/>
- Google is your friend...

# Not sure what you're doing?

- The users' list:
  - [users@lists.planet-lab.org](mailto:users@lists.planet-lab.org)
- Moderated
- V. High signal to noise ratio
- Surprisingly helpful people!
- ALL users should subscribe.

# Something seems to be broken?

- Support list:
  - [support@planet-lab.org](mailto:support@planet-lab.org)
- Front-end to trouble ticket system at Princeton
- Responses within 24 hours or so
- Sometimes responses go to the users list



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel **Research**  
Berkeley

intel®

# Want to get more involved?

- The support community list:
  - [support-community@planet-lab.org](mailto:support-community@planet-lab.org)
- Copy of mail sent to [support@planet-lab.org](mailto:support@planet-lab.org), plus intra-support traffic
- Chance to see how much work they really do!

# Want to get much more involved?

- The architects' list:
  - [arch@lists.planet-lab.org](mailto:arch@lists.planet-lab.org)
- Discussion of architectural directions and technical decisions
- Come to PlanetLab meetings
  - Or host one!
- Write proposals up as PDNs



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel **Research**  
Berkeley

intel®

# If you don't like it, fix it!

- Infrastructure services can be run by any PlanetLab user
- Ask PLC nicely if you need extra access privileges
  - E.g. ability to create slices
- Tell people about your service on [users@lists.planet-lab.org](mailto:users@lists.planet-lab.org)
- Complain about why you can't build your service on [arch@lists.planet-lab.org](mailto:arch@lists.planet-lab.org)



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel  
**Research**  
Berkeley

intel®

# If you don't like it, fix it!

- Download "PlanetLab-in-a-box"
- All node OS code available from <https://cvs.planet-lab.org/>
- Requirement for a node is only that it implements the node manager interface (mostly).

# Above all...

- Have fun!
- Enjoy the experience of running a service for real
- Go and build useful things
- Do interesting research
- Remember: PlanetLab is a community
  - It's yours as much as anyone else's.



PLANETLAB

Timothy Roscoe, 8 May 2005

Intel **Research**  
Berkeley

intel®

# Thanks!

Timothy Roscoe (Mothy)  
[timothy.roscoe@intel.com](mailto:timothy.roscoe@intel.com)



Timothy Roscoe, 8 May 2005

