October 5th 2010 – NANOG 50

Jason Zurawski – Internet2

# The pS Performance Toolkit

# Overview

- Introduction and Motivation
- Design & Functionality
- Current Deployment Footprint
- Conclusion

# Why Worry About Network Performance?

- Why does it matter?
  - Flows getting larger (e.g. *Science datasets* in the R&E world)
  - Special requirements (e.g. *Streaming media* is sensitive to jitter)
  - Number of users/devices is increasing
  - ***Everything*** is cross domain
- Where are the problems?
  - *Network Core*? Everything is well connected, well provisioned, and flawlessly configured, ***RIGHT***?
  - *End Systems*? Properly tuned for optimal TCP performance (no matter the operating system), proper drivers installed and functioning optimally, ***RIGHT***?
- Performance Debugging = Difficult Problem
  - Which Tools to Use?
  - Availability and Location of Tools
  - Facilitate the Sharing of Results – could be useful to others…

# Why Worry About Network Performance?

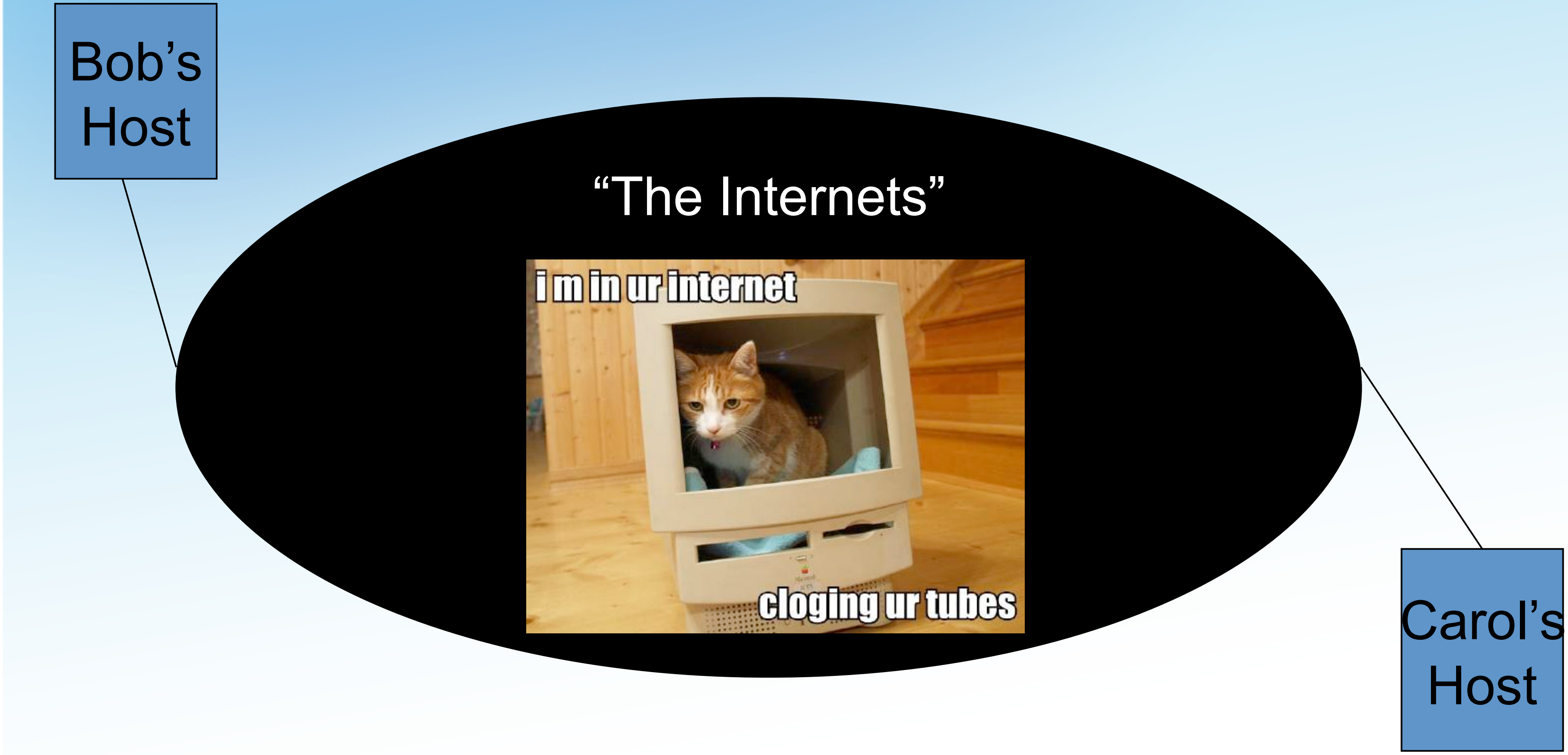
- Most network design lends itself to the introduction of flaws:
  - Heterogeneous equipment
  - Cost factors heavily into design – e.g. *Get what you pay for*
  - Design heavily favors ***protection*** and ***availability*** over performance
- Communication protocols are not advancing as fast as networks
  - *TCP/IP* is still the king of the protocol stack
    - Guarantees reliable transfers
    - Adjusts to failures in the network
    - Adjusts speed to be *fair* for all
- End Systems
  - Good enough for ***web*** and ***email***?
- User Expectations
  - “The Network is Slow/Broken”
  - Empower users to be more informed/more helpful

# Underlying Assumption

- When problems exist, it's the networks fault!
  - Easy to blame a resource, but where is the real problem?
    - Host (Disk, CPU, Kernel, NIC Drivers)
    - Network Interface Cards (Host and Network Device)
    - Routers/Switches Fabric, Routing Algorithms, Configurations
    - Physical Infrastructure (Fibers, Connectors)
    - Protocols

- The network is viewed as a single resource in many cases
  - **Reality** – complex series of components
  - Multiple vendors/technologies
  - Multiple configuration options
  - Crossing administrative domains

# Network View (Layman’s Terms)

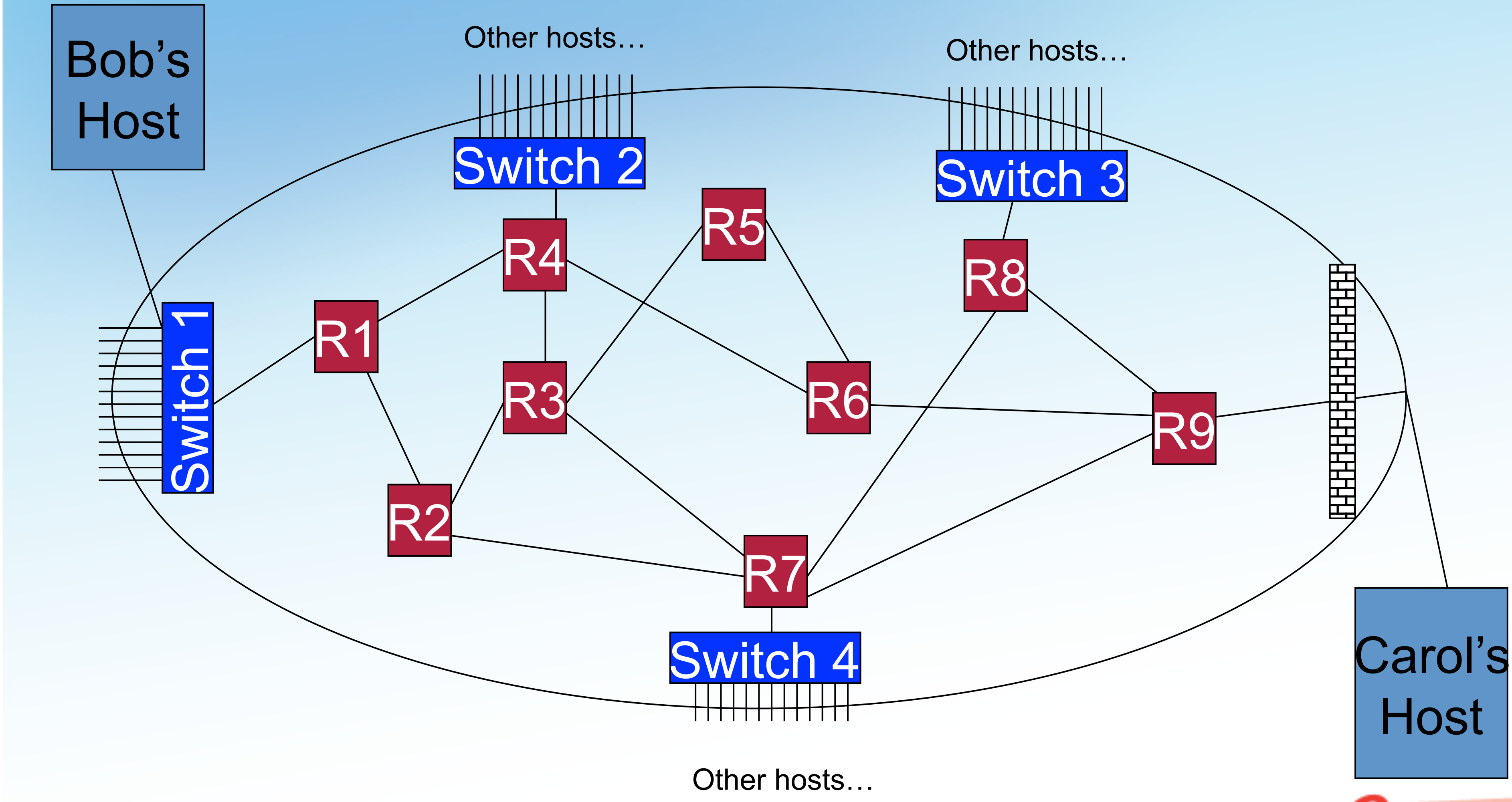


Bob’s Host

“The Internets”

Carol’s Host

# Network View (Actual)



# A New Way Forward

- Encourage use of existing tools
  - Don't need new tools, lots of fine measurement software already
  - Work with what "others are using", because it probably works…
  - Facilitate use of remote tools and sharing of resources

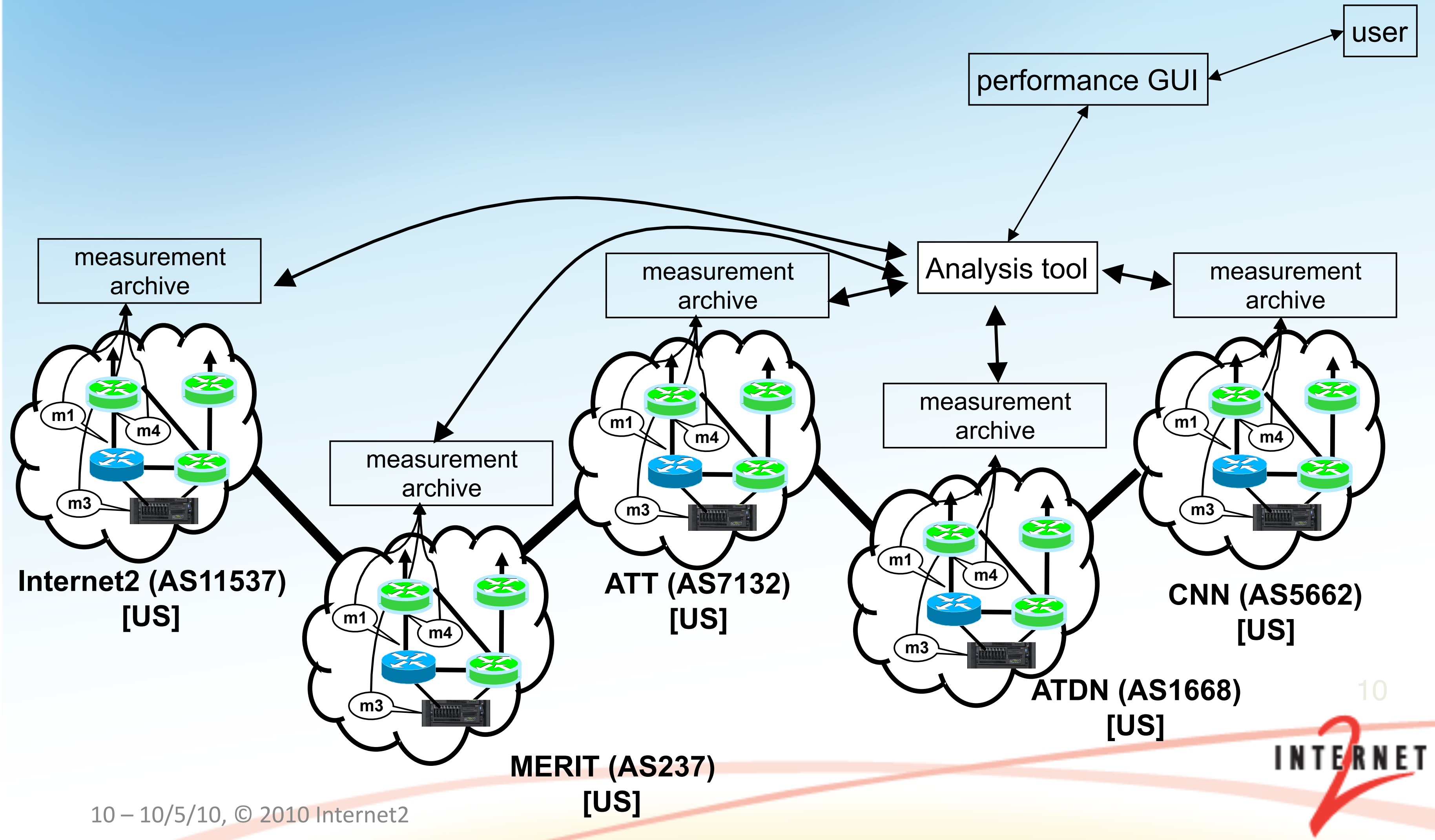
- Single Step Deployment
  - ***Set it and forget it!***
  - Ease of installation = wider deployment

- Debugging problems – end to end
  - Instrumentation and monitoring ***on all parts of the path***
  - Data views that correlate results automatically, suggest solutions

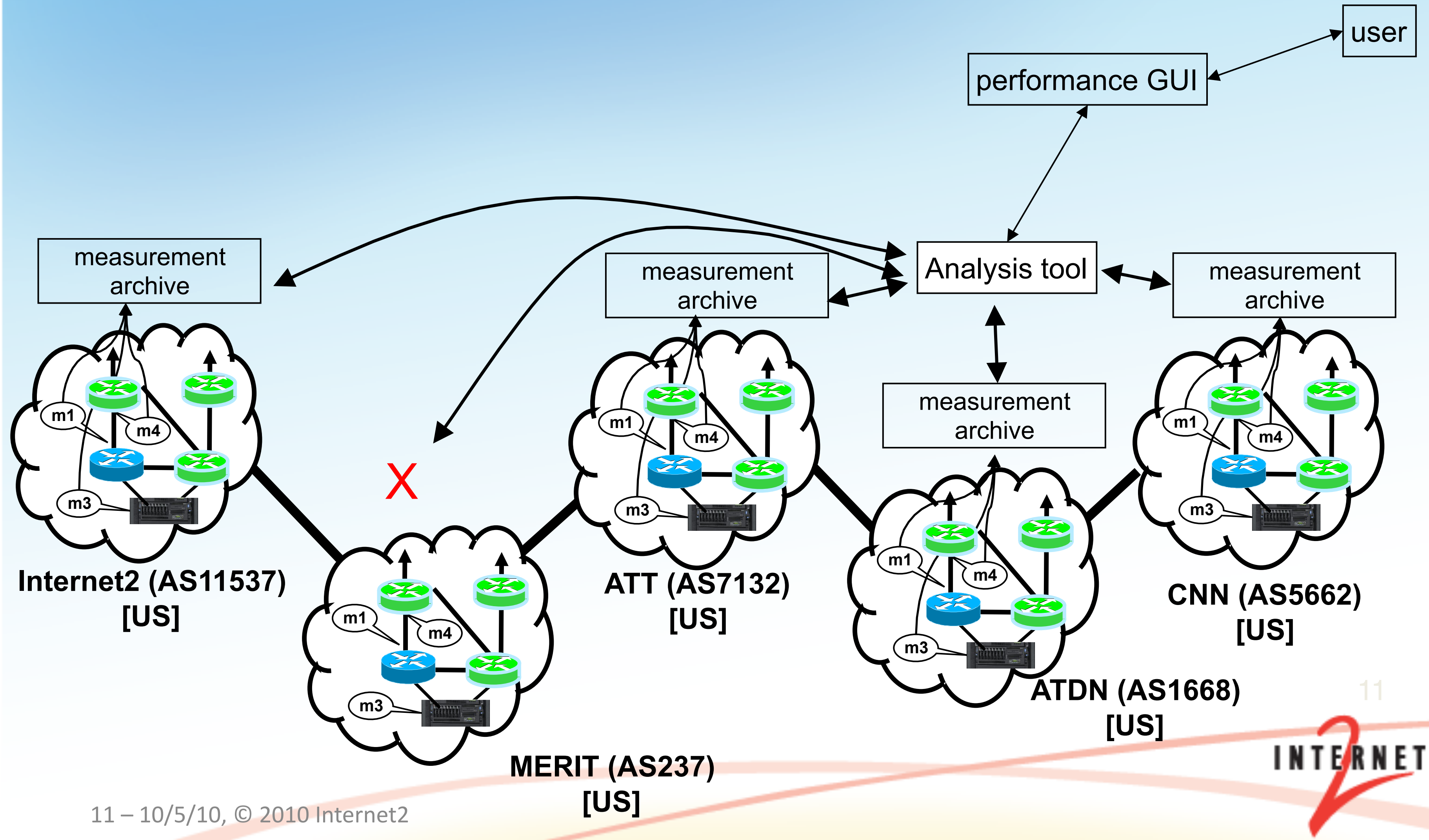
# Possible Solutions

- Finding a solution to network performance problems can be broken into two distinct steps:
  - Use of *Diagnostic Tools* to locate problems
    - Tools that actively measure performance (e.g. Latency, Available Bandwidth)
    - Tools that passively observe performance (e.g. error counters)
  - *Regular Monitoring* to establish performance baselines and alert when expectation drops.
    - Using diagnostic tools in a structured manner
    - Visualizations and alarms to analyze the collected data
- Incorporation of either of these techniques must be:
  - *ubiquitous*, e.g. the solution works best when it is available everywhere
  - seamless (e.g. *federated*) in presenting information from different resources and domains

# Everyone *should* participate in the monitoring infrastructure



# Concept is still ***possible*** with partial information



# Possible Solutions

- Desirable design features for any solution
  - Component Based
    - Functionality should be split into logical units
    - Each function (e.g. visualization) should function through well defined communication with other components (e.g. data storage)
  - Modular
    - Monolithic designs rarely work
    - Components allow choice of how to operate a customized end solution.
  - Accessible
    - Well defined interfaces (e.g. APIs)
    - Easily installed and configured

- Initial design should facilitate future expansion

# What is perfSONAR?

- A collaboration
  - Production network operators focused on designing and building tools that they will deploy and use on their networks to provide monitoring and diagnostic capabilities to themselves and their user communities.
- An architecture & set of communication protocols
  - Web Services (WS) Architecture
  - Protocols established in the Open Grid Forum
    - Network Measurement Working Group (NM-WG)
    - Network Measurement Control Working Group (NMC-WG)
    - Network Markup Language Working Group (NML-WG)
- Several interoperable software implementations
  - perfSONAR-MDM
  - perfSONAR-PS
- A Deployed Measurement infrastructure

# perfSONAR Inception

- *perfSONAR* originated from discussions between Internet2's End-to-End Performance Initiative (*E2Epi*), and the Géant2 project in September 2004.
- Members of the OGF's (then GGF) NM-WG provided guidance on the encoding of network measurement data.
- Additional network partners, including ESnet and RNP provided development resources and served as early adopters.
- The first release of *perfSONAR* branded software was available in July 2006 (Java based software).
- All *perfSONAR* branded software is open source
- All products looking to be labeled as *perfSONAR compliant* must establish protocol compliance based on the public standards of the OGF

# perfSONAR Architecture Overview

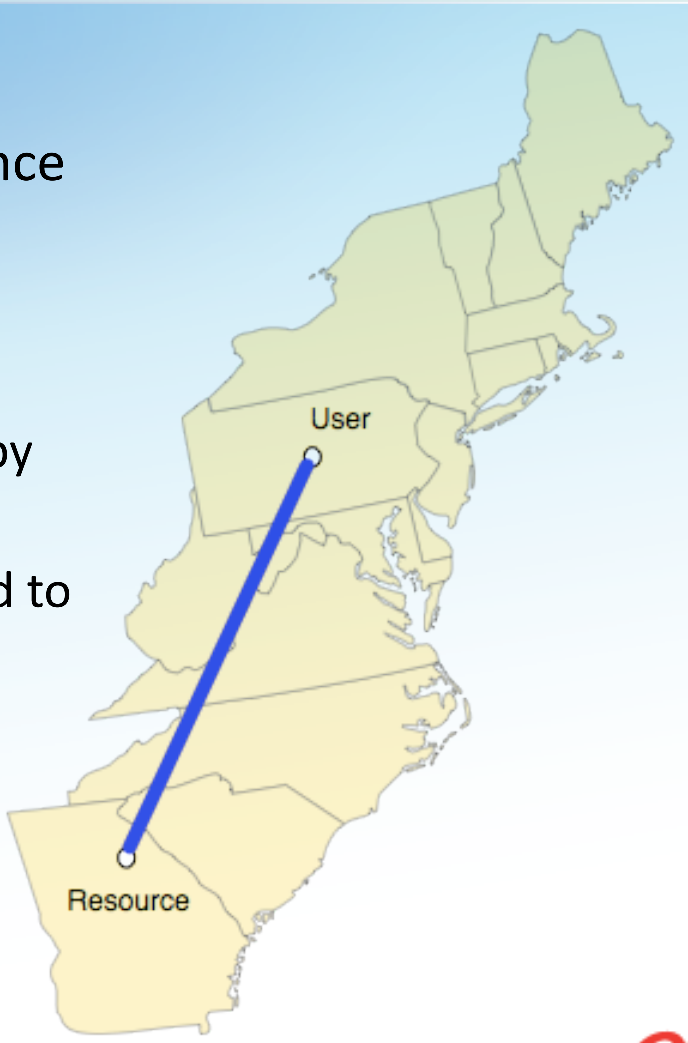
- Interoperable network measurement middleware designed as a Service Oriented Architecture (SOA):
  - Each component is modular
  - All are Web Services (WS) based
  - The global *perfSONAR* framework as well as individual deployments are decentralized
  - All *perfSONAR* tools are Locally controlled
  - All *perfSONAR* tools are capable of federating locally and globally
- perf*SONAR* Integrates:
  - Network measurement tools and archives (e.g. stored measurement results)
  - Data manipulation
  - Information Services
    - Discovery
    - Topology
  - Authentication and authorization

# perfSONAR Architecture Overview

- The key concept of *perfSONAR* is that each entity (e.g. "services") performs a function
  - Each service provides a limited set of functionality
    - Collecting measurements between arbitrary points
    - Managing the registration and location of distributed services
- Services interact through exchanges
  - Standardized message formats and protocols
- A collection of *perfSONAR* services within a domain is a ***deployment***
  - Deploying *perfSONAR* can be done *À la carte*, or through a complete solution
- Services federate with each other, locally and globally
  - Services are designed to automatically discover the presence of other *perfSONAR* components
  - Clients are designed with this distributed paradigm in mind

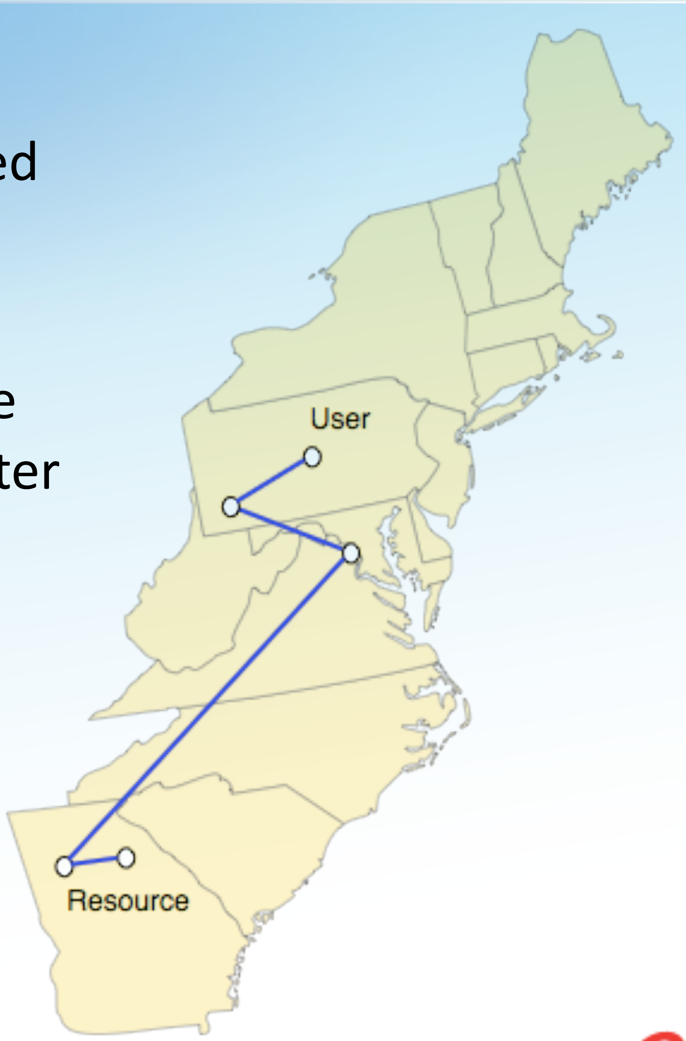
# Example perfSONAR Use Case

- perfSONAR should be used to diagnose an end-to-end performance problem
  - User is attempting to download a remote resource
  - Resource and user are separated by distance
  - Both are assumed to be connected to high speed networks
- Operation does not go as planned, where to start?



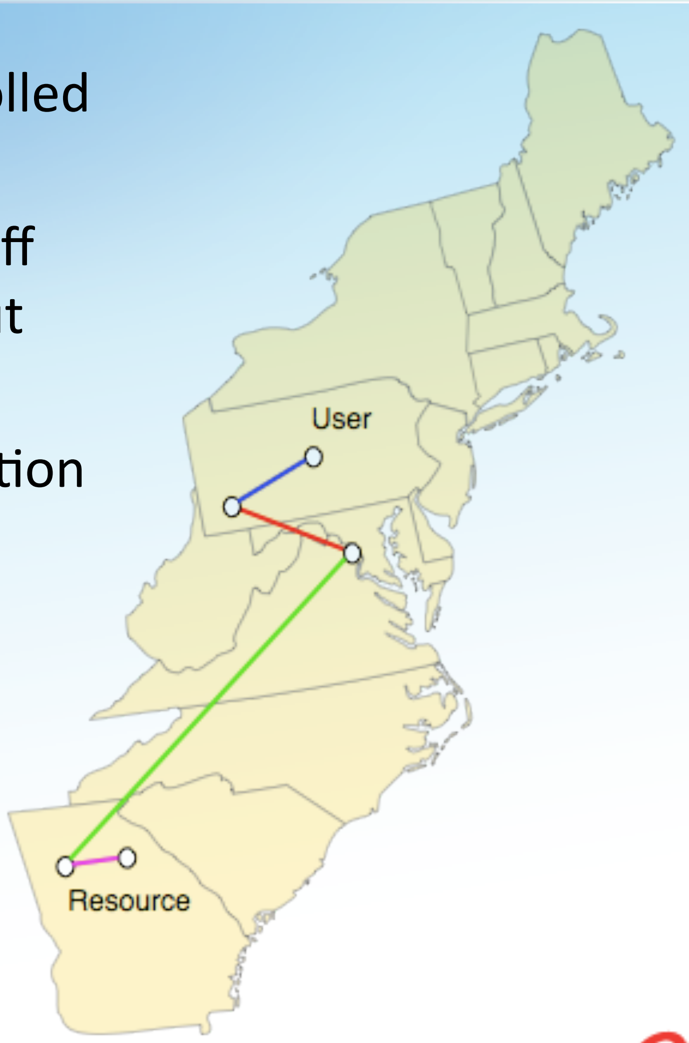
# Example perfSONAR Use Case

- Simple tools like ***traceroute*** can be used to determine the path traveled
- There could be a performance problem anywhere in here
- The problem may be something we could fix, but the chances are greater that it is not



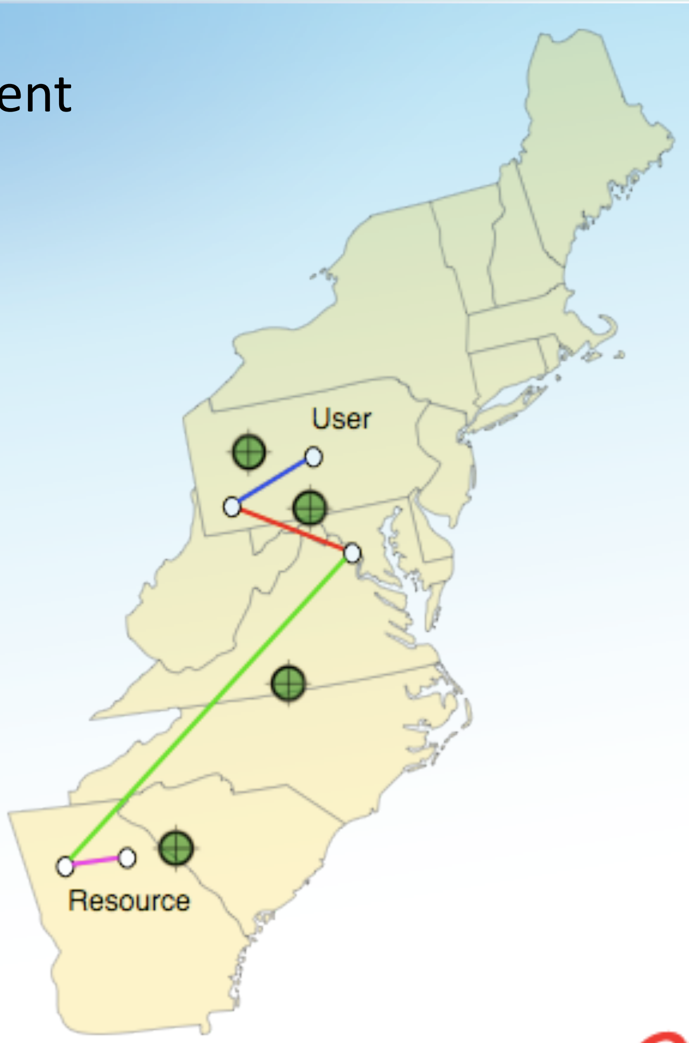
# Example perfSONAR Use Case

- Each segment of the path is controlled by a different domain.
- Each domain will have network staff that could help fix the problem, but how to contact them?
- All we really want is some information regarding performance



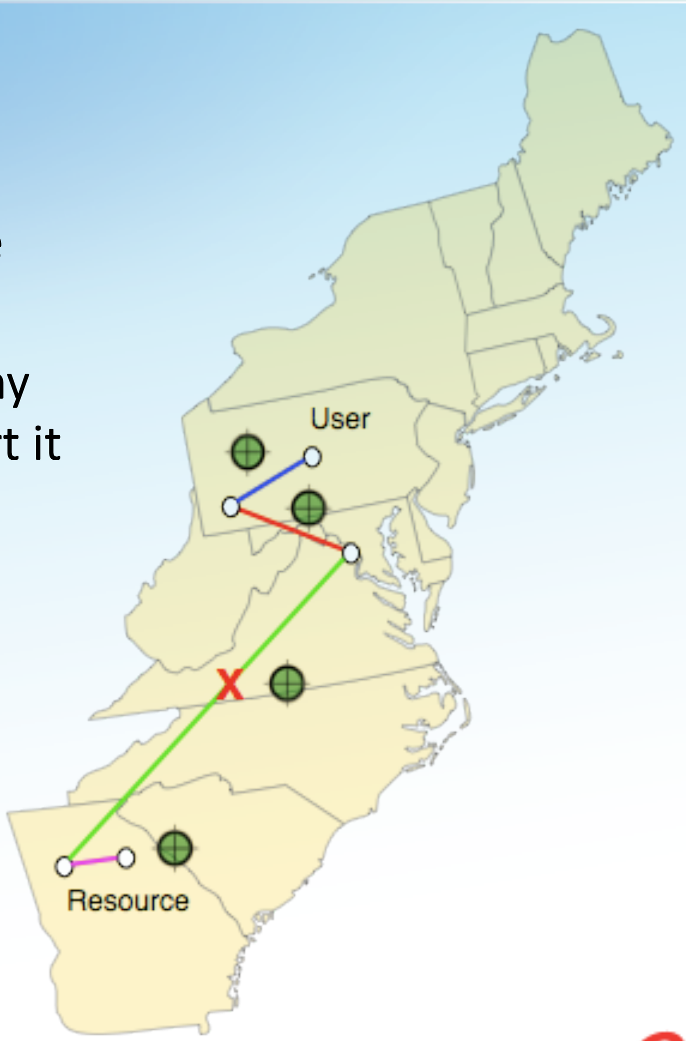
# Example perfSONAR Use Case

- Each domain has made measurement data available via perfSONAR
- The user was able to discover this automatically
- Automated tools such as visualizations and analyzers can be powered by this network data



# Example perfSONAR Use Case

- In the end, the problem is isolated based on testing.
  - May have gone unnoticed in some cases (e.g. a “soft failure”)
  - Could have been observed by many others … that didn’t think to report it
- The user (or operations staff) can contact the domain in question to inquire about this performance problem
- When fixed the transfer should progress as intended



# Overview

- Introduction and Motivation
- Design & Functionality
- Current Deployment Footprint
- Conclusion

# Complete Network Monitoring Solution

- Wide Array of Tools
  - Hit some major metrics (network utilization, latency, bandwidth)
  - Use what others are using (Cacti, ping, iperf)
- Installation
  - ***One Step*** – package management or *all in one* appliance
  - Linux Live CD – Popular way to try out before committing to a distro…
- Functionality
  - ***One off*** diagnostic testing
  - Regular monitoring
  - Alerting capability
  - Visualizations

# pS Performance Toolkit

- The ‘pS’ stands for ‘perfSONAR’
- http://psps.perfsonar.net/toolkit
- Current Version: 3.2
  - CentOS 5.5 Based
  - Previous generations: based on Knoppix
- Available Tools:
  - Latency: PingER/OWAMP/Reverse Traceroute
  - Bandwidth: BWCTL (Iperf/Nuttcp), NDT, NPAD
  - Passive Measurements: (Cacti)
- Purpose:
  - Install in any ‘free’ machine
  - Diagnostic = non destructive to host
  - Regular Monitoring = store results on host
- ***Live*** vs ***Disk Based*** Installs

# Support Structure

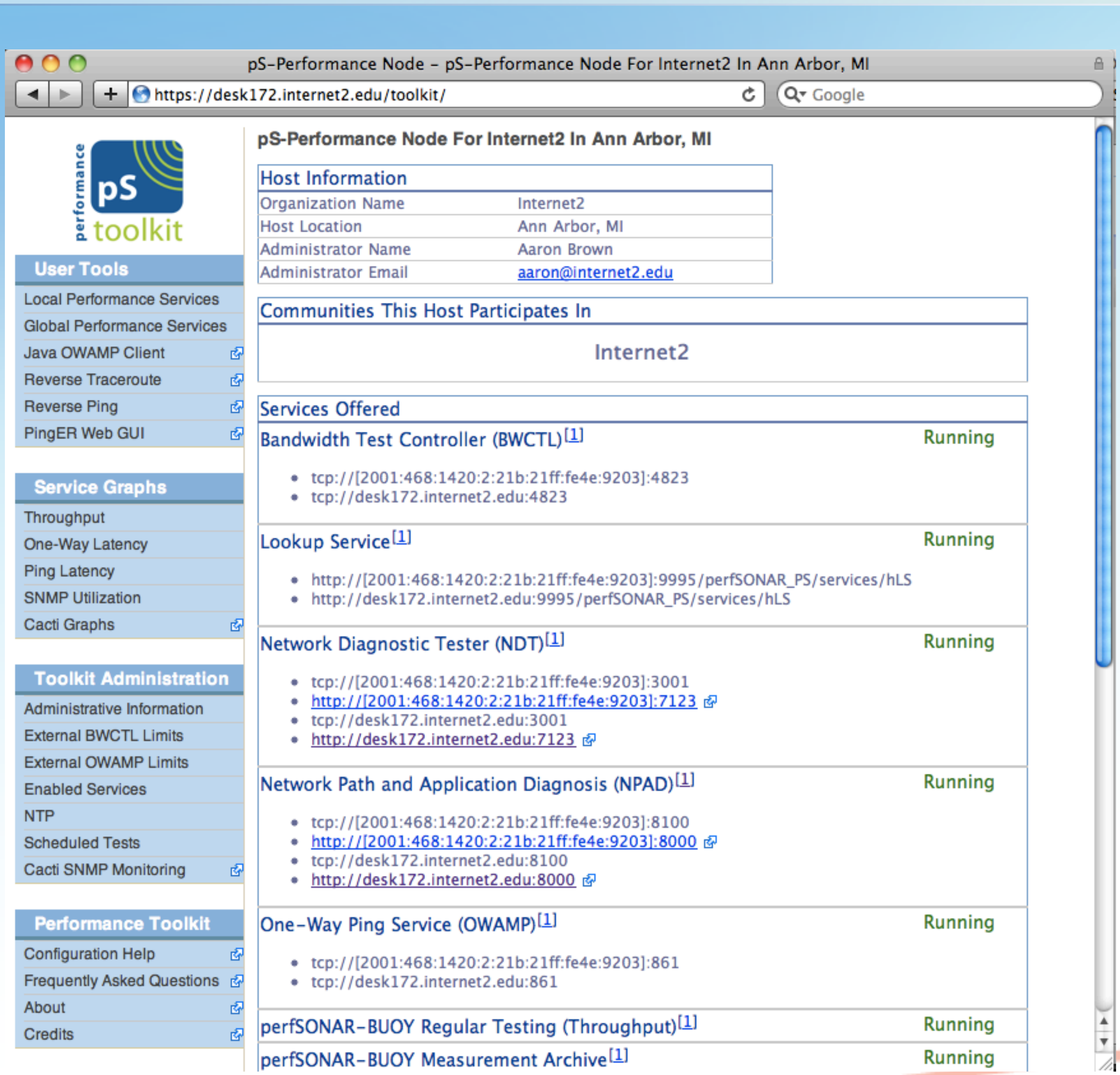
- Open Source Development
  - Team comprised of R&E Network operators, Universities, Federal Labs
  - All source code publically available though anonymous SVN
- Open Source Support
  - Mailing list(s)
  - Public bug tracking
- Upgrade Path
  - Burn a new CD
  - Use of Package Managers (YUM)
  - Migration tools to backup historical measurement data

# pS-Performance Toolkit – On Deck

- 6-12 Month time horizon
  - New Tools (Nagios, Ganglia) in the works
  - Improved GUIs for data analysis
    - Integrated data views
    - Network health views
  - Alert infrastructure for services and data
    - Notify if bad results are seen
  - Improved support for multilayer measurement correlation
  - Authentication and Authorization – Protect access to tools and data

- Beyond
  - Web interface for managing groups of pS-Performance Nodes
  - More diagnostic tools and archives
  - Integration of outside research and production tools

# pS Performance Toolkit – Web Interface



pS-Performance Node For Internet2 In Ann Arbor, MI

| Host Information | |
|---|---|
| Organization Name | Internet2 |
| Host Location | Ann Arbor, MI |
| Administrator Name | Aaron Brown |
| Administrator Email | aaron@internet2.edu |

Communities This Host Participates In

Internet2

Services Offered

Bandwidth Test Controller (BWCTL)[1] Running

- tcp://[2001:468:1420:2:21b:21ff:fe4e:9203]:4823
- tcp://desk172.internet2.edu:4823

Lookup Service[1] Running

- http://[2001:468:1420:2:21b:21ff:fe4e:9203]:9995/perfSONAR_PS/services/hLS
- http://desk172.internet2.edu:9995/perfSONAR_PS/services/hLS

Network Diagnostic Tester (NDT)[1] Running

- tcp://[2001:468:1420:2:21b:21ff:fe4e:9203]:3001
- http://[2001:468:1420:2:21b:21ff:fe4e:9203]:7123
- tcp://desk172.internet2.edu:3001
- http://desk172.internet2.edu:7123

Network Path and Application Diagnosis (NPAD)[1] Running

- tcp://[2001:468:1420:2:21b:21ff:fe4e:9203]:8100
- http://[2001:468:1420:2:21b:21ff:fe4e:9203]:8000
- tcp://desk172.internet2.edu:8100
- http://desk172.internet2.edu:8000

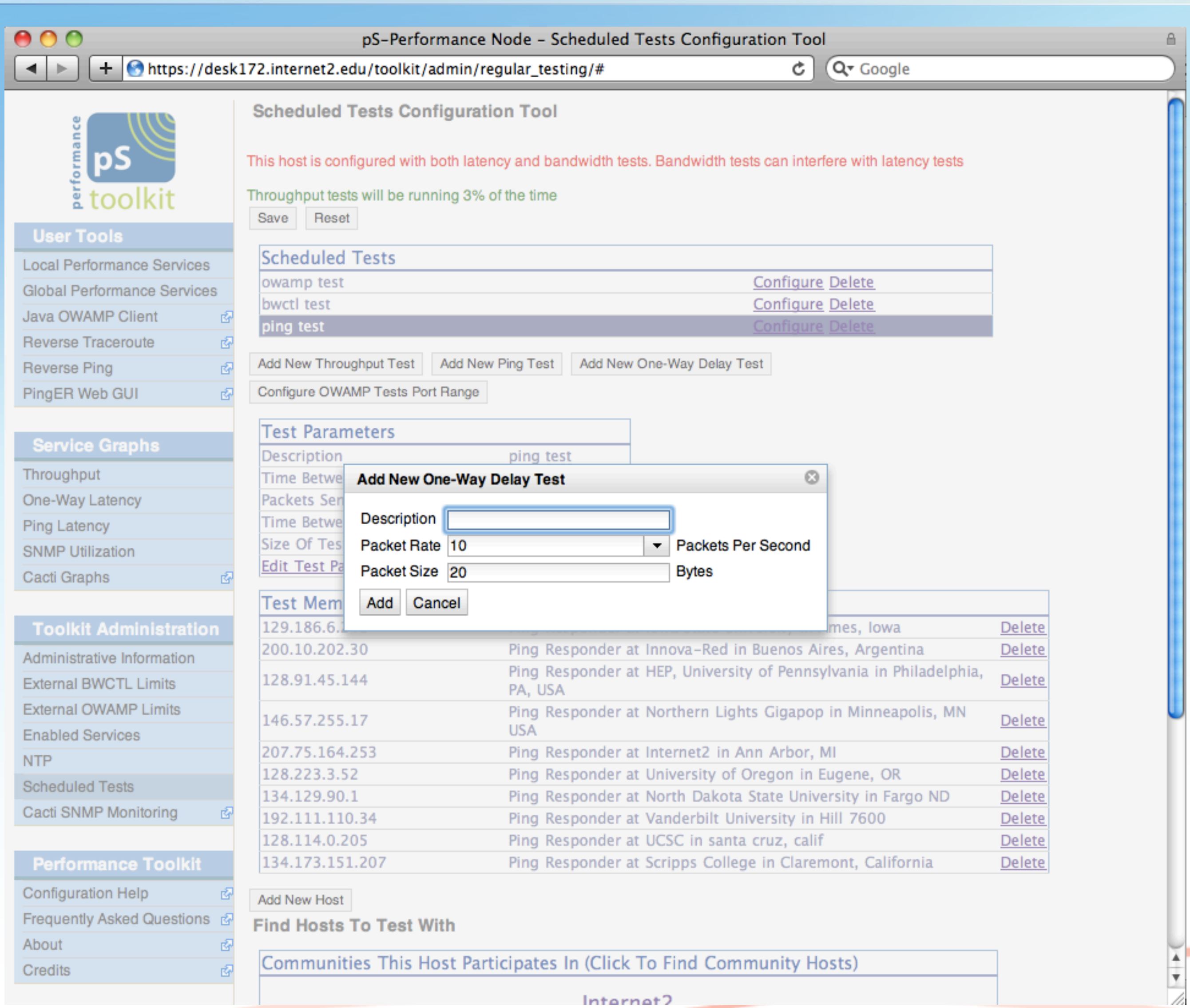
One-Way Ping Service (OWAMP)[1] Running

- tcp://[2001:468:1420:2:21b:21ff:fe4e:9203]:861
- tcp://desk172.internet2.edu:861

perfSONAR-BUOY Regular Testing (Throughput)[1] Running

perfSONAR-BUOY Measurement Archive[1] Running

# pS Performance Toolkit – Scheduling Interface



## Scheduled Tests Configuration Tool

This host is configured with both latency and bandwidth tests. Bandwidth tests can interfere with latency tests

Throughput tests will be running 3% of the time

| Scheduled Tests | |
|---|---|
| owamp test | Configure Delete |
| bwctl test | Configure Delete |
| ping test | Configure Delete |

| Test Parameters | |
|---|---|
| Description | ping test |
| Time Betwe | |
| Packets Sen | |
| Time Betwe | |
| Size Of Tes | |
| Edit Test Pa | |

**Add New One-Way Delay Test**

Description:

Packet Rate: 10 Packets Per Second

Packet Size: 20 Bytes

Add Cancel

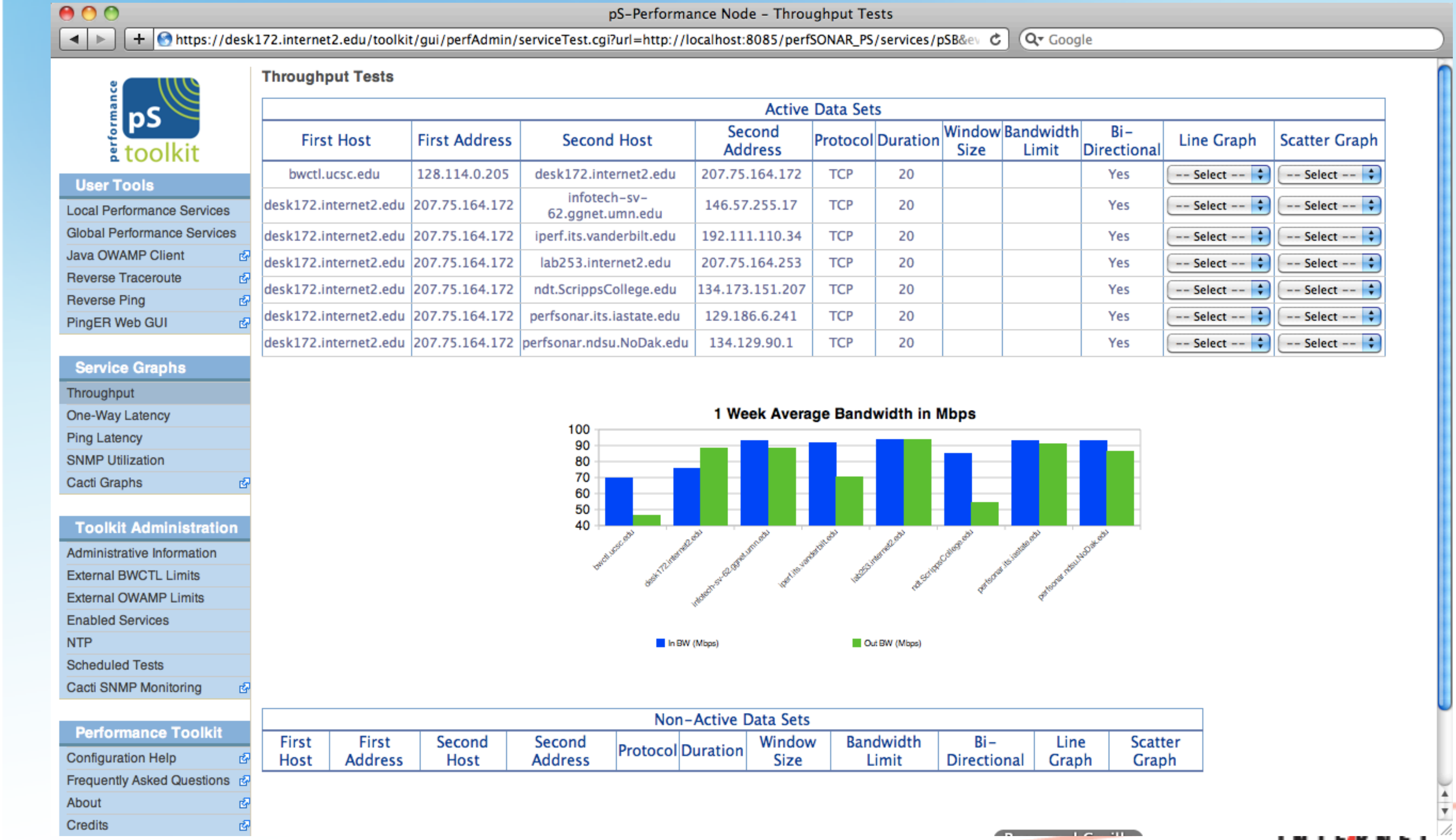
| Test Mem | | |
|---|---|---|
| 129.186.6. | mes, Iowa | Delete |
| 200.10.202.30 | Ping Responder at Innova-Red in Buenos Aires, Argentina | Delete |
| 128.91.45.144 | Ping Responder at HEP, University of Pennsylvania in Philadelphia, PA, USA | Delete |
| 146.57.255.17 | Ping Responder at Northern Lights Gigapop in Minneapolis, MN USA | Delete |
| 207.75.164.253 | Ping Responder at Internet2 in Ann Arbor, MI | Delete |
| 128.223.3.52 | Ping Responder at University of Oregon in Eugene, OR | Delete |
| 134.129.90.1 | Ping Responder at North Dakota State University in Fargo ND | Delete |
| 192.111.110.34 | Ping Responder at Vanderbilt University in Hill 7600 | Delete |
| 128.114.0.205 | Ping Responder at UCSC in santa cruz, calif | Delete |
| 134.173.151.207 | Ping Responder at Scripps College in Claremont, California | Delete |

### Find Hosts To Test With

Communities This Host Participates In (Click To Find Community Hosts)

Internet2

# pS Performance Toolkit – Data View


pS-Performance Node - Throughput Tests
https://desk172.internet2.edu/toolkit/gui/perfAdmin/serviceTest.cgi?url=http://localhost:8085/perfSONAR_PS/services/pSB&ev
Google
pS performance toolkit
1 Week Average Bandwidth in Mbps
100
90
80
70
60
50
40
bwctl.ucsc.edu
desk172.internet2.edu
infotech-sv-62.ggnet.umn.edu
iperf.its.vanderbilt.edu
lab253.internet2.edu
ndt.ScrippsCollege.edu
perfsonar.its.iastate.edu
perfsonar.ndsu.NoDak.edu
In BW (Mbps)
Out BW (Mbps)

**User Tools**

- Local Performance Services
- Global Performance Services
- Java OWAMP Client
- Reverse Traceroute
- Reverse Ping
- PingER Web GUI

**Service Graphs**

- Throughput
- One-Way Latency
- Ping Latency
- SNMP Utilization
- Cacti Graphs

**Toolkit Administration**

- Administrative Information
- External BWCTL Limits
- External OWAMP Limits
- Enabled Services
- NTP
- Scheduled Tests
- Cacti SNMP Monitoring

**Performance Toolkit**

- Configuration Help
- Frequently Asked Questions
- About
- Credits

**Throughput Tests**

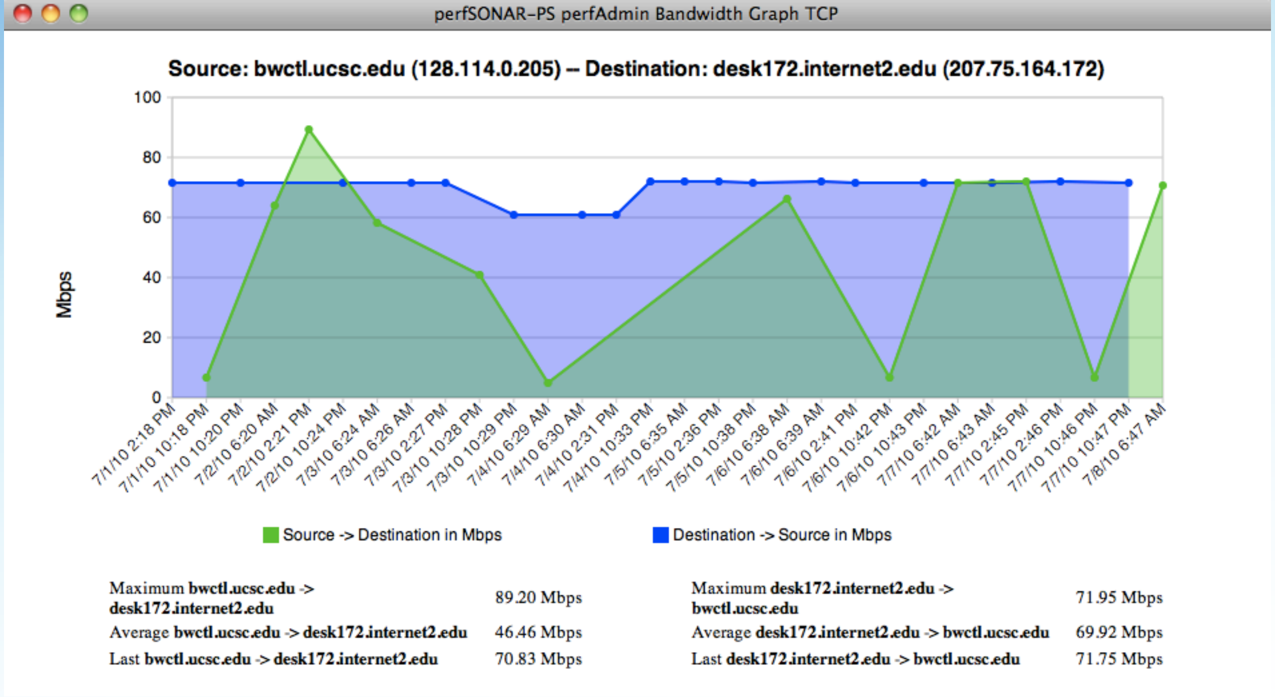
Active Data Sets

| First Host | First Address | Second Host | Second Address | Protocol | Duration | Window Size | Bandwidth Limit | Bi-Directional | Line Graph | Scatter Graph |
|---|---|---|---|---|---|---|---|---|---|---|
| bwctl.ucsc.edu | 128.114.0.205 | desk172.internet2.edu | 207.75.164.172 | TCP | 20 | | | Yes | -- Select -- | -- Select -- |
| desk172.internet2.edu | 207.75.164.172 | infotech-sv-62.ggnet.umn.edu | 146.57.255.17 | TCP | 20 | | | Yes | -- Select -- | -- Select -- |
| desk172.internet2.edu | 207.75.164.172 | iperf.its.vanderbilt.edu | 192.111.110.34 | TCP | 20 | | | Yes | -- Select -- | -- Select -- |
| desk172.internet2.edu | 207.75.164.172 | lab253.internet2.edu | 207.75.164.253 | TCP | 20 | | | Yes | -- Select -- | -- Select -- |
| desk172.internet2.edu | 207.75.164.172 | ndt.ScrippsCollege.edu | 134.173.151.207 | TCP | 20 | | | Yes | -- Select -- | -- Select -- |
| desk172.internet2.edu | 207.75.164.172 | perfsonar.its.iastate.edu | 129.186.6.241 | TCP | 20 | | | Yes | -- Select -- | -- Select -- |
| desk172.internet2.edu | 207.75.164.172 | perfsonar.ndsu.NoDak.edu | 134.129.90.1 | TCP | 20 | | | Yes | -- Select -- | -- Select -- |

Non-Active Data Sets

| First Host | First Address | Second Host | Second Address | Protocol | Duration | Window Size | Bandwidth Limit | Bi-Directional | Line Graph | Scatter Graph |
|---|---|---|---|---|---|---|---|---|---|---|

# pS Performance Toolkit – Graphing Results

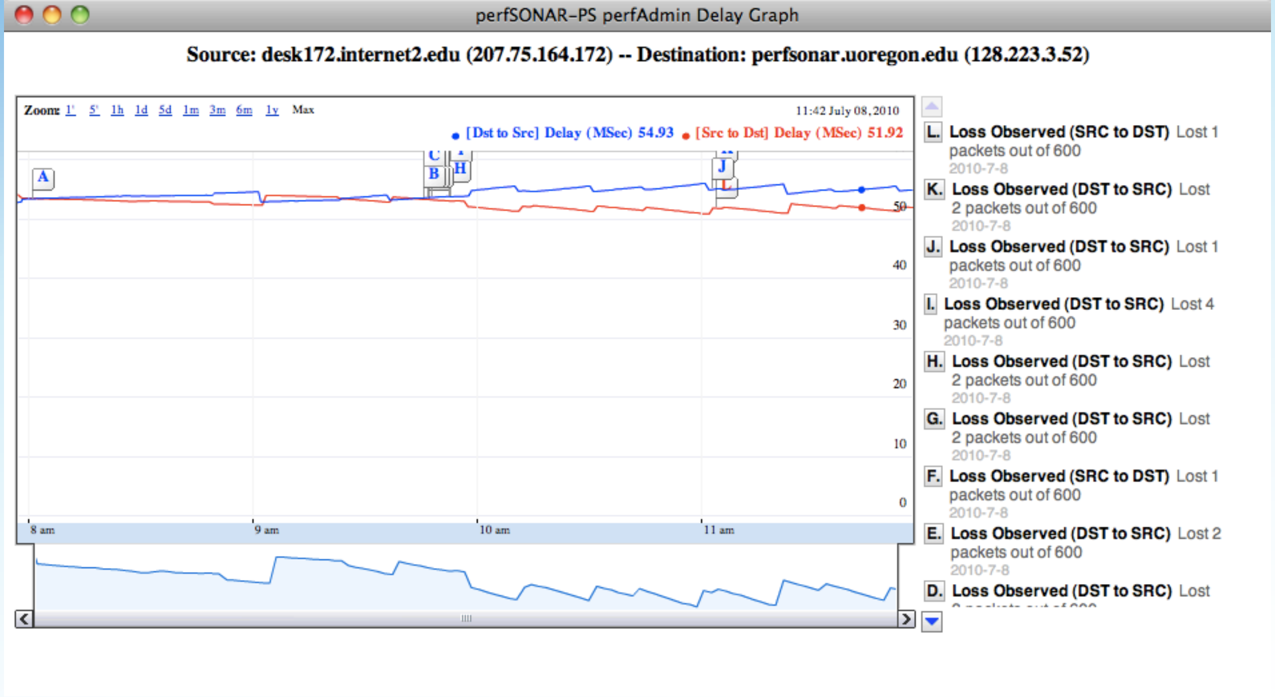


perfSONAR-PS perfAdmin Bandwidth Graph TCP

**Source: bwctl.ucsc.edu (128.114.0.205) -- Destination: desk172.internet2.edu (207.75.164.172)**

| | | | |
|---|---|---|---|
| Maximum **bwctl.ucsc.edu -> desk172.internet2.edu** | 89.20 Mbps | Maximum **desk172.internet2.edu -> bwctl.ucsc.edu** | 71.95 Mbps |
| Average **bwctl.ucsc.edu -> desk172.internet2.edu** | 46.46 Mbps | Average **desk172.internet2.edu -> bwctl.ucsc.edu** | 69.92 Mbps |
| Last **bwctl.ucsc.edu -> desk172.internet2.edu** | 70.83 Mbps | Last **desk172.internet2.edu -> bwctl.ucsc.edu** | 71.75 Mbps |

# pS Performance Toolkit – Graphing Results


perfSONAR-PS perfAdmin Delay Graph
Zoom: 1' 5' 1h 1d 5d 1m 3m 6m 1y Max
11:42 July 08, 2010
● [Dst to Src] Delay (MSec) 54.93
● [Src to Dst] Delay (MSec) 51.92
A
B
C
H
J
L
50
40
30
20
10
0
8 am
9 am
10 am
11 am

Source: desk172.internet2.edu (207.75.164.172) -- Destination: perfsonar.uoregon.edu (128.223.3.52)

L. **Loss Observed (SRC to DST)** Lost 1 packets out of 600
2010-7-8

K. **Loss Observed (DST to SRC)** Lost 2 packets out of 600
2010-7-8

J. **Loss Observed (DST to SRC)** Lost 1 packets out of 600
2010-7-8

I. **Loss Observed (DST to SRC)** Lost 4 packets out of 600
2010-7-8

H. **Loss Observed (DST to SRC)** Lost 2 packets out of 600
2010-7-8

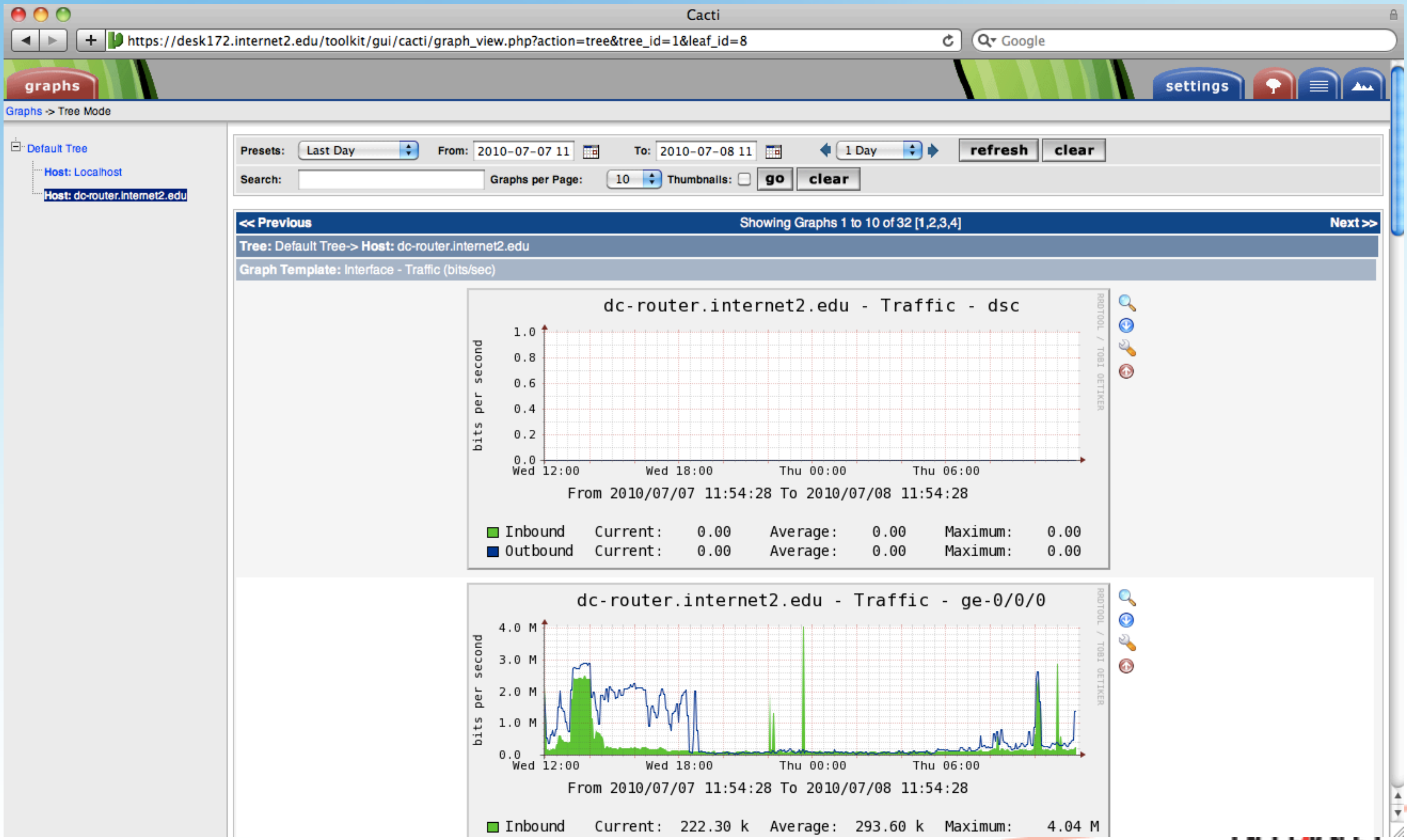
G. **Loss Observed (DST to SRC)** Lost 2 packets out of 600
2010-7-8

F. **Loss Observed (SRC to DST)** Lost 1 packets out of 600
2010-7-8

E. **Loss Observed (DST to SRC)** Lost 2 packets out of 600
2010-7-8

D. **Loss Observed (DST to SRC)** Lost

# pS Performance Toolkit – Cacti Integration



Cacti

https://desk172.internet2.edu/toolkit/gui/cacti/graph_view.php?action=tree&tree_id=1&leaf_id=8

Graphs -> Tree Mode

- Default Tree
  - Host: Localhost
  - Host: dc-router.internet2.edu

Presets: Last Day From: 2010-07-07 11 To: 2010-07-08 11 1 Day refresh clear

Search: Graphs per Page: 10 Thumbnails: □ go clear

<< Previous Showing Graphs 1 to 10 of 32 [1,2,3,4] Next >>

Tree: Default Tree-> Host: dc-router.internet2.edu

Graph Template: Interface - Traffic (bits/sec)

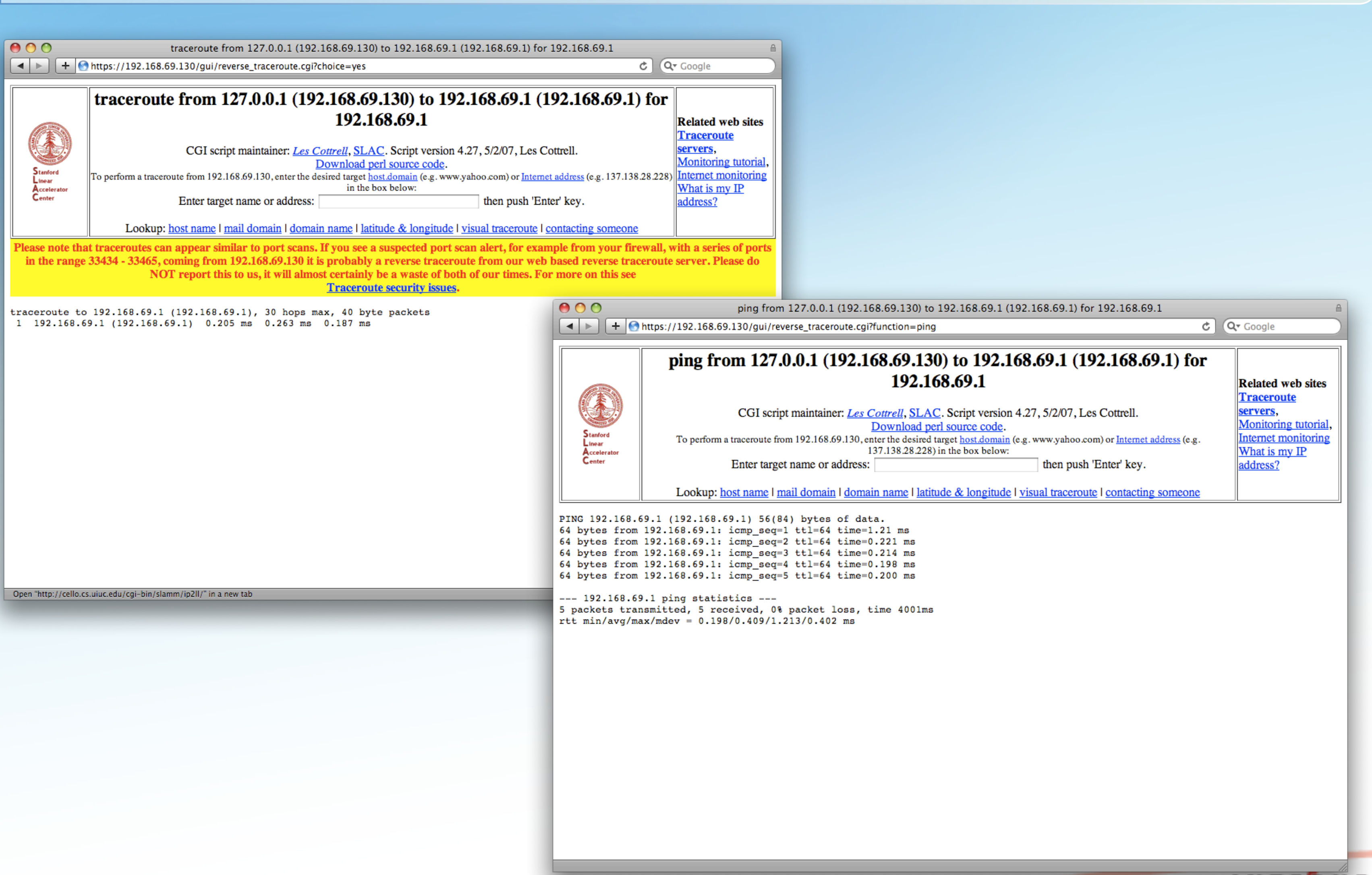
```
From 2010/07/07 11:54:28 To 2010/07/08 11:54:28

🟩 Inbound   Current:   0.00   Average:   0.00   Maximum:   0.00
🟦 Outbound  Current:   0.00   Average:   0.00   Maximum:   0.00
```

```
From 2010/07/07 11:54:28 To 2010/07/08 11:54:28

🟩 Inbound   Current:   222.30 k   Average:   293.60 k   Maximum:    4.04 M
```

# pS Performance Toolkit – Reverse Traceroute



traceroute from 127.0.0.1 (192.168.69.130) to 192.168.69.1 (192.168.69.1) for 192.168.69.1

https://192.168.69.130/gui/reverse_traceroute.cgi?choice=yes

**traceroute from 127.0.0.1 (192.168.69.130) to 192.168.69.1 (192.168.69.1) for 192.168.69.1**

CGI script maintainer: *Les Cottrell*, SLAC. Script version 4.27, 5/2/07, Les Cottrell.
Download perl source code.
To perform a traceroute from 192.168.69.130, enter the desired target host.domain (e.g. www.yahoo.com) or Internet address (e.g. 137.138.28.228) in the box below:
Enter target name or address: then push 'Enter' key.

Lookup: host name | mail domain | domain name | latitude & longitude | visual traceroute | contacting someone

**Related web sites**
**Traceroute servers**,
Monitoring tutorial,
Internet monitoring
What is my IP address?

**Please note that traceroutes can appear similar to port scans. If you see a suspected port scan alert, for example from your firewall, with a series of ports in the range 33434 - 33465, coming from 192.168.69.130 it is probably a reverse traceroute from our web based reverse traceroute server. Please do NOT report this to us, it will almost certainly be a waste of both of our times. For more on this see Traceroute security issues.**

```
traceroute to 192.168.69.1 (192.168.69.1), 30 hops max, 40 byte packets
 1  192.168.69.1 (192.168.69.1)  0.205 ms  0.263 ms  0.187 ms
```

Open "http://cello.cs.uiuc.edu/cgi-bin/slamm/ip2ll/" in a new tab

ping from 127.0.0.1 (192.168.69.130) to 192.168.69.1 (192.168.69.1) for 192.168.69.1

https://192.168.69.130/gui/reverse_traceroute.cgi?function=ping

**ping from 127.0.0.1 (192.168.69.130) to 192.168.69.1 (192.168.69.1) for 192.168.69.1**

CGI script maintainer: *Les Cottrell*, SLAC. Script version 4.27, 5/2/07, Les Cottrell.
Download perl source code.
To perform a traceroute from 192.168.69.130, enter the desired target host.domain (e.g. www.yahoo.com) or Internet address (e.g. 137.138.28.228) in the box below:
Enter target name or address: then push 'Enter' key.

Lookup: host name | mail domain | domain name | latitude & longitude | visual traceroute | contacting someone

**Related web sites**
**Traceroute servers**,
Monitoring tutorial,
Internet monitoring
What is my IP address?

```
PING 192.168.69.1 (192.168.69.1) 56(84) bytes of data.
64 bytes from 192.168.69.1: icmp_seq=1 ttl=64 time=1.21 ms
64 bytes from 192.168.69.1: icmp_seq=2 ttl=64 time=0.221 ms
64 bytes from 192.168.69.1: icmp_seq=3 ttl=64 time=0.214 ms
64 bytes from 192.168.69.1: icmp_seq=4 ttl=64 time=0.198 ms
64 bytes from 192.168.69.1: icmp_seq=5 ttl=64 time=0.200 ms

--- 192.168.69.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4001ms
rtt min/avg/max/mdev = 0.198/0.409/1.213/0.402 ms
```

# Overview

- Introduction and Motivation
- Design
- Current Deployment Footprint
- Conclusion

# Current Deployment Footprint

- *perfSONAR* is gaining traction as an interoperable and extensible monitoring solution
- Adoption has progressed in the following areas:
  - R&E networks including backbone, regional, and exchange points
  - Universities on a national and international basis
  - Federal labs and agencies in the United States (e.g. *JET* nets)
  - Scientific Virtual Organizations, notably the LHC project
- Recent interest has also accrued from:
  - International R&E network partners and exchange points
  - Commercial Providers in the United States
  - Hardware manufactures
- Live pS Status:
  - Services: http://www.perfsonar.net/activeServices
  - Locations: http://www.perfsonar.net/activeServices/IS

# Deployment Footprint - Targets

- Commercial ISPs
  - Success with measurement projects like MLab
  - Would like to see wider adoption of toolkit as a test platform
    - Internal performance testing
    - Testing from the home router to an aggregation point
- For Profit Companies
  - Major content providers
  - Use on VPNs
- R&E/Federal Space
  - Continued adoption for Campuses/Backbones/Labs
  - Monitoring of commercial peering as well as R&E (with the help of the commercial provider)

# Overview

- Introduction and Motivation
- Design
- Current Deployment Footprint
- Conclusion

# Conclusion

- Performance Problems Exist
  - Networks expanding, growing in size and capability
  - Number of users and end devices growing
- Solving Problems Takes Time – Tools also Exist to Help
  - Deployment of a pS Performance Toolkit is simple
  - Availability of Diagnostics and Regular Monitoring
  - Upgrade path = burn a new CD
- Open Source
- Growing Deployments
  - R&E Networks
  - Inroads into Commercial Ventures

# For more information

- General information: http://www.perfsonar.net
- More about what you heard today: http://psps.perfsonar.net/toolkit
- perfSONAR-PS tools and software: http://software.internet2.edu
- A hook to the global lookup service: http://www.perfsonar.net/activeServices/IS/
- More human-readable list of services: http://www.perfsonar.net/activeServices/

# Mailing Lists

- Development (by approval of the project)
  - https://lists.internet2.edu/sympa/subscribe/perfsonar-dev
- User Support
  - https://lists.internet2.edu/sympa/subscribe/perfsonar-ps-users
  - https://lists.internet2.edu/sympa/subscribe/performance-node-users
- Announcements
  - https://lists.internet2.edu/sympa/subscribe/perfsonar-ps-announce
  - https://lists.internet2.edu/sympa/subscribe/performance-node-announce
- Working Groups
  - https://lists.internet2.edu/sympa/subscribe/performance-wg
  - https://lists.internet2.edu/sympa/subscribe/is-wg
  - http://www.ogf.org/mailman/listinfo/nm-wg
  - http://www.ogf.org/mailman/listinfo/nmc-wg
  - http://www.ogf.org/mailman/listinfo/nml-wg

# Thanks!

- Questions or Comments?
- zurawski@internet2.edu

# The pS Performance Toolkit

October 5th 2010, NANOG 50

Jason Zurawski – Internet2

For more information, visit http://psps.perfsonar.net/toolkit