August 10th 2010, OSG Site Admin Workshop - Network Performance

Jason Zurawski, Internet2

Diagnostics vs Regular Monitoring

Agenda

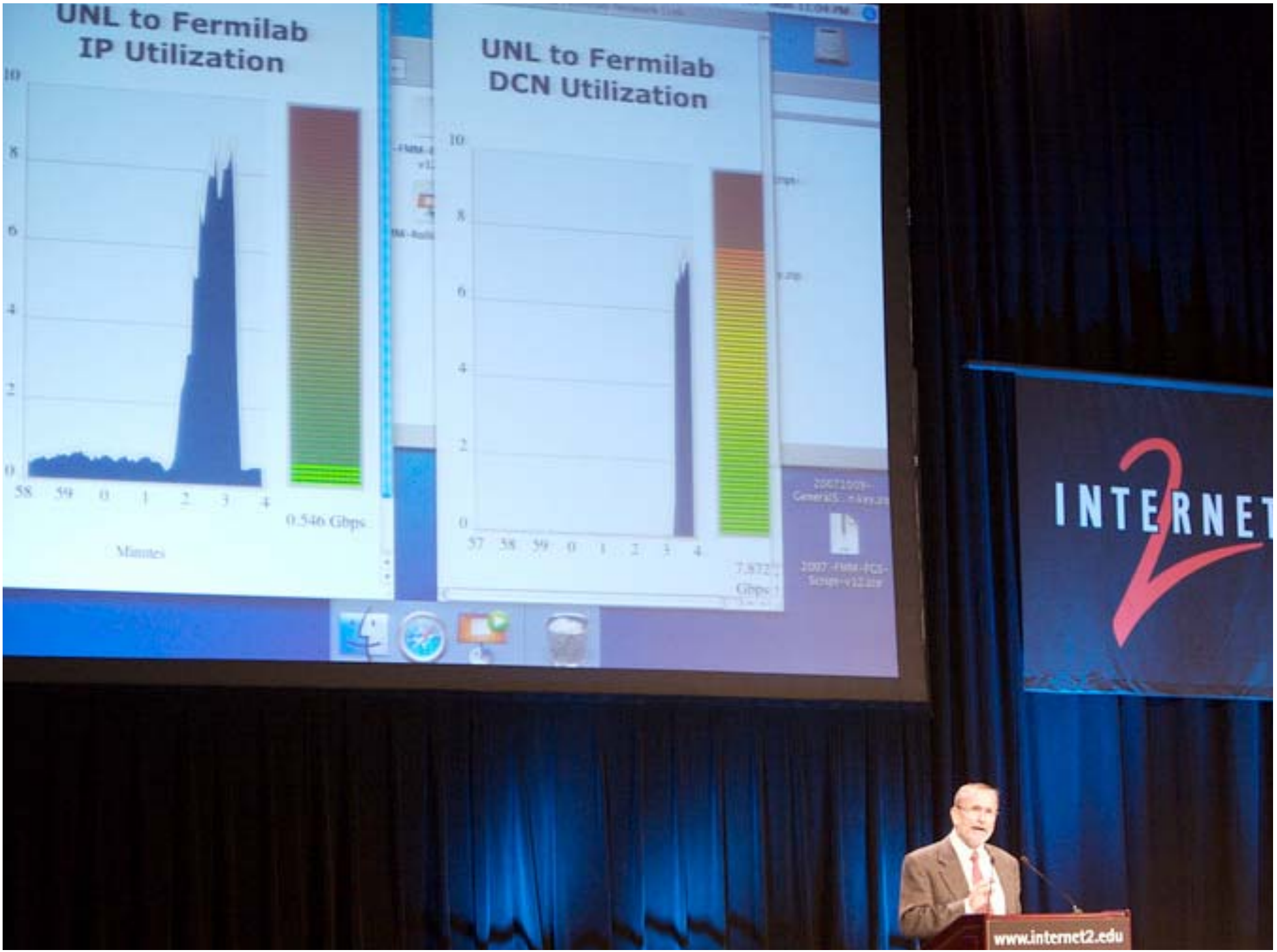
- Tutorial Agenda:
  - Network Performance Primer - Why Should We Care? (**15 Mins**)
  - Getting the Tools (**10 Mins**)
  - Use of the BWCTL Server and Client (**30 Mins**)
  - Use of the OWAMP Server and Client (**30 Mins**)
  - Use of the NDT Server and Client (**30 Mins**)
  - BREAK (**15 mins**)
  - Diagnostics vs Regular Monitoring (**30 Mins**)
  - Network Performance Exercises (**1 hr 30 Mins**)

Performance Monitoring Motivation

- 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

On Demand vs Scheduled Testing

- On-Demand testing can help solve existing problems once they occur
- Regular performance monitoring can quickly identify and locate problems before users complain
  - Alarms
  - Anomaly detection
- Testing and measuring performance increases the value of the network to all participants


UNL to Fermilab
IP Utilization
0.546 Gbps
Minutes
UNL to Fermilab
DCN Utilization
INTERNET
www.internet2.edu

What is perfSONAR?

- Most organizations perform monitoring and diagnostics of their own network
  - SNMP Monitoring via common tools (e.g. MRTG, Cacti)
  - Enterprise monitoring (e.g. Nagios)
- Networking is increasingly a cross-domain effort
  - International collaborations in many spaces (e.g. science, the arts and humanities) are common
  - Interest in development and use of R&E networks at an all time high
- Monitoring and diagnostics ***must*** become a cross-domain effort
  - Complete view of all paths
  - Eliminate "who to contact" and "what to ask for" - 24/7 availability of diagnostic observations

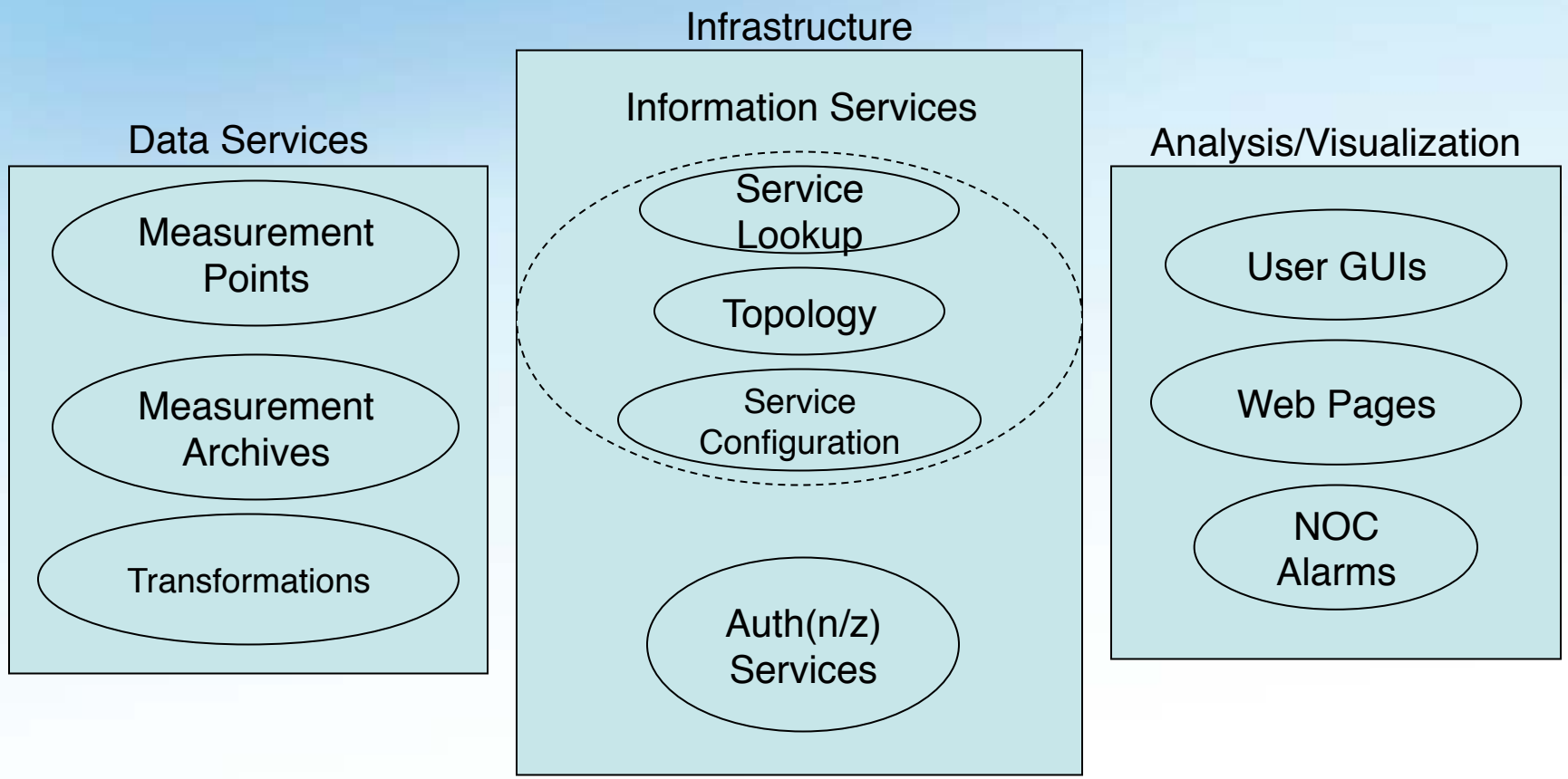
What is perfSONAR?

- 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 Architecture Overview

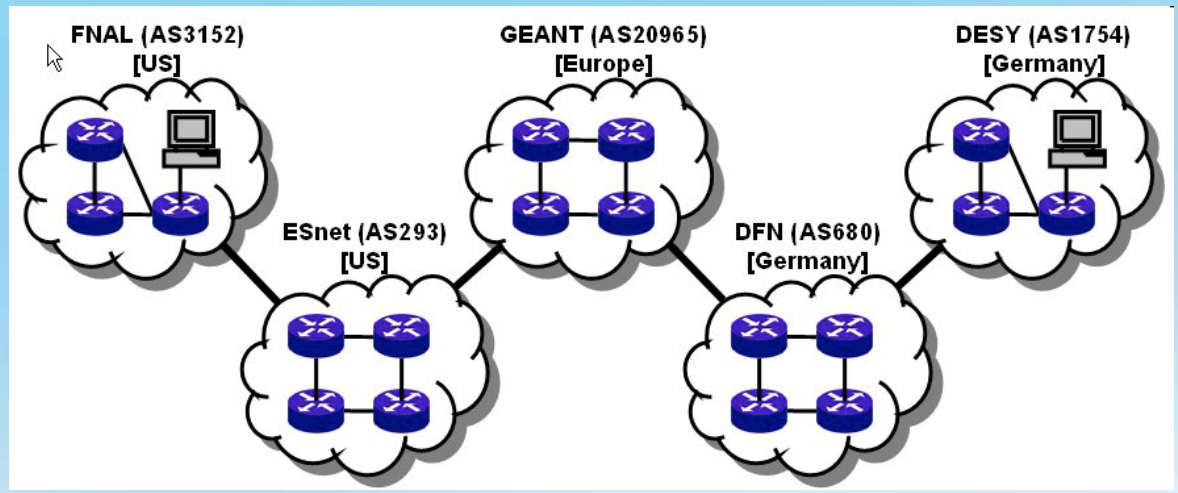
- Interoperable network measurement middleware designed as a Service Oriented Architecture (SOA):
  - Each component is modular
  - All are Web Services (WS) based
  - The global *perfSONAR* framework as well as individual deployments are decentralized
  - All *perfSONAR* tools are Locally controlled
  - 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

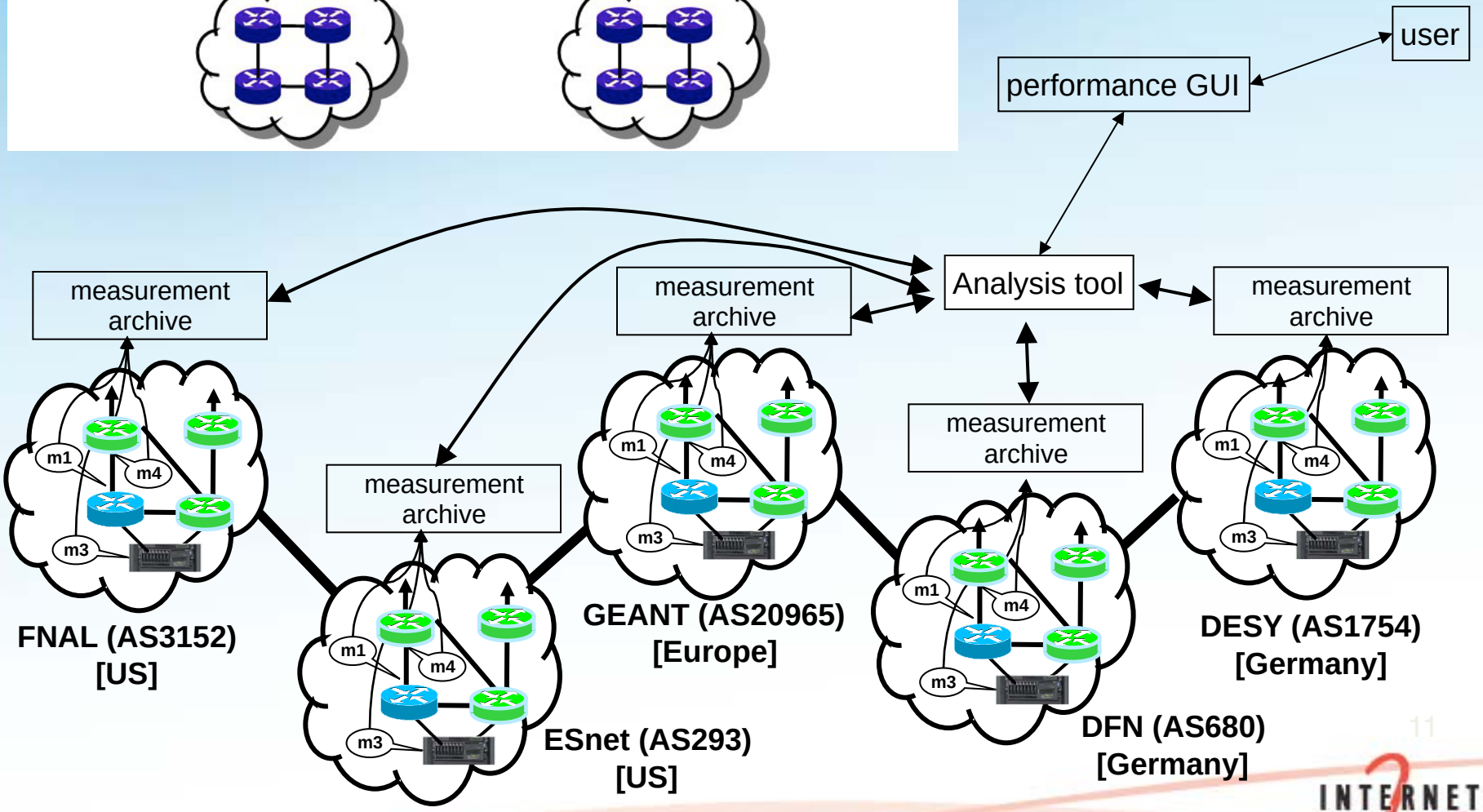


perfSONAR Architecture Overview

- A *perfSONAR* deployment can be any combination of services
  - An instance of the *Lookup Service* is required to share information
  - Any combination of data services and analysis and visualization tools is possible
- *perfSONAR* services have the ability to federate globally
  - The *Lookup Service* communicates with a confederated group of directory services (e.g. the *Global Lookup Service*)
  - Global discovery is possible through APIs
- *perfSONAR* is most effective ***when all paths are monitored***
  - Debugging network performance must be done *end-to-end*
  - Lack of information for specific domains can delay or hinder the debug process



Many collaborations are inherently multi-domain, so for an end-to-end monitoring tool to work everyone must participate in the monitoring infrastructure



Who is perfSONAR?

- The *perfSONAR* Consortium is a joint collaboration between
  - ESnet
  - Géant
  - Internet2
  - Rede Nacional de Ensino e Pesquisa (RNP)
- Decisions regarding protocol development, software branding, and interoperability are handled at this organization level
- There are at least two independent efforts to develop software frameworks that are *perfSONAR* compatible.
  - perfSONAR-MDM
  - perfSONAR-PS
  - Others? The beauty of open source software is we will never know the full extent!
- Each project works on an individual development roadmap and works with the consortium to further protocol development and insure compatibility

Who is perfSONAR-PS?

- perfSONAR-PS is comprised of several members:
  - ESnet
  - Fermilab
  - Georgia Tech
  - Indiana University
  - Internet2
  - SLAC
  - The University of Delaware
- perfSONAR-PS products are written in the perl programming language and are available for installation via source or RPM (Red Hat Compatible) packages
- perfSONAR-PS is also a major component of the pS Performance Toolkit – A bootable Linux CD containing measurement tools.

perfSONAR-PS Availability

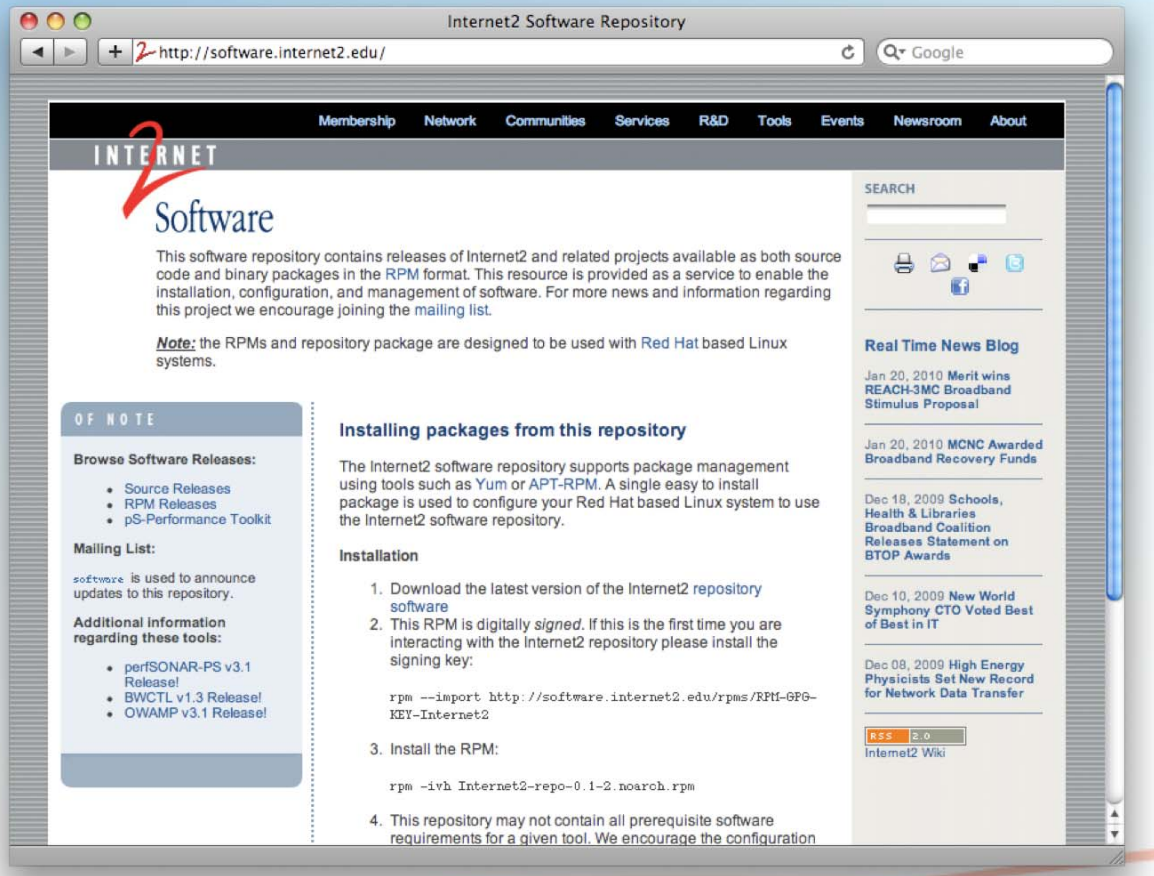
- *perfSONAR-PS* is an implementation of the *perfSONAR* measurement infrastructure and protocols written in the *perl* programming language
- All products are available as platform and architecture independent source code.
- All products are available as **RPM**s (e.g. *RPM Package Manager*). The *perfSONAR-PS* consortium directly supports the following operating systems:
  - **CentOS** (version 5)
- RPMs are compiled for the x86 (should work w/ x86 64 bit) architecture.
- Functionality on other platforms and architectures is possible, but not supported. Attempts are done at the user's own risk.
  - Should work:
    - **Scientific Linux** (versions 4 and 5)
    - **Red Hat Enterprise Linux** (versions 4 and 5)
  - Harder, but possible:
    - **Fedora Linux** (any recent version)
    - **SuSE** (any recent version)
    - **Debian Variants** (…)

perfSONAR-PS Availability

- The pS Performance Toolkit (*pSPT*) is a Linux ISO image (e.g. a *LiveCD*) packed by Internet2 for both easy of installation and configuration of performance tools
  - Prior:
    - Based on Knoppix Linux
  - Current:
    - Based on CentOS (version 5) Linux
  - Designed for x86 architecture
  - No explicit support for x86 64 bit but compatibility is expected
- Product also contains other relevant measurement tools and *perfSONAR-PS* dependencies.
- Support structure is limited to the following goals:
  - Updated versions of all software (operating system and performance) with each release
  - Monitoring and alerts regarding critical security vulnerabilities for all software. Critical patches and releases available for severe cases
  - Semi annual (4 times per year) minor releases

perfSONAR-PS Availability

- *perfSONAR-PS* and the *pSPT* are available from http://software.internet2.edu



perfSONAR-PS Availability

- To facilitate installation and updates on the supported platforms, installation is available through several package managers:
  - YUM
  - Up2date
  - APT-RPM
- Instructions to enable are available on http://software.internet2.edu
- Installing software becomes a simple one step operation
  - Dependencies are managed by the operating system
  - Software is identified by name, and can be searched for

perfSONAR-PS Availability

- Using YUM to search for packages:

```
[zurawski@clean-centos5 ~]$ sudo yum search perfSONAR
perl-perfSONAR_PS-LookupService.noarch : perfSONAR_PS Lookup Service
perl-perfSONAR_PS-TopologyService.noarch : perfSONAR_PS Topology Service
perl-perfSONAR_PS-Status.noarch : perfSONAR-PS Status Service
perl-perfSONAR_PS-PingER-server.noarch : perfSONAR_PS PingER  Measurement Archiv
e and Collection System
perl-perfSONAR_PS-perfAdmin.noarch : perfSONAR_PS perfAdmin
perl-perfSONAR_PS-perfSONARBUOY-client.noarch : perfSONAR_PS perfSONARBUOY Web S
ervice Client and Measurement System
perl-perfSONAR_PS-LSRegistrationDaemon.noarch : perfSONAR_PS Lookup Service Regi
stration Daemon
perl-perfSONAR_PS-perfSONARBUOY-server.noarch : perfSONAR_PS perfSONARBUOY Measu
rement Archive and Collection System
perl-perfSONAR_PS-perfSONARBUOY-config.noarch : perfSONAR_PS perfSONARBUOY Confi
guration Information
perl-perfSONAR_PS-PingER-GUI.i386 : perfSONAR_PS PingER  data charts GUI
perl-perfSONAR_PS-SNMPMA.noarch : perfSONAR_PS SNMP Measurement Archive
```

perfSONAR-PS Availability

- Using YUM to install packages:

```
[zurawski@clean-centos5 ~]$ sudo yum install owamp-client
Setting up Install Process
Parsing package install arguments
Resolving Dependencies
--> Running transaction check
---> Package owamp-client.i386 0:3.2rc1-1 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                Arch     Version          Repository          Size
================================================================================
Installing:
 owamp-client           i386     3.2rc1-1         Internet2           198 k

Transaction Summary
================================================================================
Install       1 Package(s)
Update        0 Package(s)
Remove        0 Package(s)

Total download size: 198 k
Is this ok [y/N]:
```

perfSONAR-PS Availability

- perfSONAR-PS is working to build a strong user community to support the use and development of the software.
- perfSONAR-PS Mailing Lists
  - Users List: https://mail.internet2.edu/wws/subrequest/perfsonar-ps-users
  - Announcement List: https://mail.internet2.edu/wws/subrequest/perfsonar-ps-announce
- pSPT Mailing Lists
  - Users List: https://mail.internet2.edu/wws/subrequest/performance-node-users
  - Announcement List: https://mail.internet2.edu/wws/subrequest/performance-node-announce

perfSONAR Adoption

- *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

Regular Monitoring Motivation

- Now that we have seen the purpose and makeup of the *perfSONAR* infrastructure, it's time to see what it can do in the real world
- *perfSONAR* is used by network engineers to identify many types of performance problem
  - A ***Divide and Conquer*** strategy is necessary to isolate problems
  - A ***structured methodology*** helps to eliminate duplicate or useless steps
  - *perfSONAR* works best when everyone participates, holes in deployment lead to holes in the problem solving phase
- The following sections will outline the proper deployment strategy and describe some real work use cases

How it ***Should*** Work

- To accurately and swiftly address network performance problems the following steps should be undertaken
  - Identify the problem: if there a user in one location is complaining about performance to another, get as much information as possible
    - Is the problem un-directional? Bi-directional?
    - Does the problem occur all the time, frequently, or rarely?
    - Does the problem occur for only a specific application, many applications, or only some applications?
    - Is the problem reproducible on other machines?
  - Gather information about the environment
    - Hosts
    - Network Path
    - Configuration (where applicable)
    - Resources available

How it ***Should*** Work

- Cont.
  - Methodically approach the problem
    - Test using the same tool everywhere, gather results
    - Before moving on to the next tool, did you gather everything of value?
    - Are the results consistent?
  - After proceeding through all tools and approaches, form theories
    - Can the problem be isolated to a specific resource or component?
    - Can testing be performed to eliminate dead ends?
- Consider the following example:
  - International path
  - Problems noted
  - We know the path
  - We have tools available

Scenario: Multi-domain International Path

Desirable Case: Expected Performance

Typical: Poor Performance … Somewhere

Typical: Poor Performance … Somewhere

Solution: Test Points + Regular Monitoring

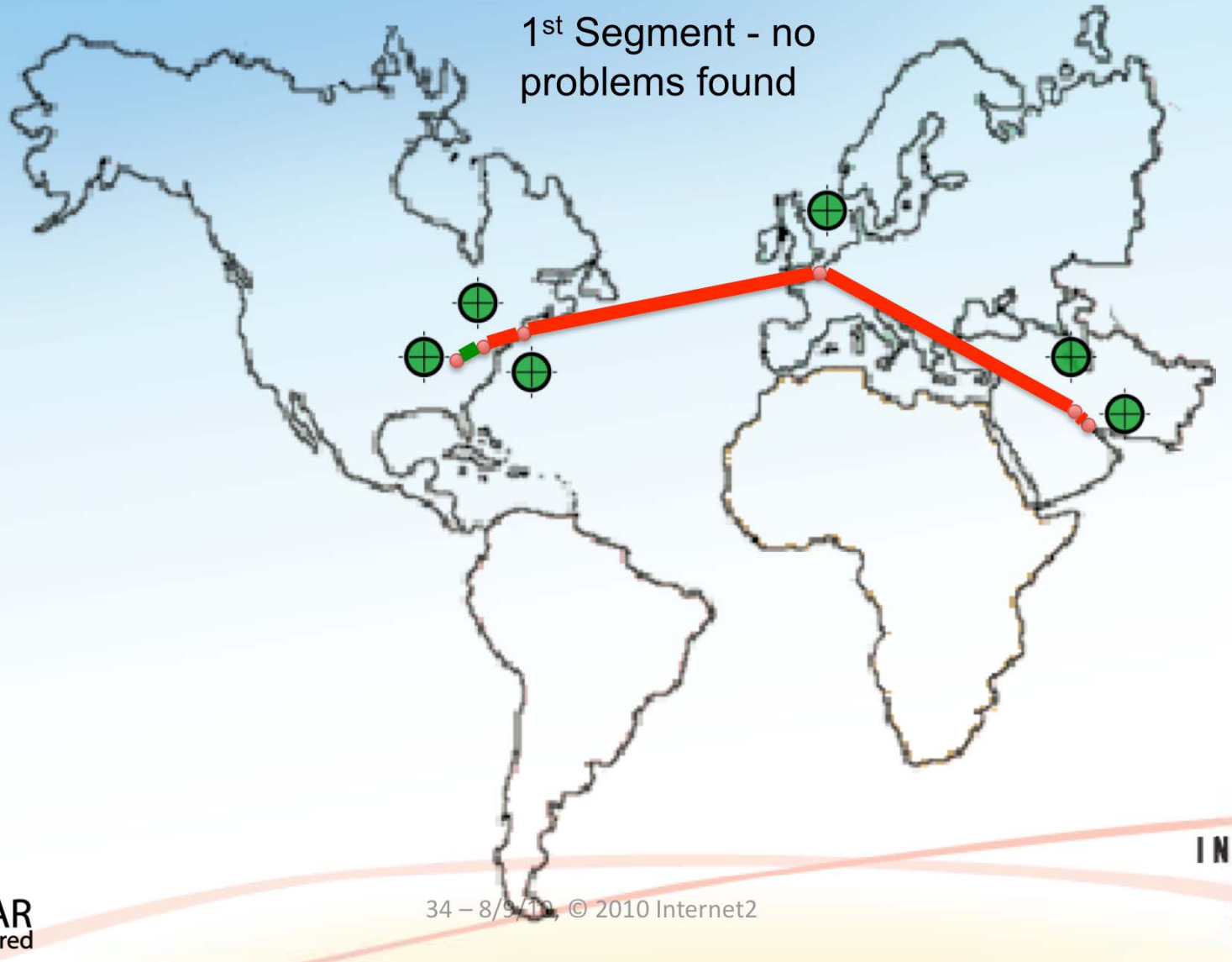
perfSONAR: Backbone and Exchanges

perfSONAR: Regional Networks

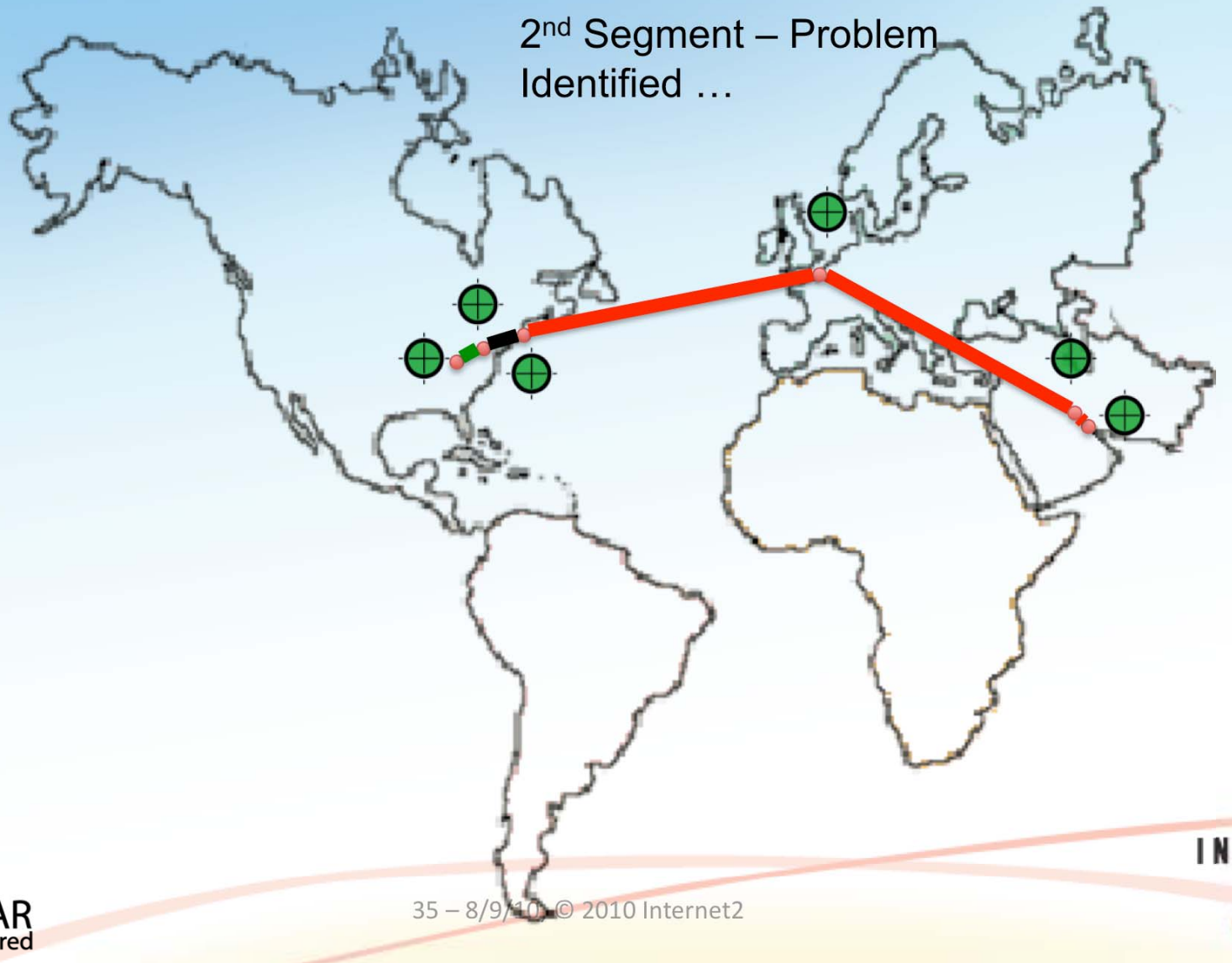
perfSONAR: Campus

Path Decomposition – Isolate the Problem

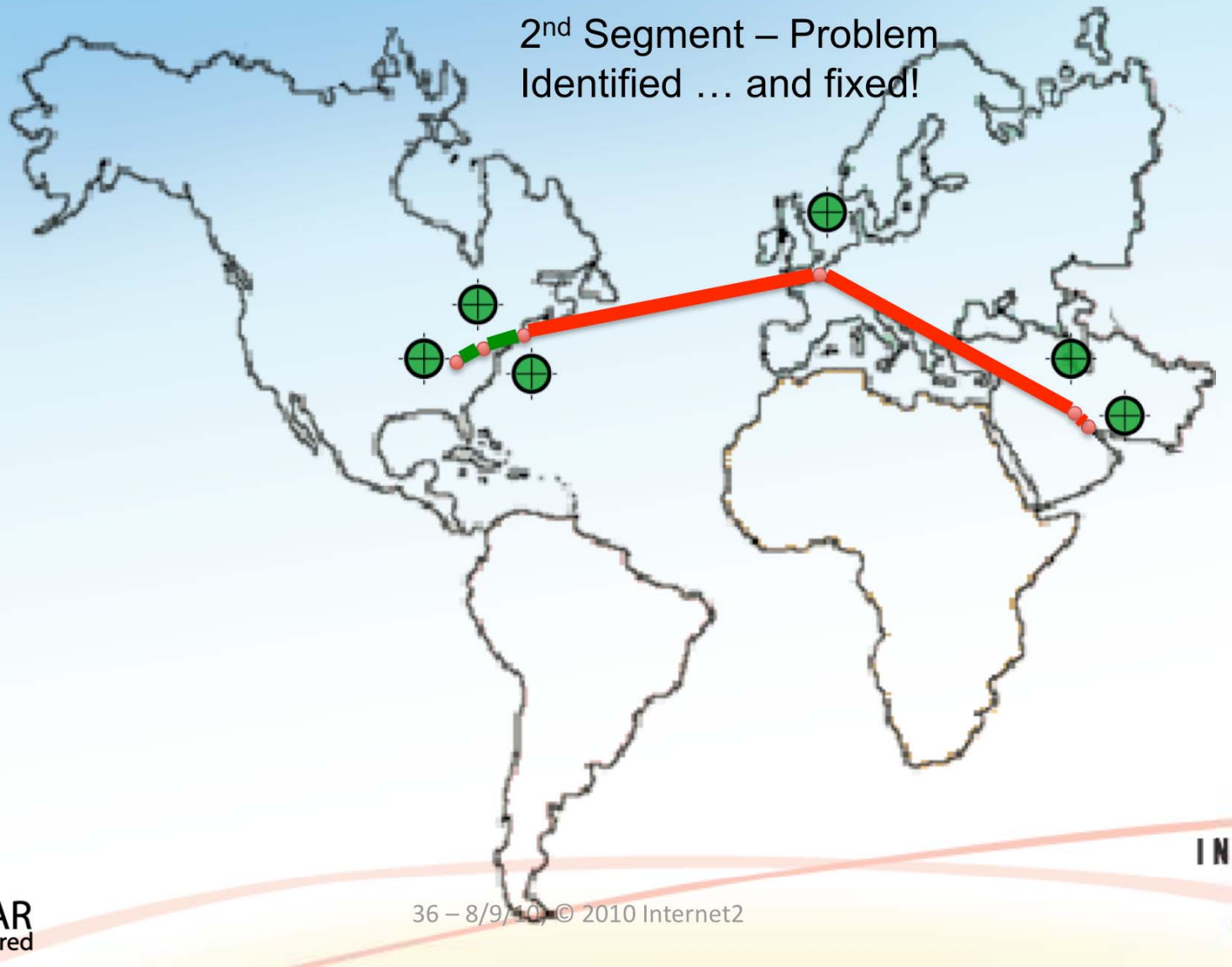
Path Decomposition – Isolate the Problem



Path Decomposition – Isolate the Problem

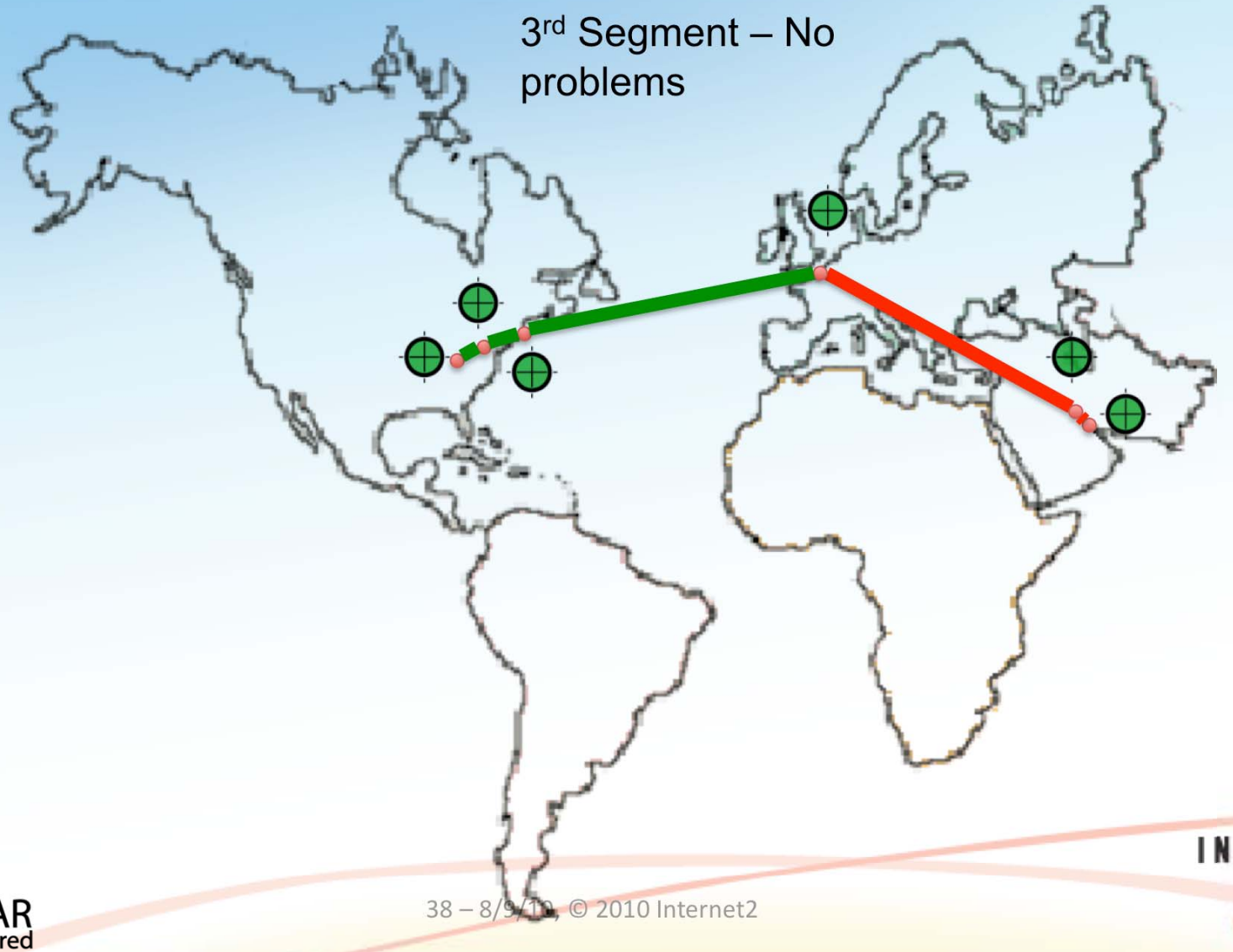


Path Decomposition – Isolate the Problem

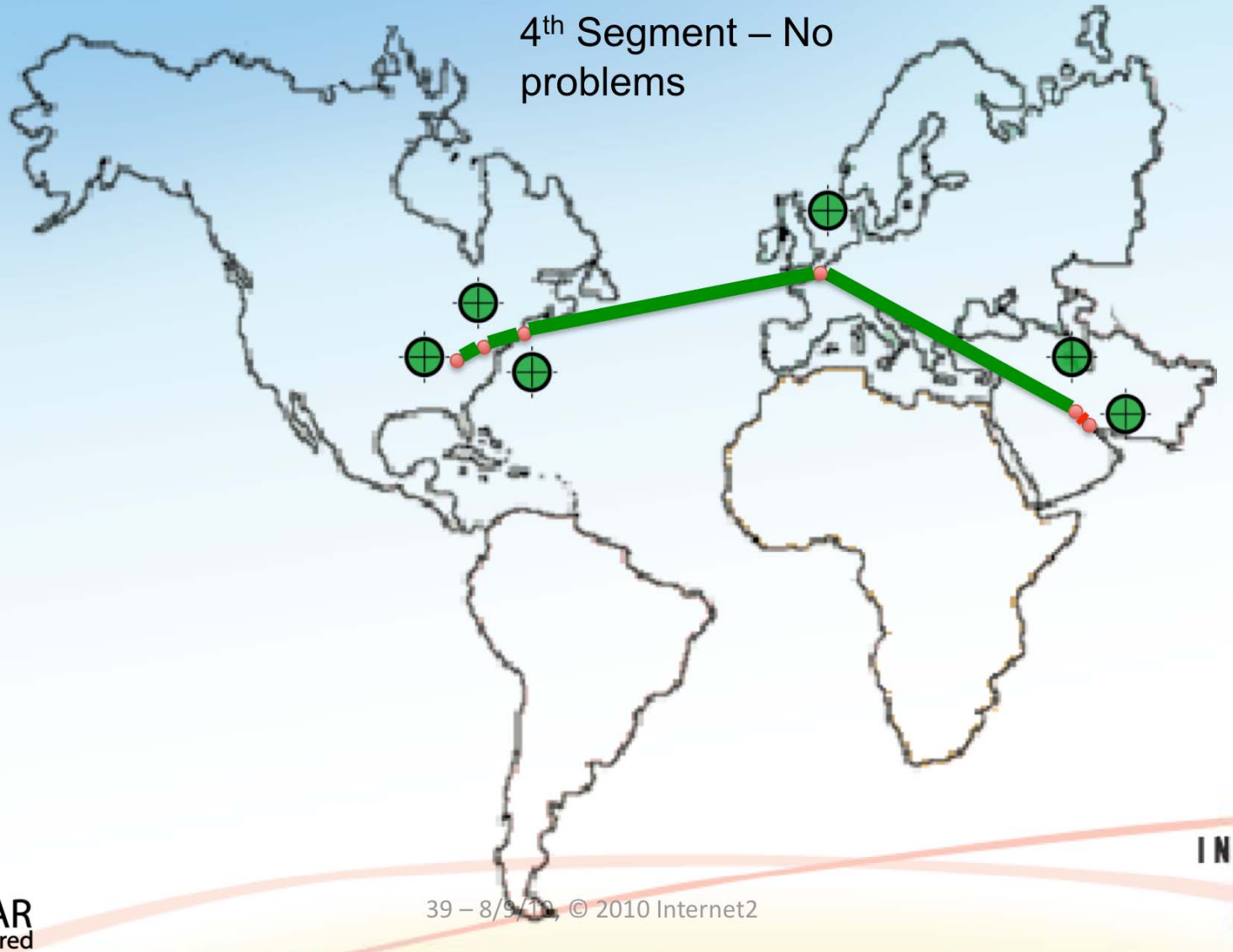


Path Decomposition – Isolate the Problem

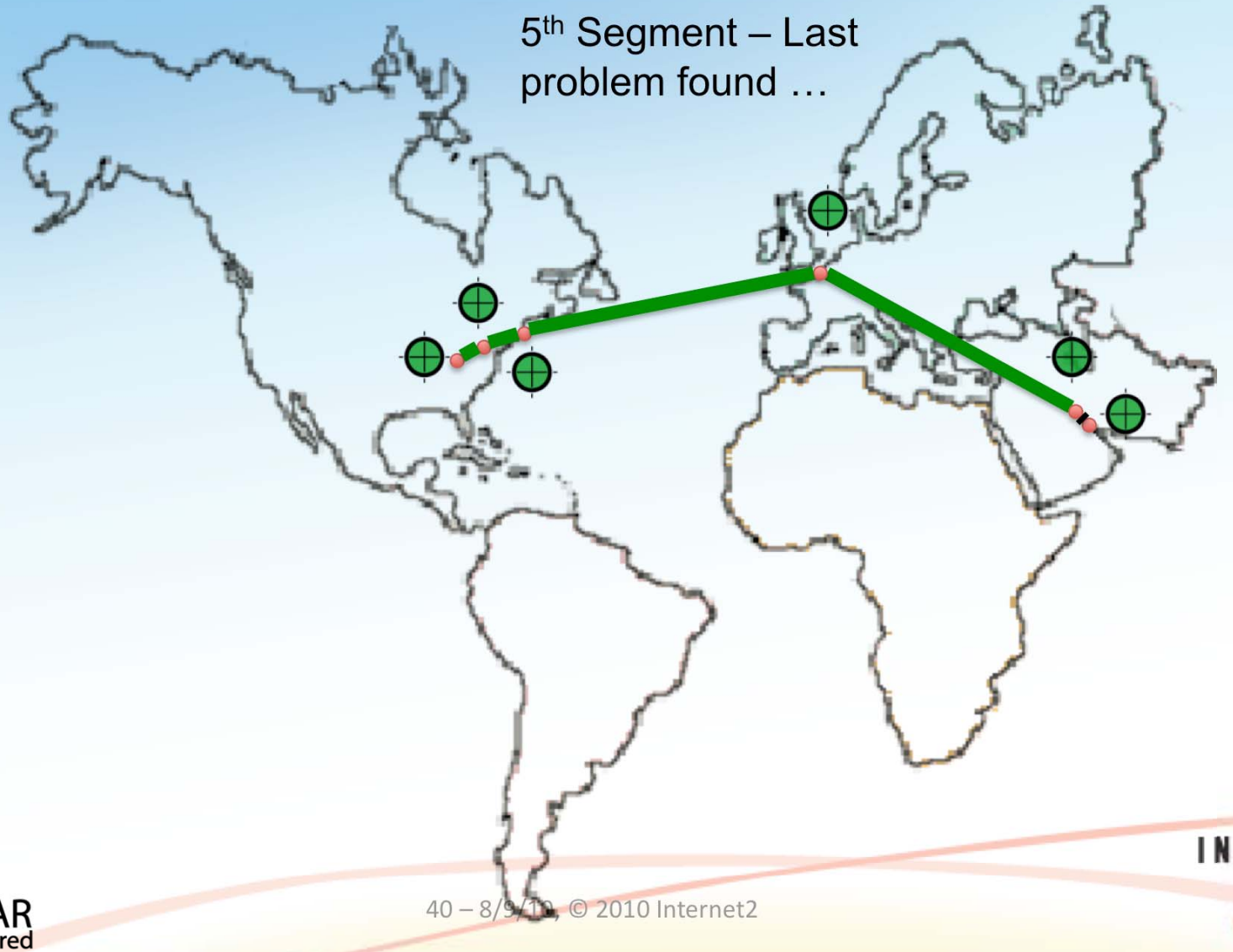
Path Decomposition – Isolate the Problem



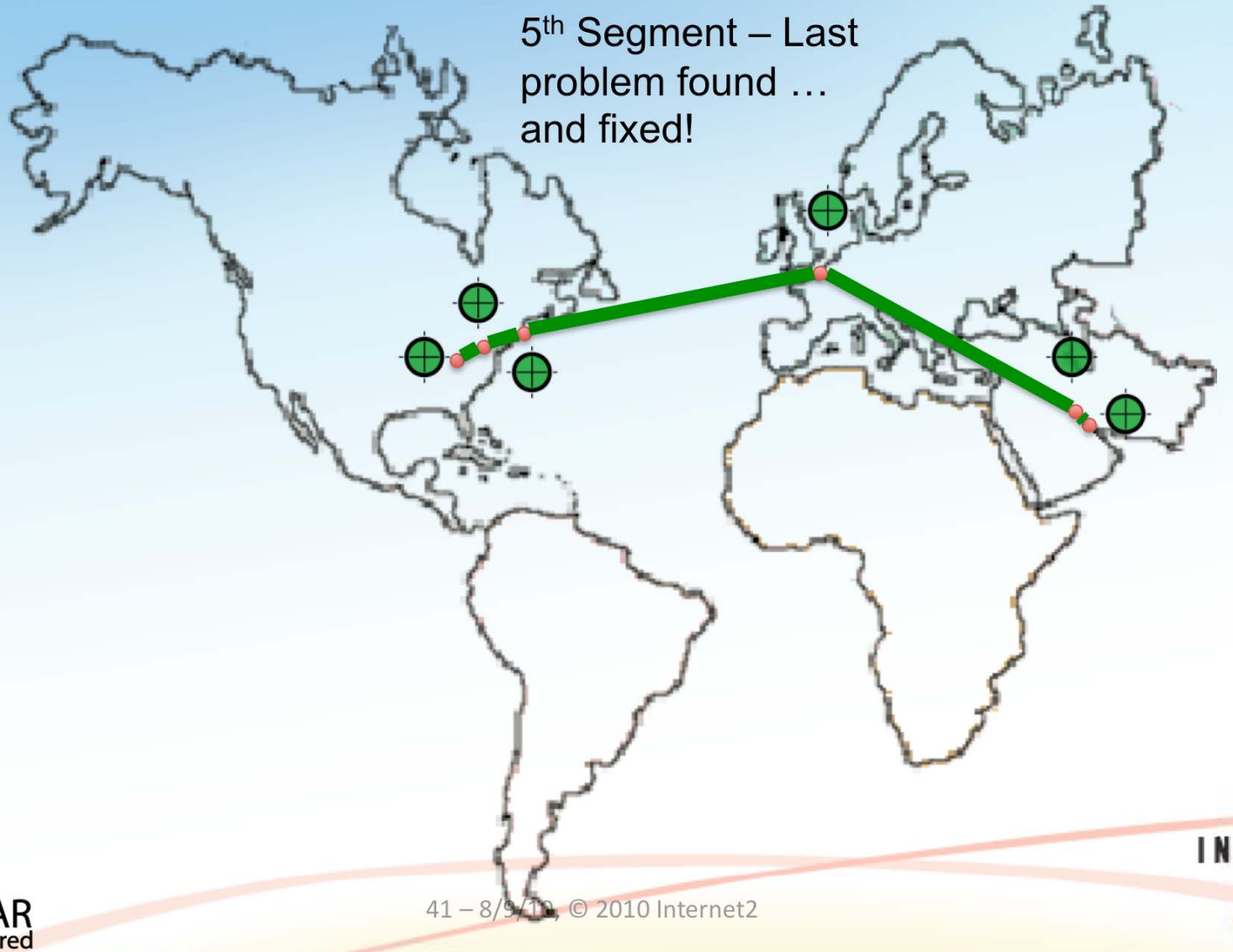
Path Decomposition – Isolate the Problem



Path Decomposition – Isolate the Problem



Path Decomposition – Isolate the Problem



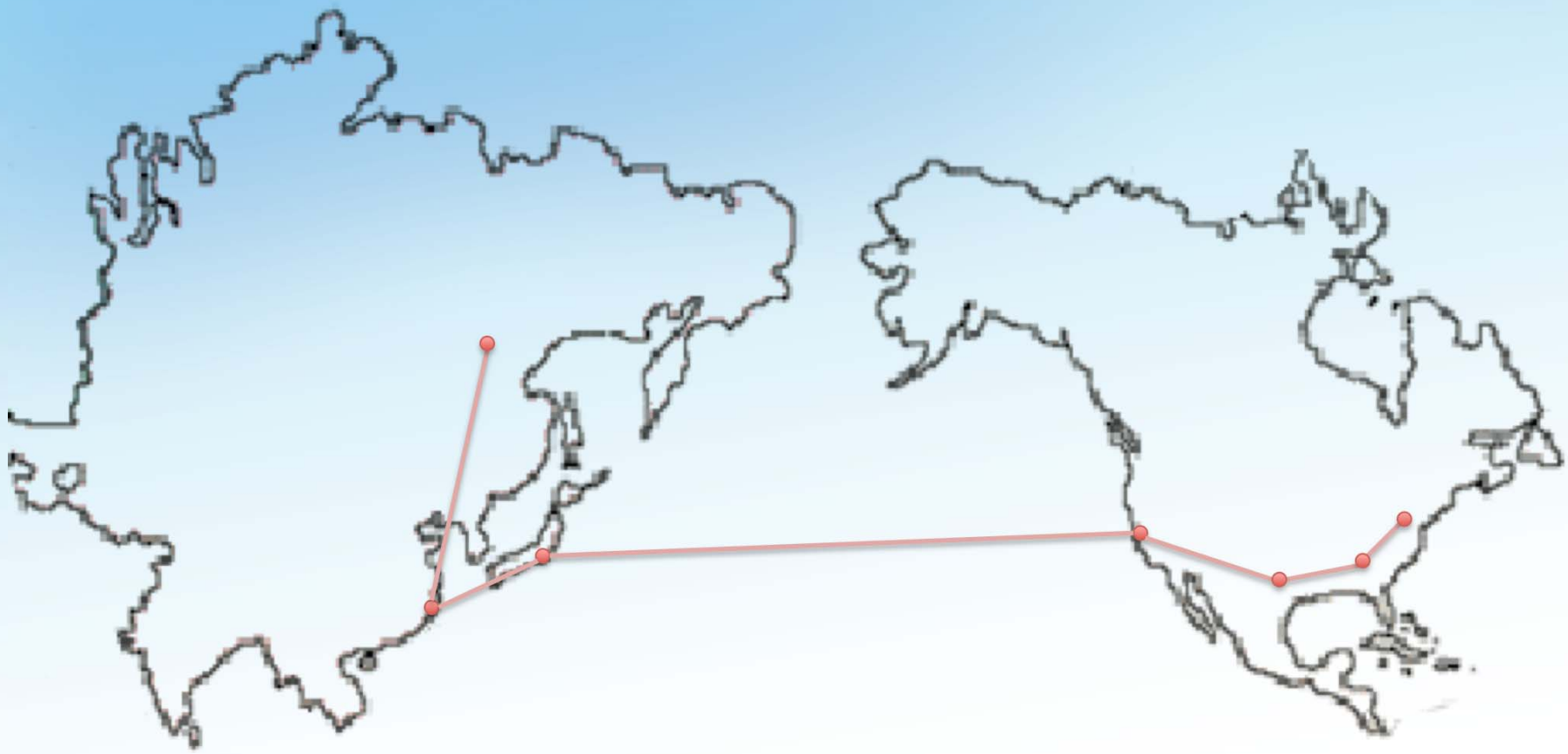
Lessons Learned

- Problem resolution requires proper tools
  - Specialized to given task (e.g. Bandwidth, Latency)
  - Widely available where the problems will be
- Isolating a problem is a well defined, multi-step process
  - Rigid set of steps – systematic approach to prevent causing new problems
- Diagnostics, as well as regular monitoring, can reveal true network performance

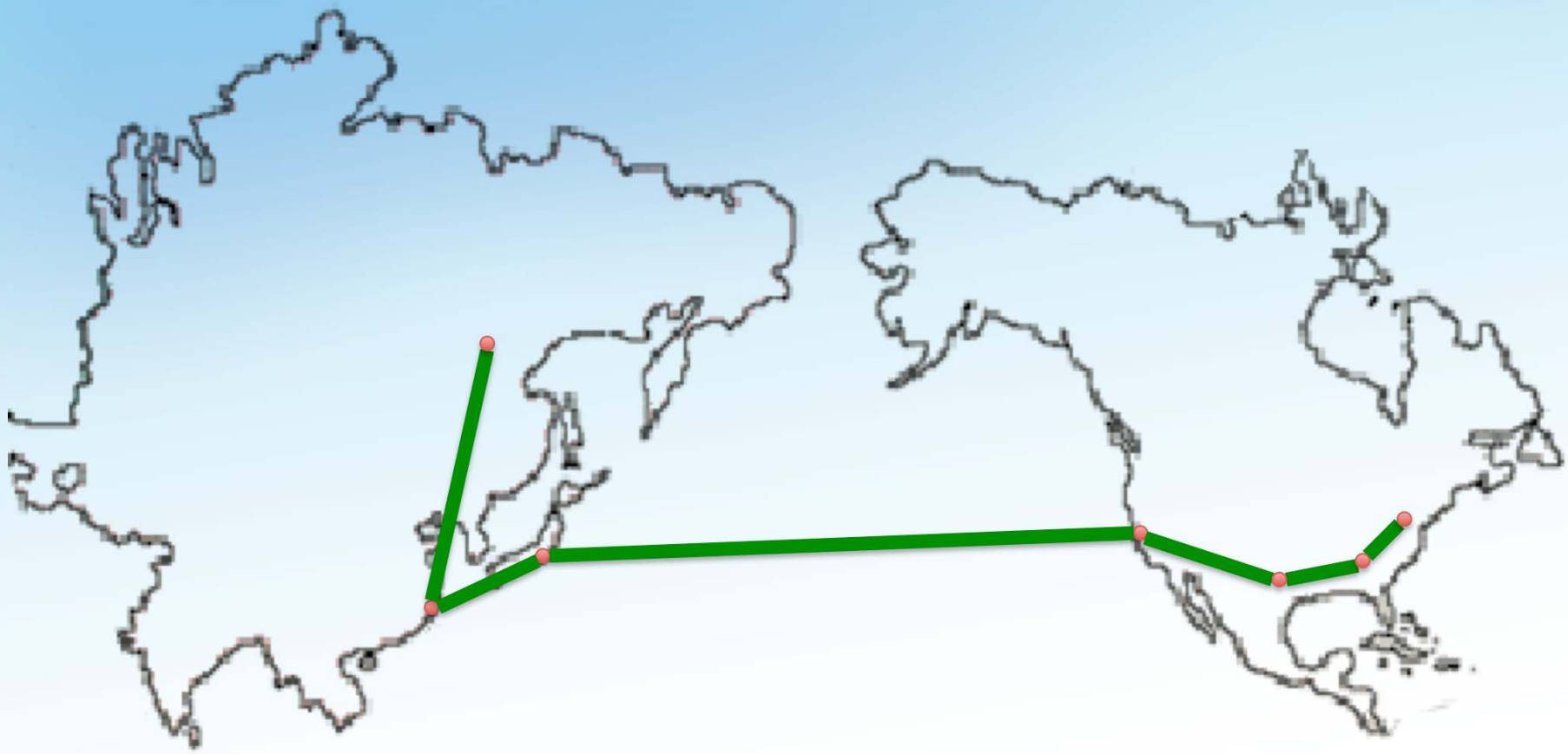
How it ***Probably*** Works

- If the suggested steps aren't taken (or followed in an ad-hoc manner), results will vary.
  - Skipping steps leads to missing clues
- Deployment and participation may vary, this leads to some gaps in the debugging process
- Consider the following example:
  - International path
  - Problems noted
  - We know the path
  - We have tools available - almost everywhere

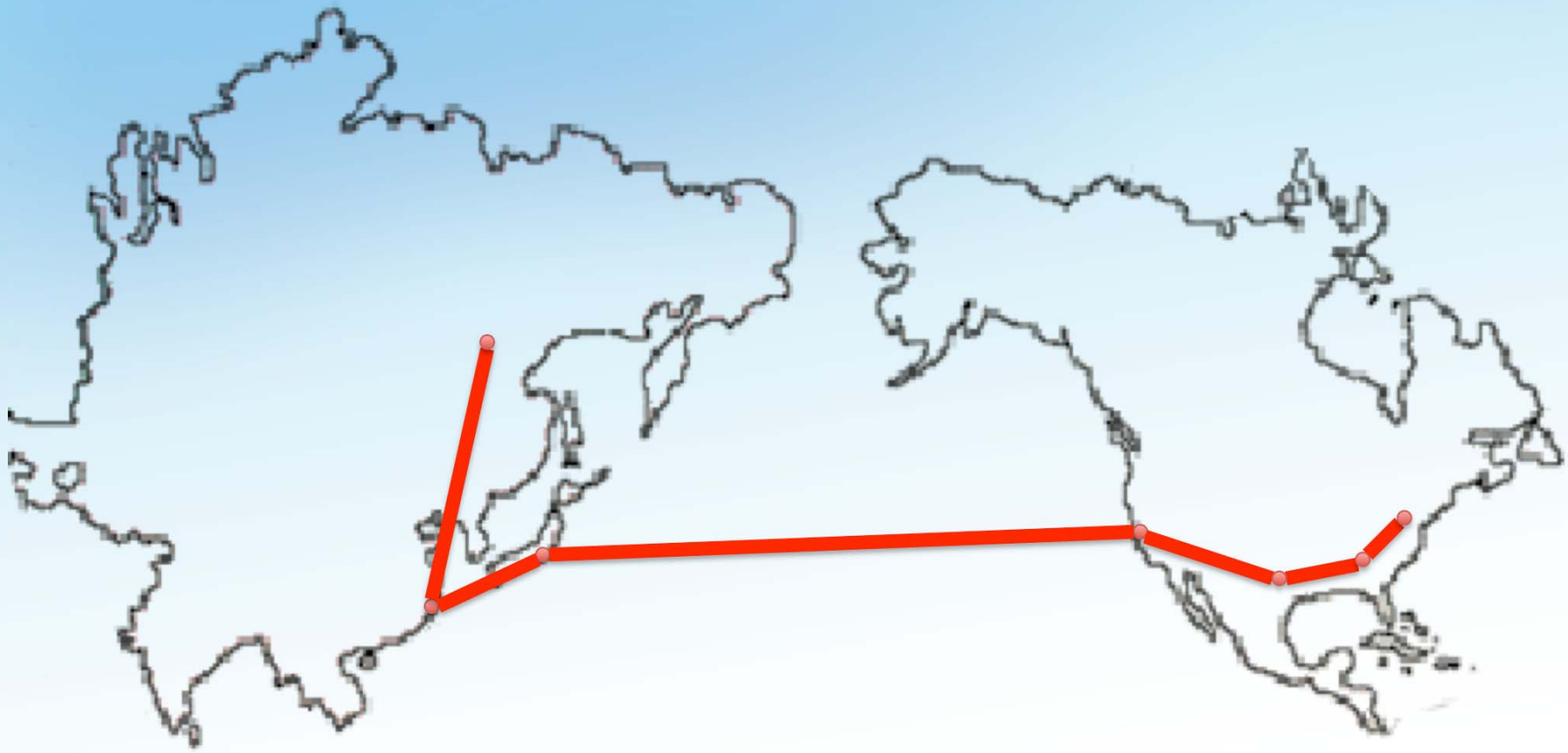
Scenario: Multi-domain International Path



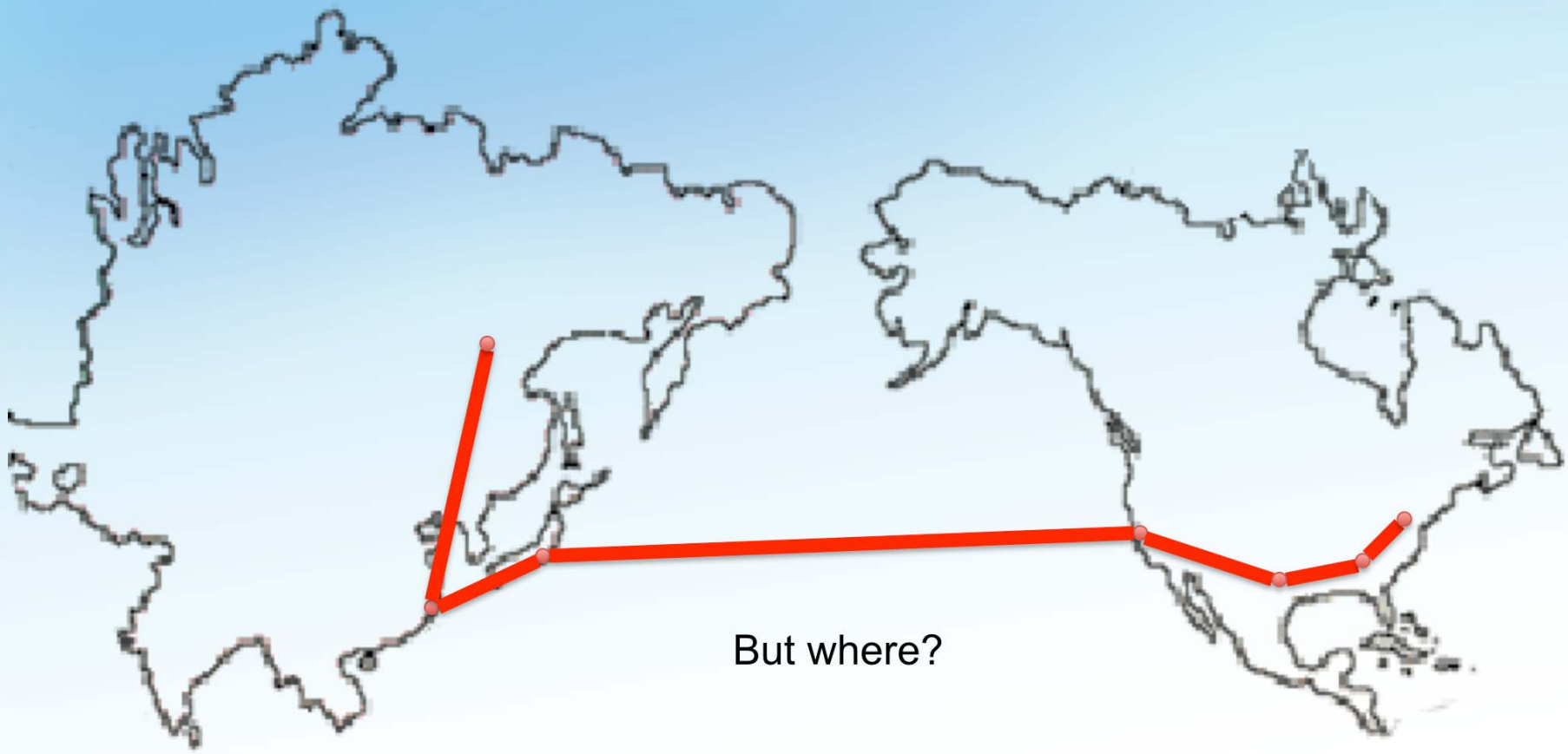
Desirable Case: Expected Performance



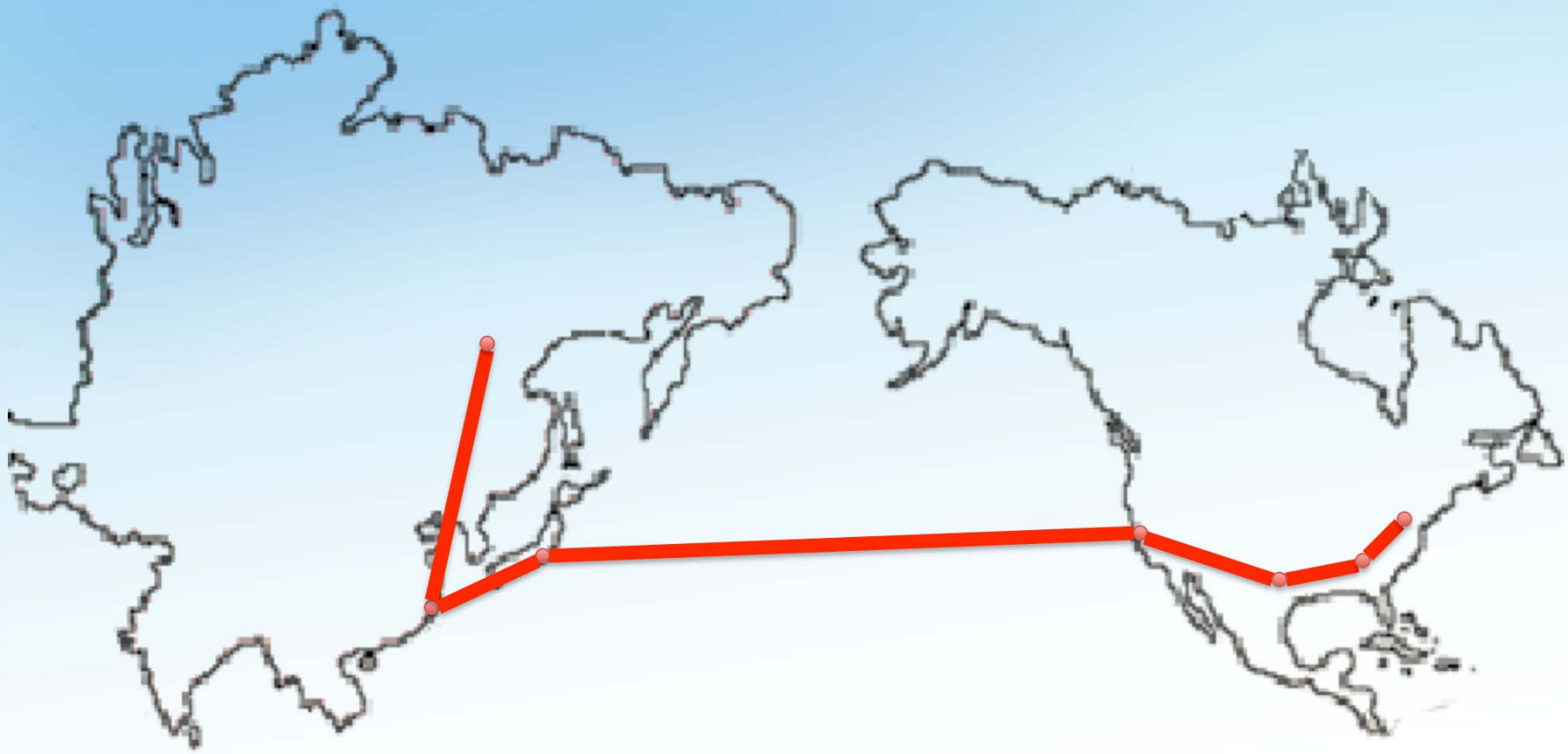
Typical: Poor Performance … Somewhere



Typical: Poor Performance … Somewhere



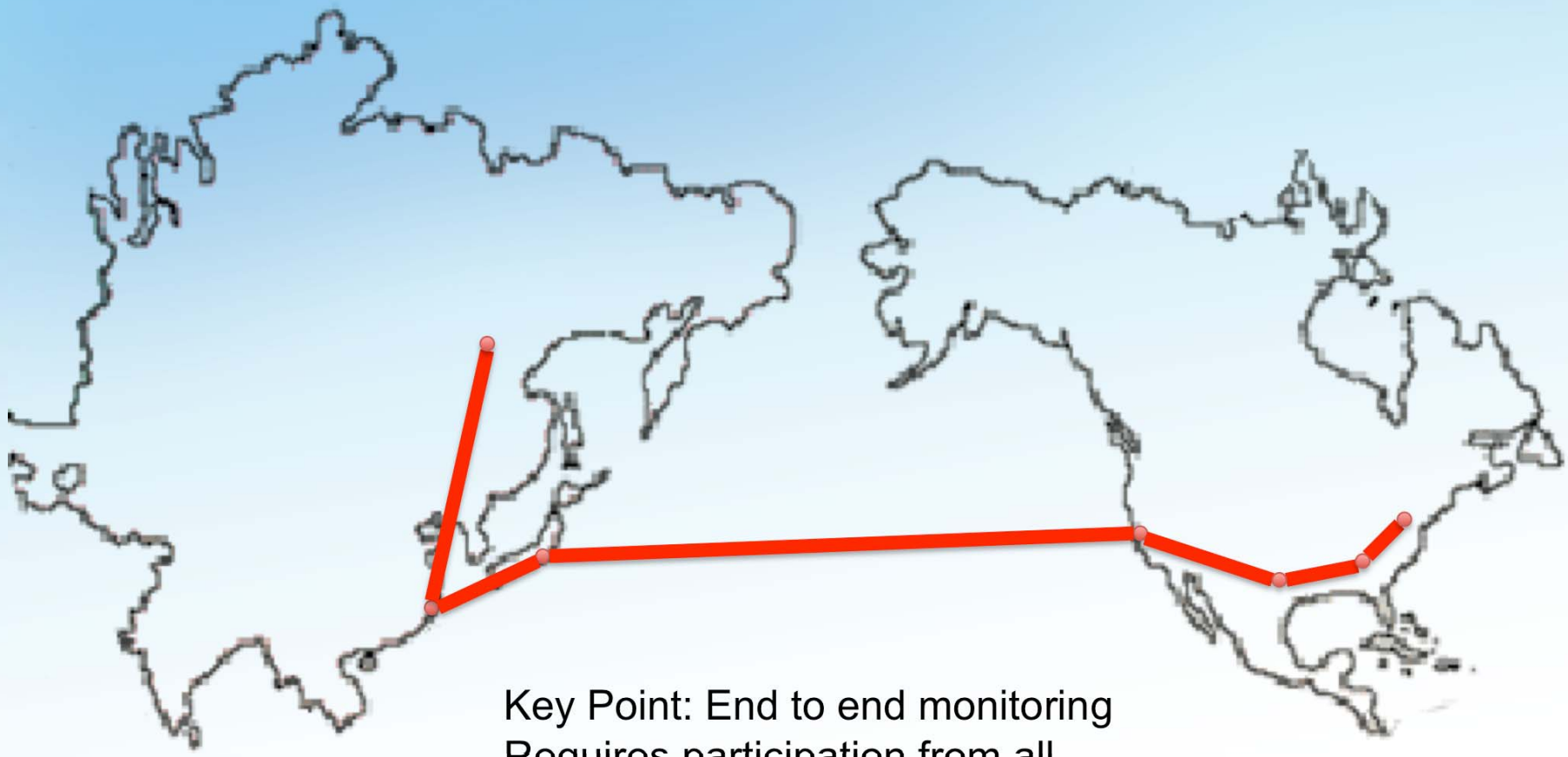
Solution: Test Points + Regular Monitoring



perfSONAR powered

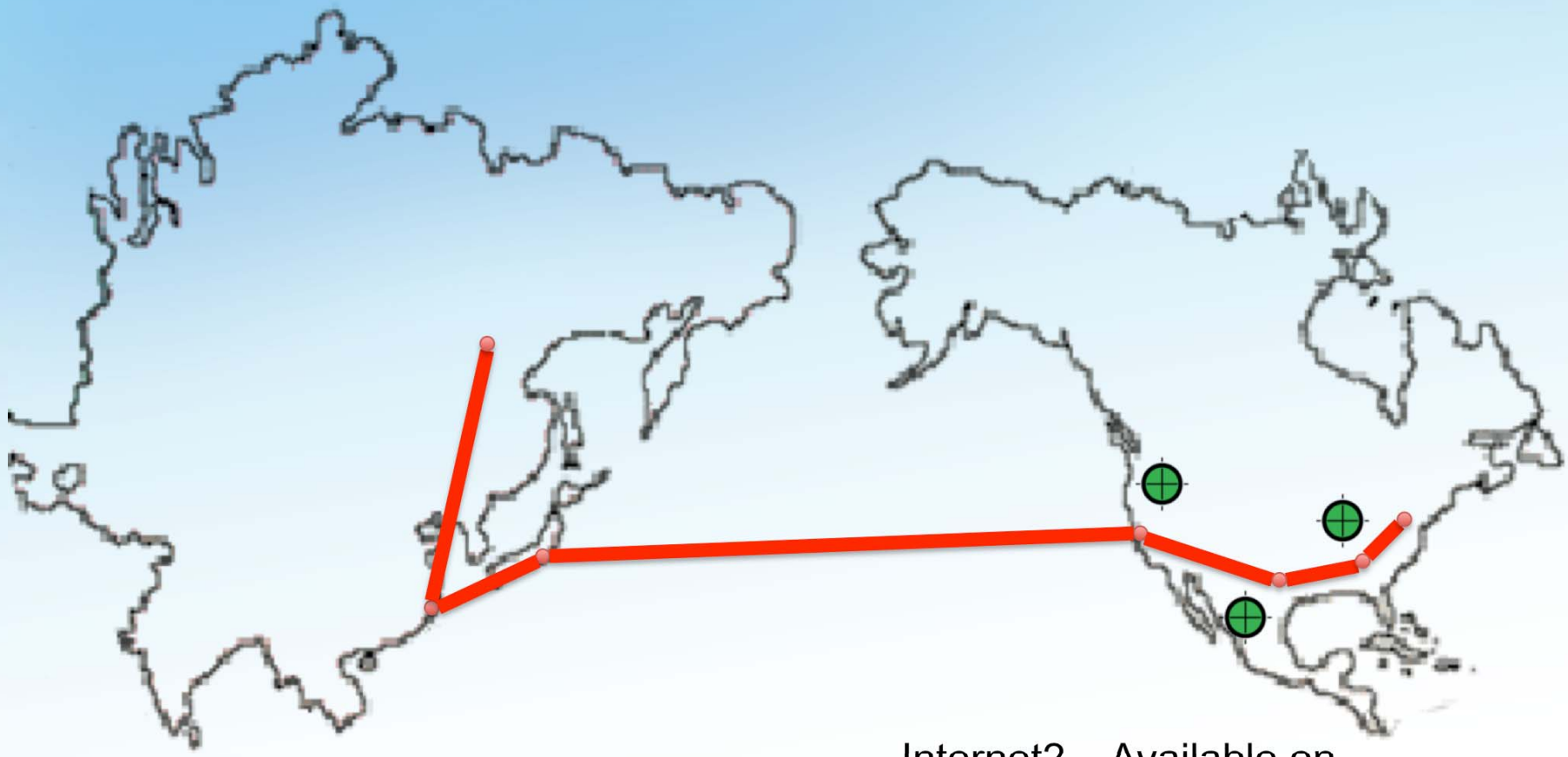
INTERNET 2

Solution: Test Points + Regular Monitoring



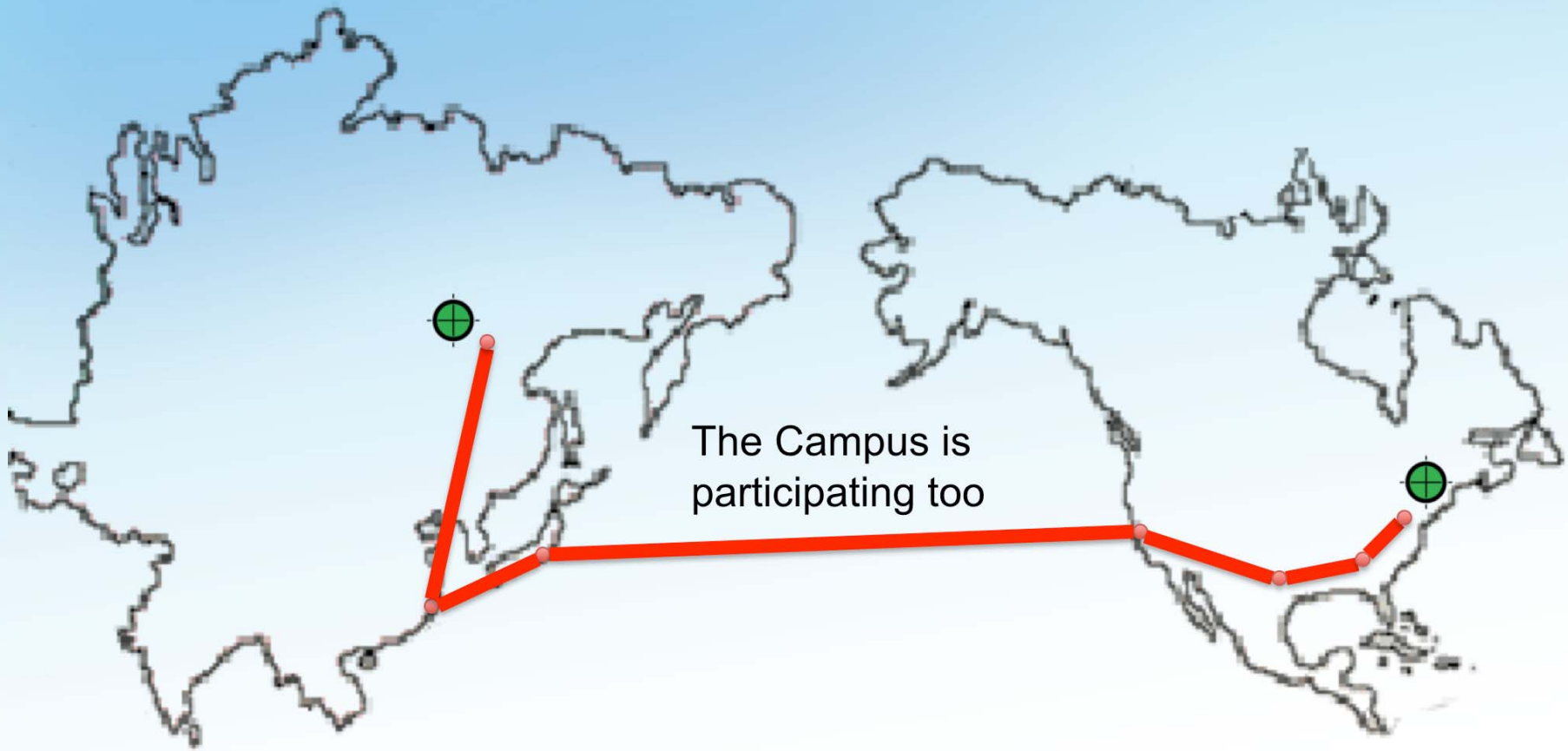
Key Point: End to end monitoring
Requires participation from all domains

Typical: Poor Performance … Somewhere

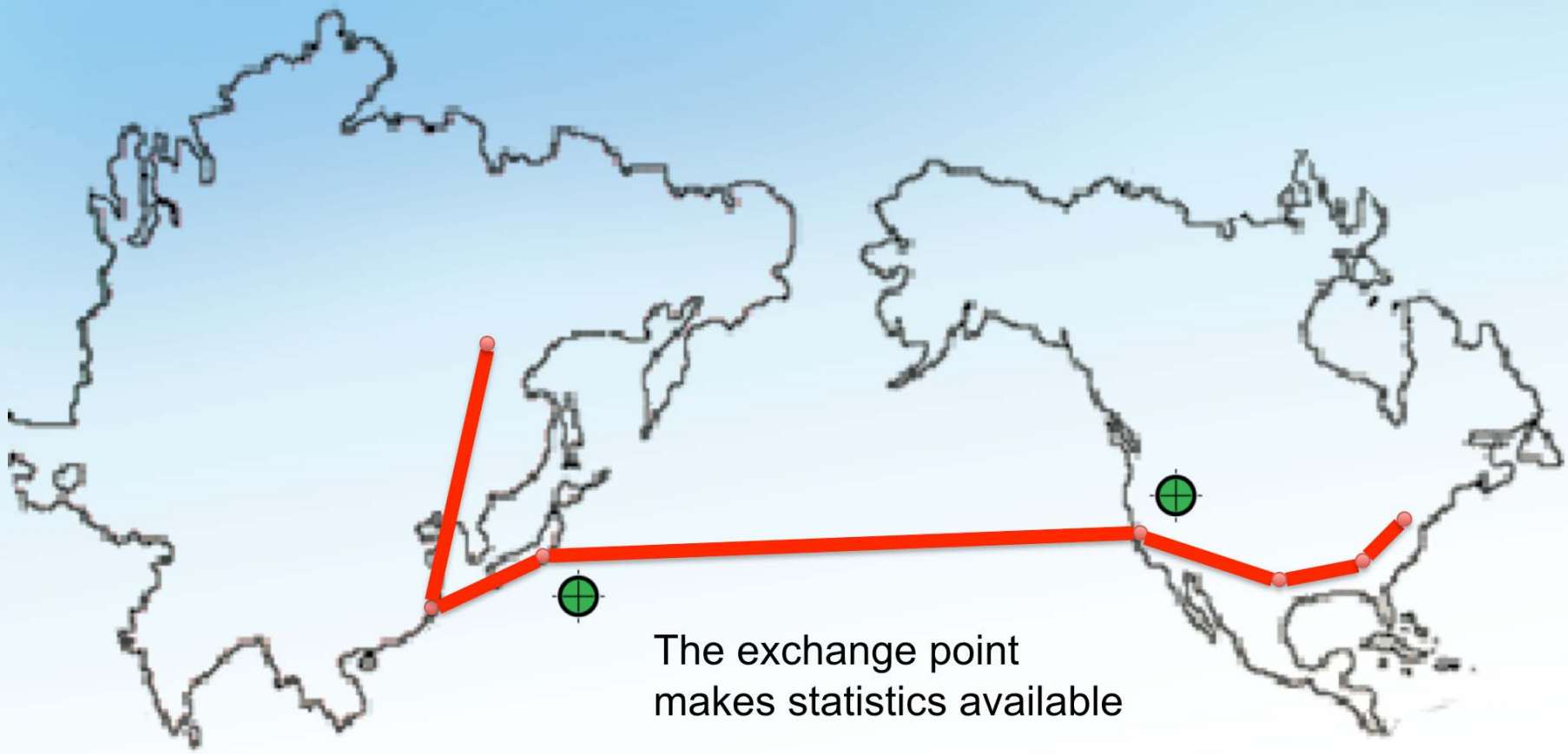


Internet2 – Available on the backbone

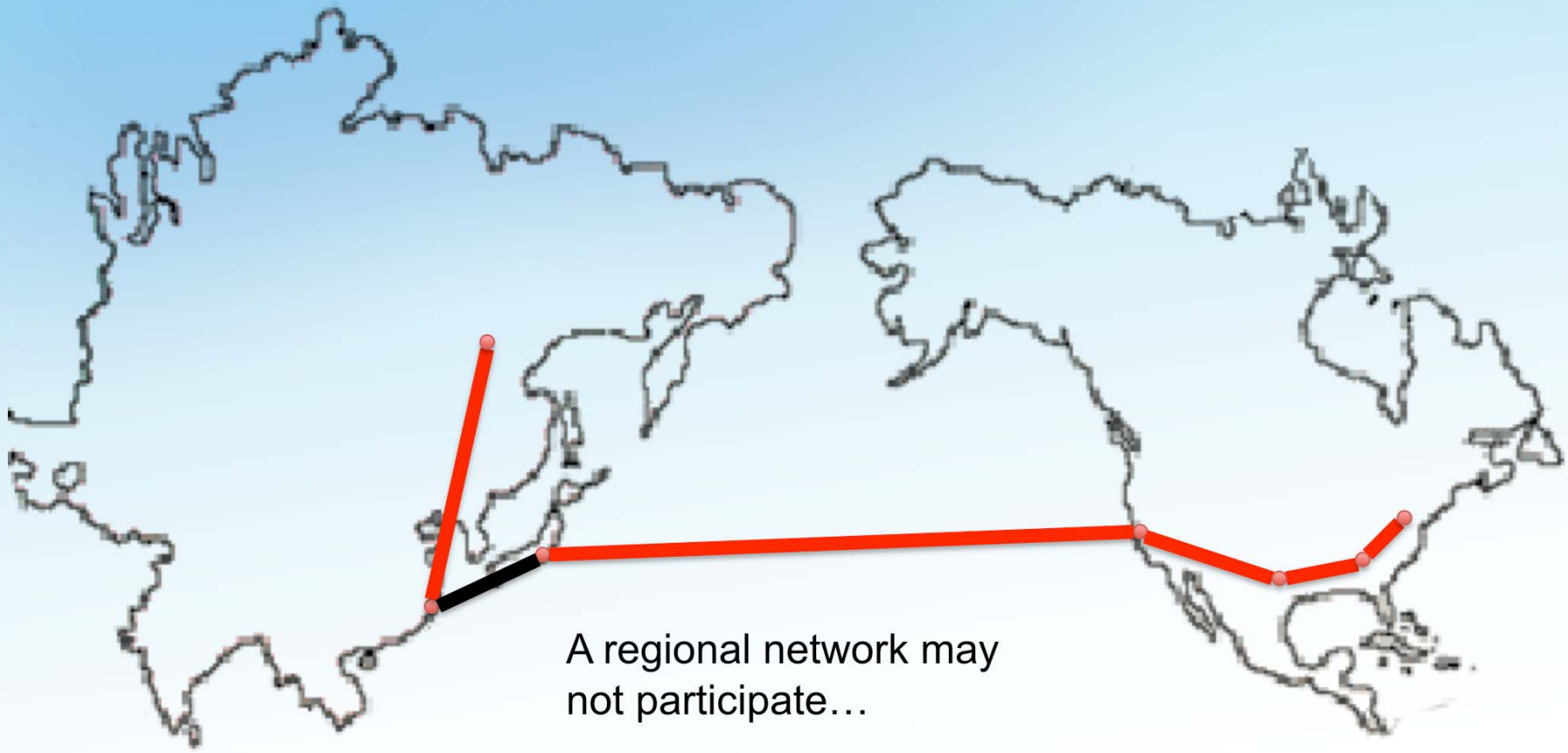
Typical: Poor Performance … Somewhere



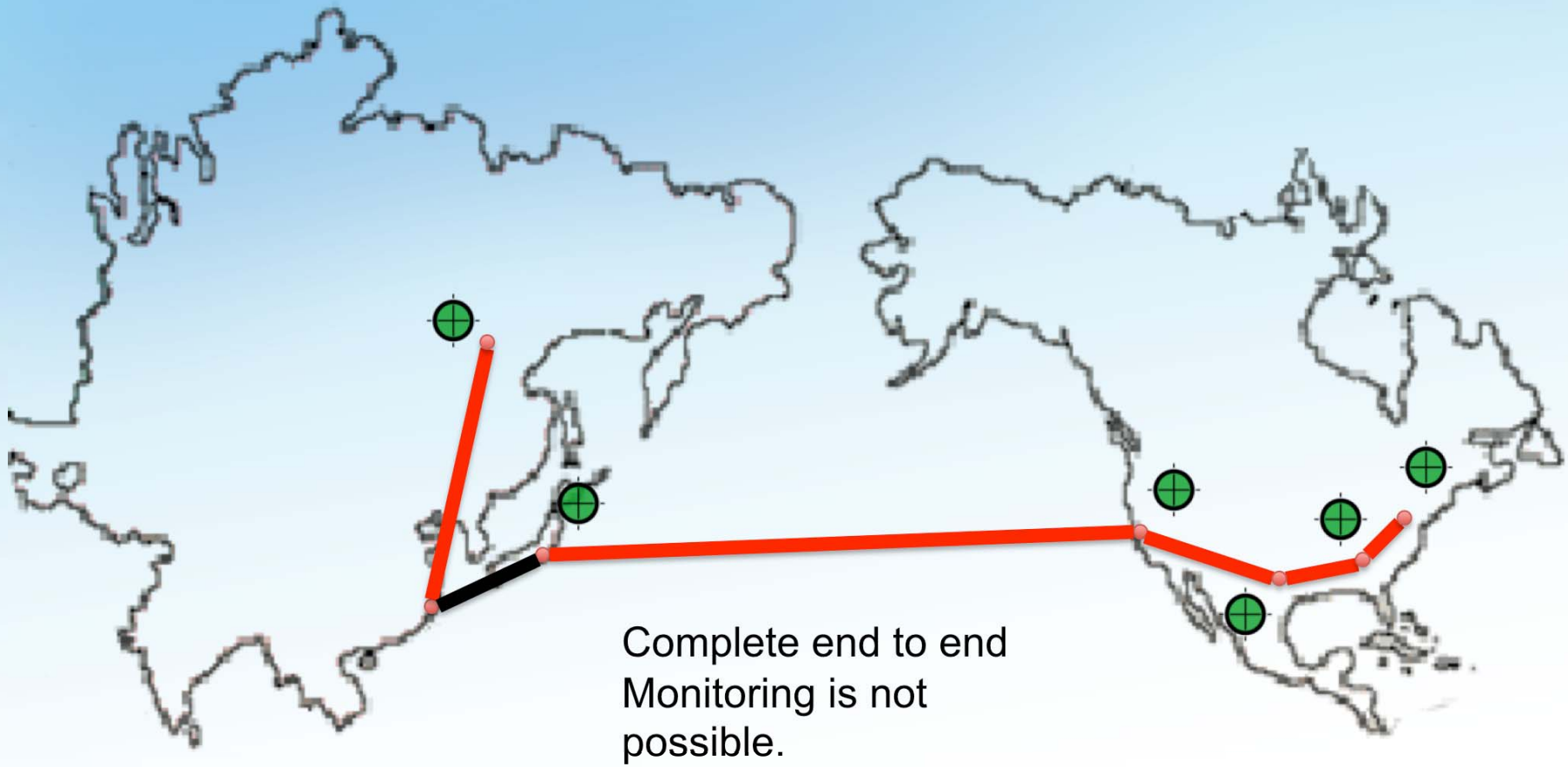
Typical: Poor Performance … Somewhere



Typical: Poor Performance … Somewhere



Typical: Poor Performance … Somewhere



Lessons Learned

- Missing part of the path leaves us with a huge disadvantage
- May discover some problems through isolation on the path we know, could miss something
  - Most network problems occur on the demarcation between networks
  - Testing *around* the problem won't work (we still have to transit this network)

Why is Science Data Movement Different?

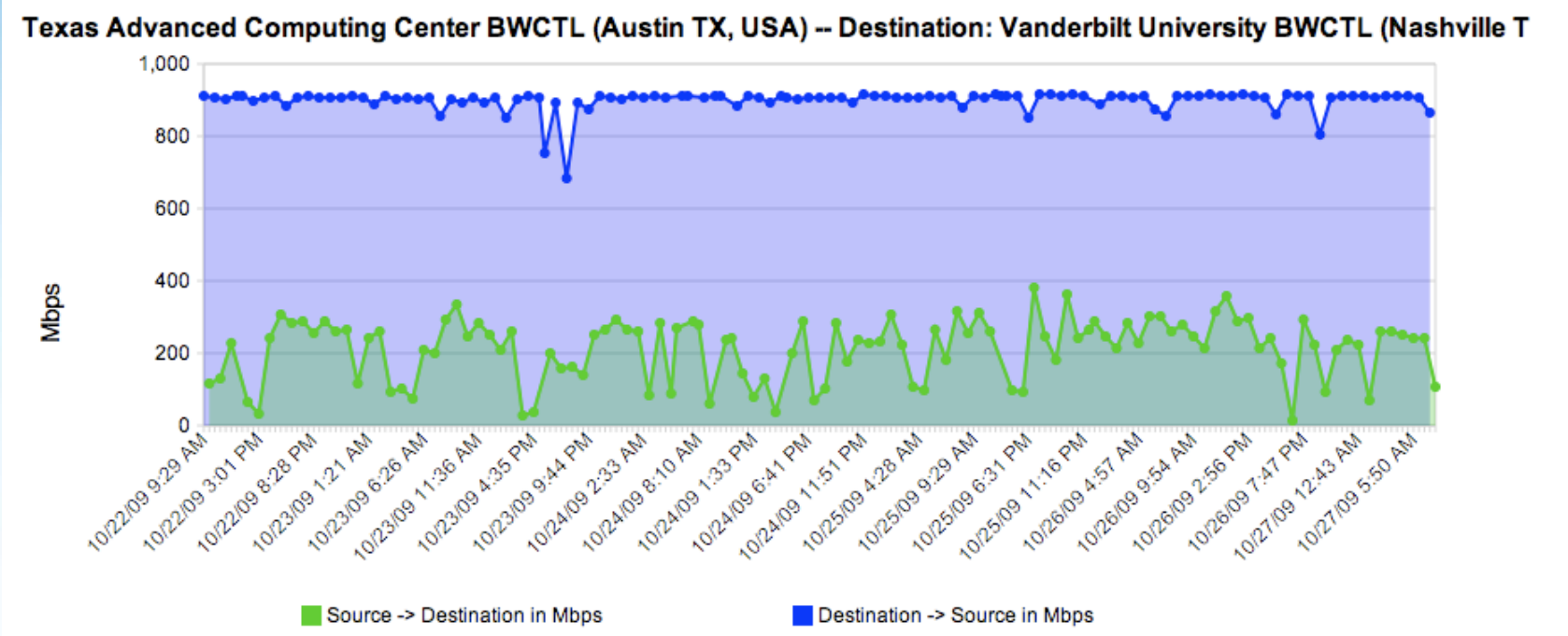
- Different Requirements
  - Campus network is not designed for large flows
    - ***Enterprise*** requirements
    - 100s of Mbits is common, any more is rare (or viewed as *strange*)
    - Firewalls
    - Network is designed to mitigate the risks since the common hardware (e.g. Desktops and Laptops) are un-trusted
  - Science is different
    - Network needs to be robust and stable (e.g. predictable performance)
    - 10s of Gbits of traffic (N.B. that its probably not sustained – but could be)
    - Sensitive to enterprise protections (e.g. firewalls, LAN design)
- ***Fixing*** is not easy
  - Design the base network for science, attach the enterprise on the side (expensive, time consuming, and good luck convincing your campus this is necessary…)
  - Mitigate the problems by moving your science equipment to the edge
    - Try to bypass that firewall at all costs
    - Get as close to the WAN connection as you can

Identifying Common Network Problems

- The above examples paint a broad picture: there is a problem, *somewhere*, that needs to be fixed
- What could be out there?
  - Architecture
  - Common Problems, e.g. “Soft Failures”
- Myths and Pitfalls
  - Getting trapped is easy
  - Following a bad lead is easy too

Identifying Common Network Problems

- Audience Question: Would you complain if you knew what you were getting was not correct?



- N.B. Actual performance between Vanderbilt University and TACC – Should be about 1Gbps in both directions.

Identifying Common Network Problems

- Internet2/ESnet engineers will help members and customers debug problems if they are escalated to us
  - Goal is to solve the entire problem – end to end
  - Involves many parties (typical: End users as well as Campus, Regional, Backbone staff)
  - Slow process of locating and testing each segment in the path
  - Have tools to make our job easier (more on this later)
- Common themes and patterns for ***almost every*** debugging exercise emerge
  - Architecture (e.g. LAN design, Equipment Choice, Firewalls)
  - Configuration
  - "Soft Failures", e.g. something that doesn't severe connectivity, but makes the experience unpleasant

Architectural Considerations

- LAN vs WAN Design
  - Multiple Gbit flows [to the outside] should be close to the WAN connection
  - Eliminate the number of hops/devices/physical wires that may slow you down
  - Great performance on the LAN != Great performance on the WAN
- *You Get What you Pay For*
  - Cheap equipment will let you down
  - Network
    - Small Buffers, questionable performance (e.g. internal switching fabric can't keep up w/ LAN demand let alone WAN)
    - Lack of diagnostic tools (SNMP, etc.)
  - Storage
    - Disk throughput needs to be high enough to get everything on to the network
    - Plunking a load of disk into an incapable server is not great either
      - Bus performance
      - Network Card(s)

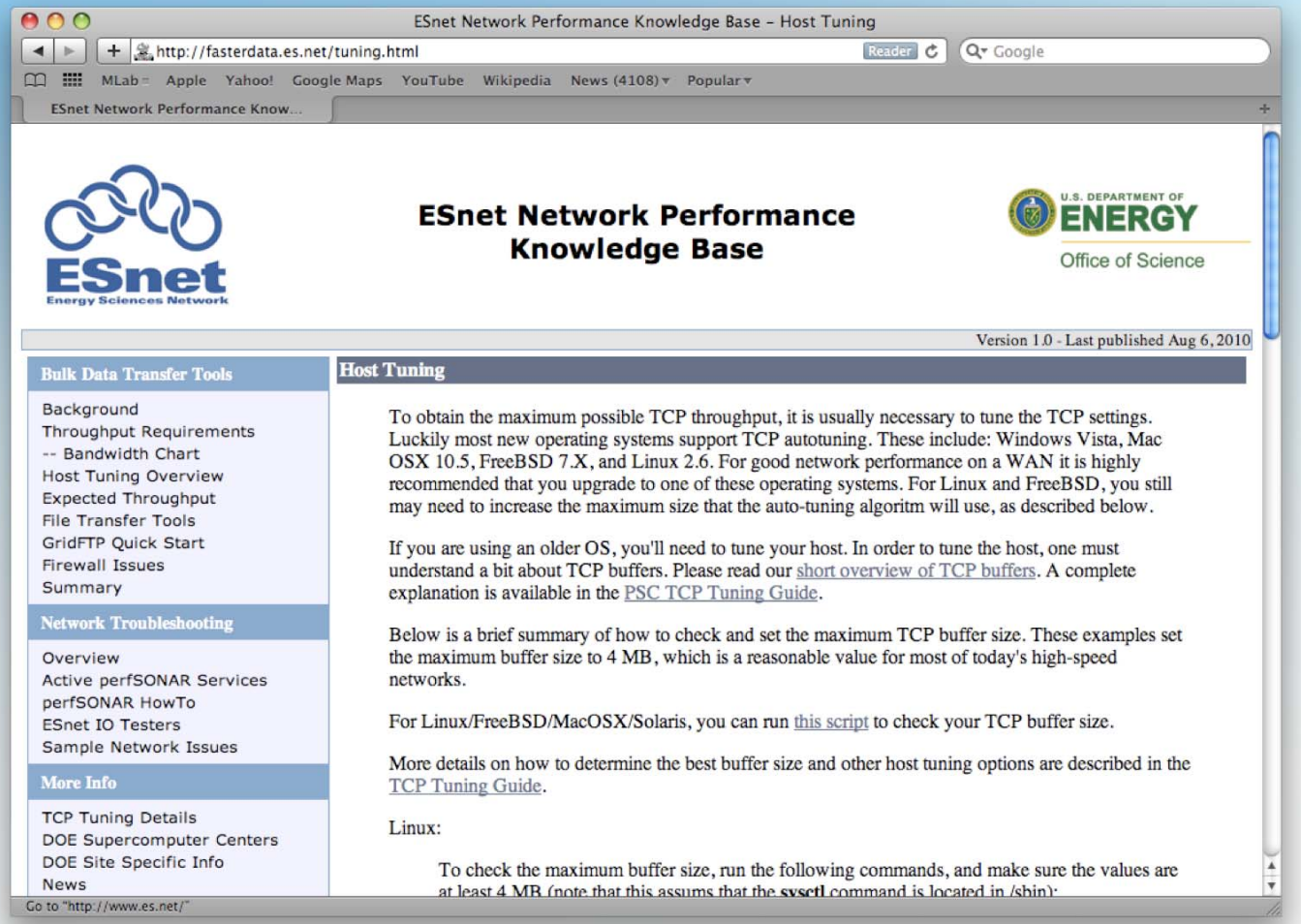
Architectural Considerations – cont.

- Firewalls
  - Designed to stop traffic
    - read this slowly a couple of times…
  - Small buffers
    - Concerned with protecting the network, not impacting your performance
  - Will be ***a lot*** slower than the original wire speed
  - A "***10G Firewall***" may handle 1 flow close to 10G, doubtful that it can handle a couple.
  - If *firewall-like* functionality is a must – consider using router filters instead

Configuration

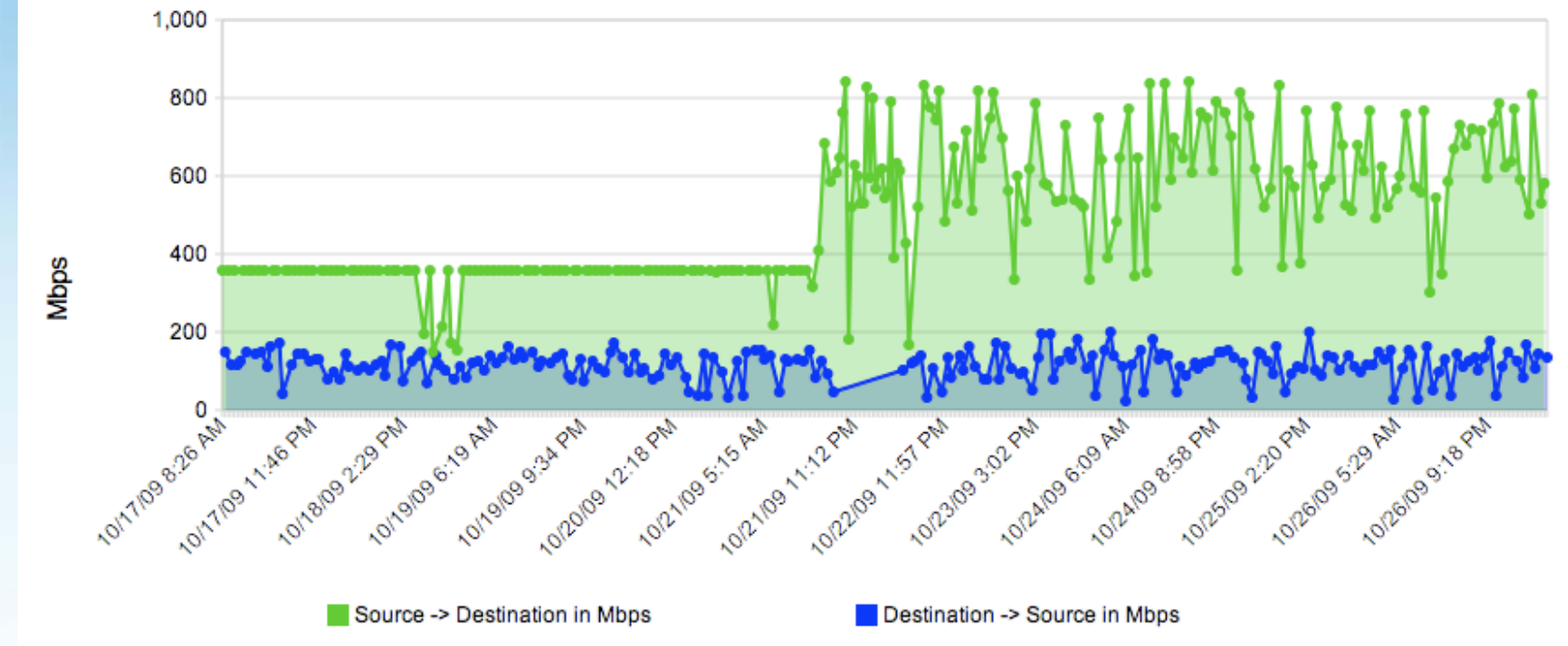
- Host Configuration
  - Tune your hosts (especially compute/storage!)
  - Changes to several parameters can yield 4 – 10X improvement
  - Takes minutes to implement/test
  - Instructions: http://fasterdata.es.net/tuning.html
- Network Switch/Router Configuration
  - ***Out of the box*** configuration may include small buffers
  - Competing Goals: Video/Audio etc. needs small buffers to remain responsive. Science flows need large buffers to push more data into the network.
  - Read your manuals and test LAN host to a WAN host to verify (not LAN to LAN).

Host Configuration



Configuration – cont.

- Host Configuration – spot when the settings were tweaked…



- N.B. Example Taken from REDDnet (UMich to TACC), using BWCTL measurement)

Soft Failures

- ***Soft Failures*** are any network problem that does not result in a loss of connectivity
  - Slows down a connection
  - Hard to diagnose and find
  - May go unnoticed by LAN users in some cases, but remote users may be the ones complaining
    - Caveat – How much time/energy do you put into listing to complaints of remote users?
- Common:
  - Dirty or Crimped Cables
  - Failing Optics/Interfaces
  - [Router] Process Switching, aka "*Punting*"
  - Router Configuration (Buffers/Queues)

Soft Failures – cont.

- Dirty or Crimped Cables and Failing Optics/Interfaces
  - Throw off very low levels of loss – may not notice on a LAN, will notice on the WAN
  - Will be detected with passive tools (e.g. SNMP monitoring)
  - Question: Would you fix it if you knew it was broken?
- [Router] Process Switching
  - "Punt" traffic to a slow path
- Router Configuration (Buffers/Queues)
  - Need to be large enough to handle science flows
  - Routing table overflow (e.g. system crawls to a halt when memory is exhausted)

Soft Failures – cont.

- Identifying and Fixing should be done through the use of monitoring and diagnostic tools
  - Establish testing points on the network
    - On the edge and in the center
  - Test to WAN points to find hidden/hard to diagnose problems
    - Where to Place and how to find?
      - Have collaborators co-allocate a testing machine
      - Use ***discovery*** tools to find them (e.g. perfSONAR)
  - Use an array of tools for different characteristics
    - Latency (One wan and Round Trip)
    - Bandwidth
    - Interface Utilization/Discards/Errors
    - Active vs Passive Testing

Myths and Pitfalls

- "My LAN performance is great, WAN is probably the same"
  - TCP recovers from loss/congestion quickly on the LAN (low RTT)
  - TCP will cut speed in half for every loss/discard on the WAN – will take a long time to recover for a large RTT/
  - Small levels of loss on the LAN (ex. 1/1000 packets) will go unnoticed, will be very noticeable on the WAN.
- "Ping is not showing loss/latency differences"
  - ICMP May be blocked/ignored by some sites
  - Routers process ICMP differently than other packets (e.g. may show phantom delay)
  - ICMP may hide some (not all) loss.
  - Will not show asymmetric routing delays (e.g. taking a different path on send vs receive)
- Our goal is to dispel these and others by educating the proper way to verify a network – we have lots of tools at our disposal but using these in the appropriate order is necessary too

For more information

- General and MDM implementation: http://www.perfsonar.net
- The PS implementation: http://psps.perfsonar.net
- 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

Diagnostics vs Regular Monitoring

August 10th 2010, OSG Site Admin Workshop – Network Performance

Jason Zurawski – Internet2

For more information, visit http://www.internet2.edu/workshops/npw