Strategies:
How to deploy and what to measure?

perfSONAR 5.2 training, Dublin, Ireland

December 2 & 3, 2025

Lætitia Delvaux, PCSS

perfSONAR is developed by a partnership of

Regular perfSONAR Tests

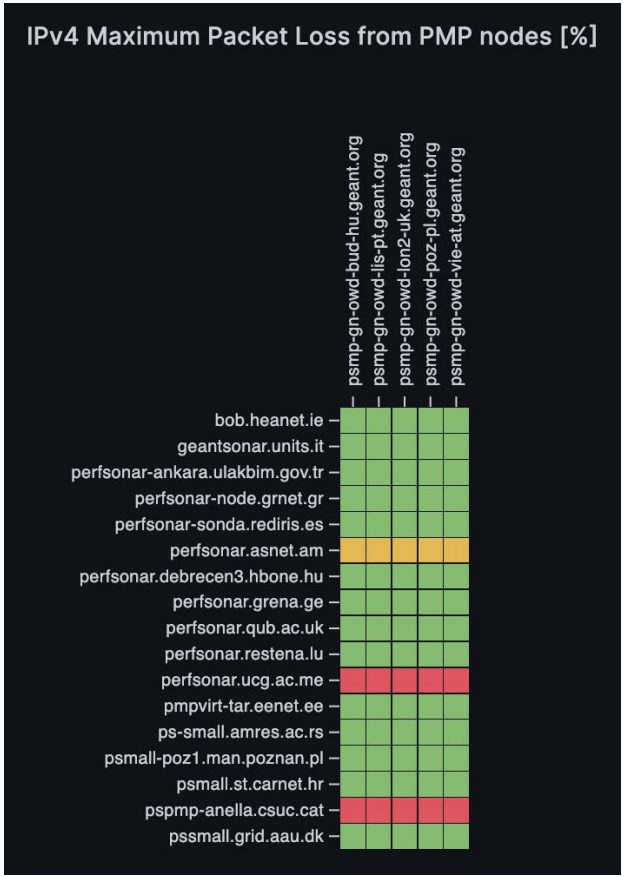
- We run regular tests to check for three things
  - TCP throughput
  - One way packet loss and delay
  - traceroute
- perfSONAR has mechanisms for managing regular testing between perfSONAR hosts
  - Statistics collection and archiving
  - Graphs
- This infrastructure is deployed now – perfSONAR hosts at facilities can take advantage of it
- At-a-glance health check for data infrastructure

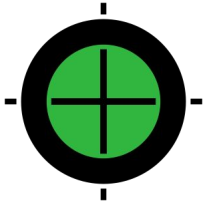
Importance of Regular Testing

- We can’t wait for users to report problems and then fix them (soft failures can go unreported for years!)
- Things just break sometimes
  - Failing optics
  - Somebody messed around in a patch panel and kinked a fiber
  - Hardware goes bad
- Problems that get fixed have a way of coming back
  - System defaults come back after hardware/software upgrades
  - New employees may not know why the previous employee set things up a certain way and back out fixes
- Important to continually collect, archive, and alert on active throughput test results

Dashboard grids

Example, the PMP dashboard at https://pmp-archive.geant.org





Deployment plan

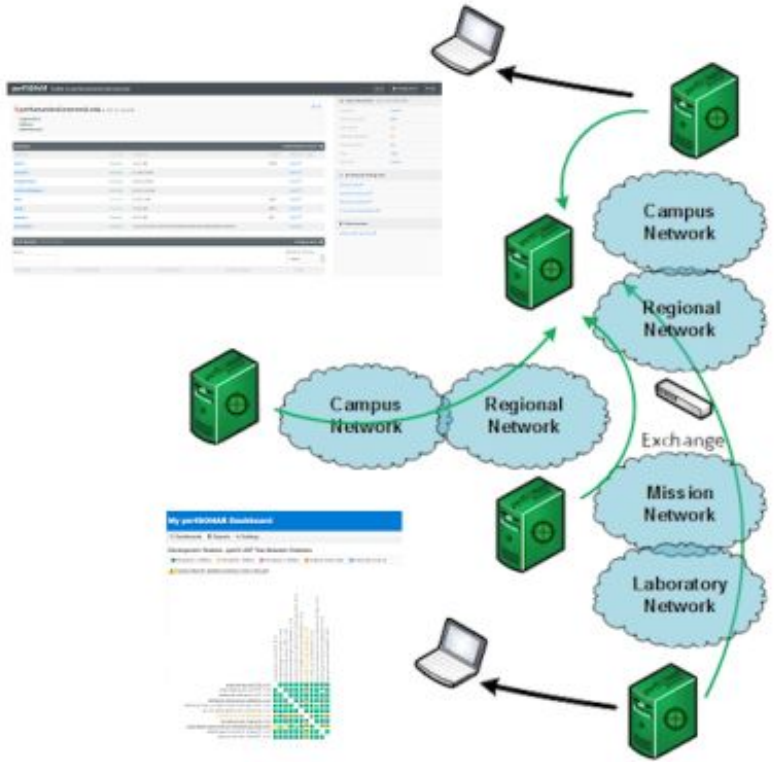
How to build a useful measurement mesh?

Regular Testing

- There are a couple of ways to do this.
  - Beacon: Let others test to you (e.g. no regular configuration is needed)
  - Island: Pick some hosts to test to – you store the data locally. No coordination with others is needed
  - Mesh: full coordination between you and others (e.g. consume a testing configuration that includes tests to everyone, and incorporate into a visualization)

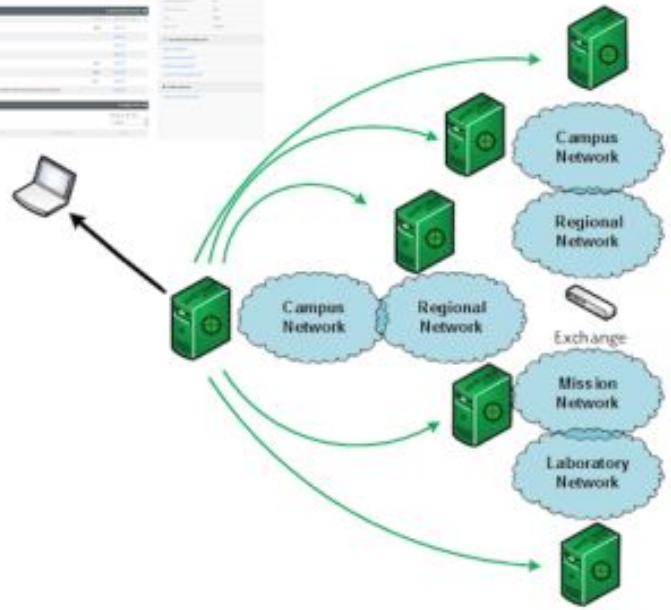
Regular Testing - Beacon

- Typically employed by a network provider (regional, backbone, IX)
  - A service to the users (allows people to test into the network)
  - If no regular tests are scheduled, minimum requirements for local storage.



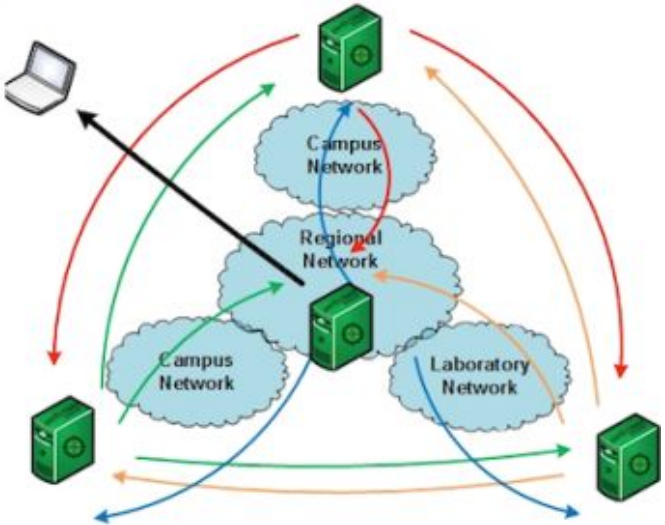
Regular Testing - Island

- A site tests against any number of the 2000+ perfSONAR nodes around the world, and store the data locally.
  - No coordination required with other sites
  - Allows a view of near horizon testing (e.g. short latency – campus, regional) and far horizon (backbone network, remote collaborators).
  - OWAMP is particularly useful for determining packet loss in the previous cases.
  - Throughput will not be as valuable when the latency is small



Regular Testing - Mesh

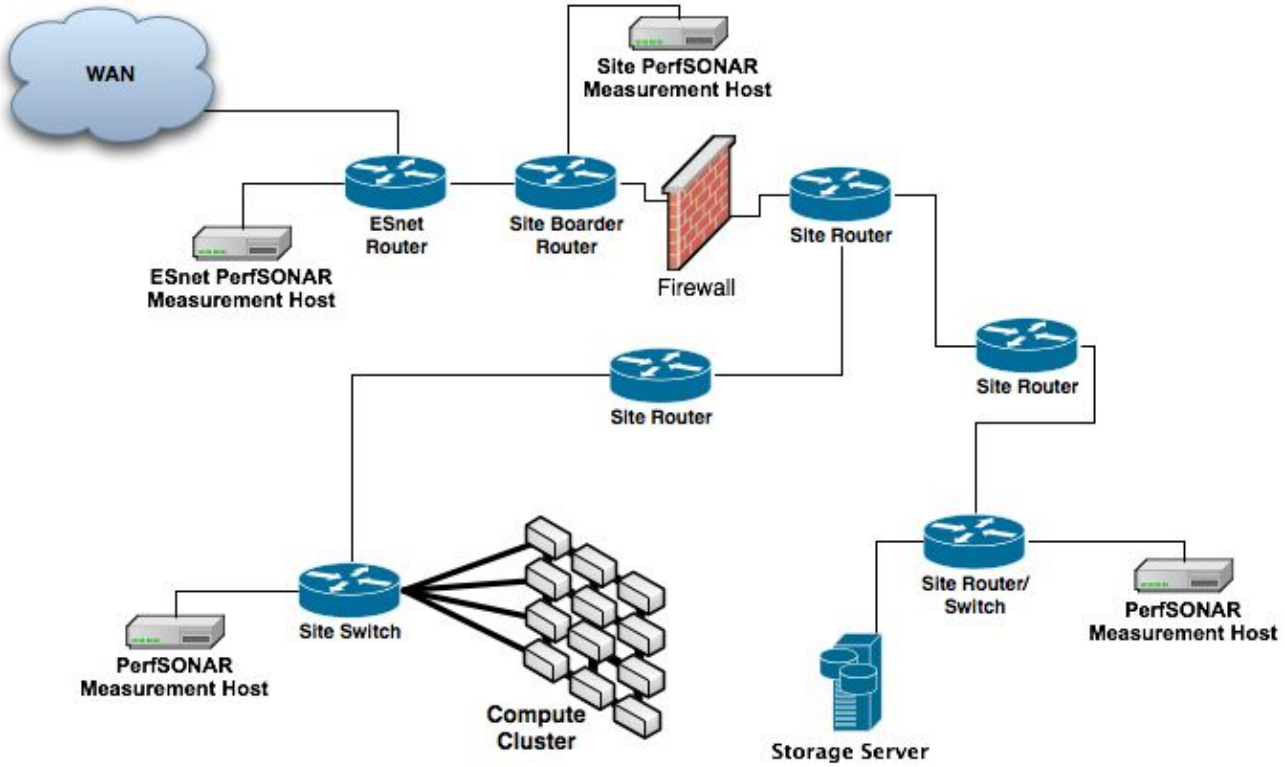
- A full mesh requires more coordination:
  - A full mesh means all hosts involved are running the same test configuration
  - A partial mesh could mean only a small number of related hosts are running a testing configuration
- In either case – bandwidth and latency will be valuable test cases



perfSONAR Deployment Locations

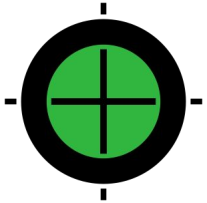
- Critical to deploy near key resources such as DTNs, big data sources
- More perfSONAR hosts allow segments of the path to be tested separately
  - Reduced visibility for devices between perfSONAR hosts
  - Must rely on counters or other means where perfSONAR can’t go
- Effective test methodology derived from protocol behavior
  - TCP suffers much more from packet loss as latency increases
  - TCP is more likely to cause loss as latency increases
  - Testing should leverage this in two ways
    - Design tests so that they are likely to fail if there is a problem
    - Mimic the behavior of production traffic as much as possible
  - Note: don’t design your tests to succeed
    - The point is not to “be green” even if there are problems
    - The point is to find problems when they come up so that the problems are fixed quickly

Sample Site Deployment



Develop a Test Plan

- What are you going to measure?
  - Achievable bandwidth
    - 2-3 regional destinations
    - 4-8 important collaborators
    - 4-8 times per day to each destination
    - 20 second tests within a region, longer across oceans and continents
  - Loss/Availability/Latency
    - OWAMP: ~10-20 collaborators over diverse paths
  - Interface Utilisation & Errors (via SNMP or streaming telemetry)
  - Write it in a pSConfig file (will be for another workshop…)
- What are you going to do with the results?
  - Grafana Alerts
  - Reports to user community
  - Grids



Performance Troubleshooting

What to do when performance is not good

Where to Start?

- perfSONAR was designed to debug end-to-end problems.
  - Easily deployed where it needs to be
  - Selection of tools that can be invoked (remotely) on demand
  - Ability to set up regular tests, and view a historical record
- Before starting any investigation, create a plan (perhaps share this with your helpdesk/operations team too …)

Where to Start?

- As we get into this, we will be using a variation of '**divide and conquer**', but we need to be careful:
  - Segment by segment analysis is not helpful – this is not a 'many parts make a whole' problem, TCP doesn't work that way
  - We want to divide our path more intelligently, and build it/take it apart slowly. Testing every single segment from ingress to egress is a false way to diagnose an end-to-end problem

Developing a Plan

- Suggested Flow Chart:
  - Problem Report
  - Triage
  - Resource Identification
  - 1st Pass Fact Finding
  - Performance Testing – Step 1 (Regular Testing Setup)
  - Performance Testing – Step 2 (OWAMP)
  - Performance Testing – Step 3 (Throughput)
  - Performance Testing – Step 4 (Optional – NDT or other tools)
  - Evaluation of Results
  - Hypothesis & Environmental Changes

Putting Theory Into Practice

- Let’s look at a real life example using some of the steps above.
- Brevity is being employed, but please ask questions if you are not making the connections.

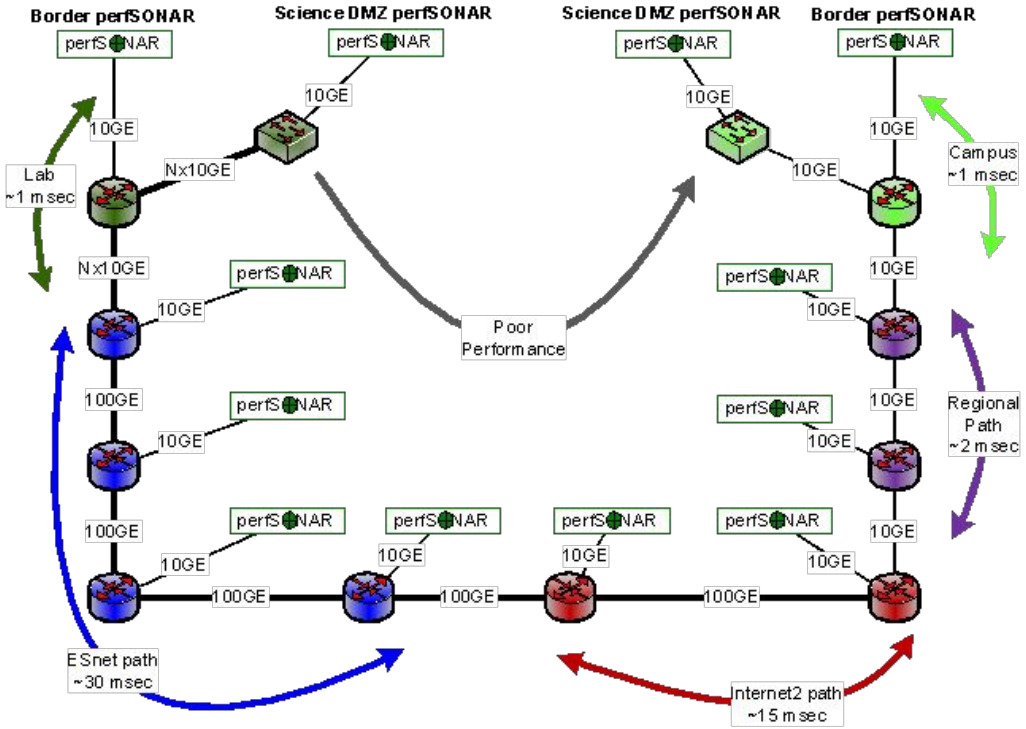
WAN Test Methodology – Problem Isolation

- We said it before, but it bears repeating: segment-to-segment testing is not helpful
  - TCP dynamics will be different, and in this case all the pieces do not equal the whole
    - E.g. high throughput on a 1ms path with high packet loss vs. the same segment in a longer 20ms path
  - Problem links can test clean over short distances
  - An exception to this is hops that go thru a firewall

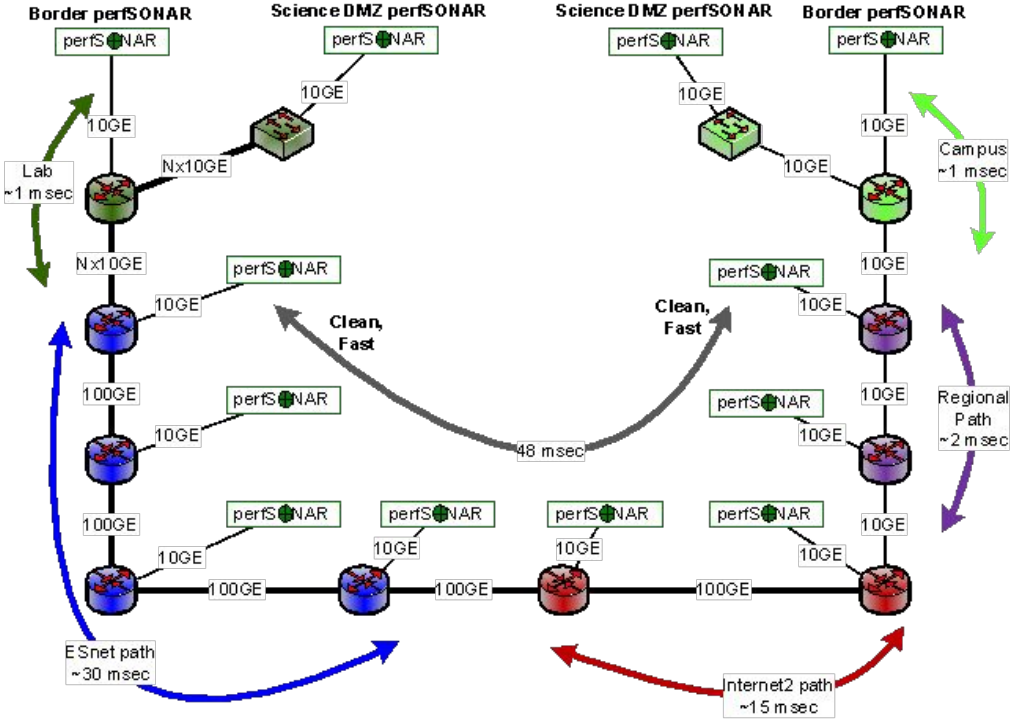
WAN Test Methodology – Problem Isolation

- Run long-distance tests
  - ***Run the longest clean test you can***, then look for the ***shortest dirty test*** that includes the path of the clean test
- In order for this to work, the testers need to be already deployed when you start troubleshooting
  - ESnet has at least one perfSONAR host at each hub location.
    - Many (most?) R&E providers in the world have deployed at least 1
  - If your provider does not have perfSONAR deployed ask them why, and then ask when they will have it done

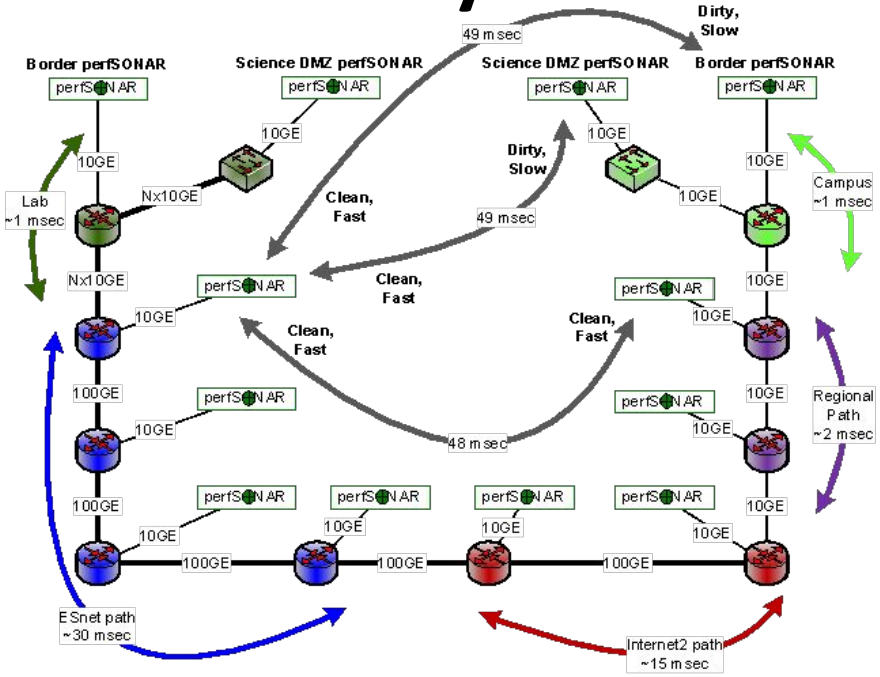
Network Performance Troubleshooting Example



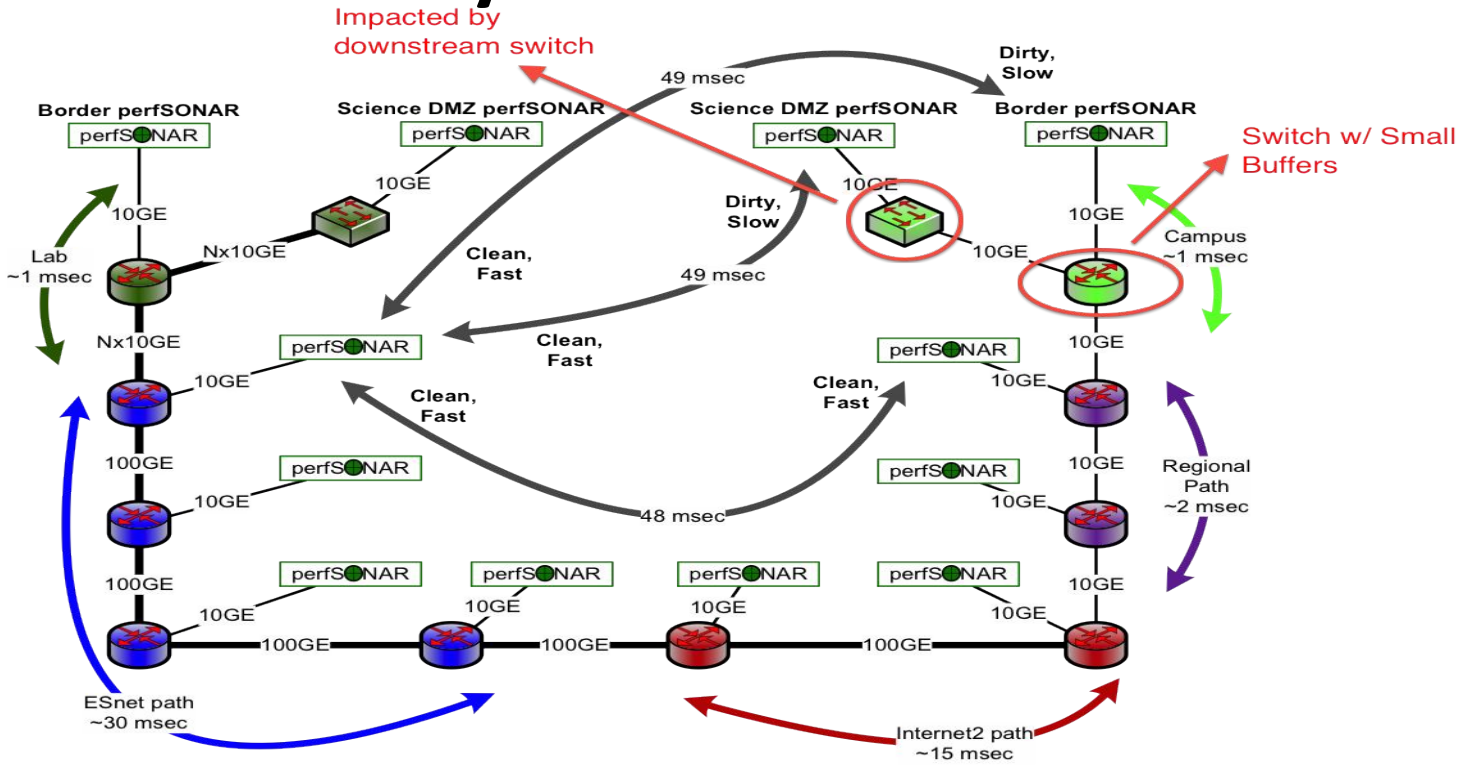
Wide Area Testing – Long Clean Test



Wide Area Testing – Poorly Performing Tests Illustrate Likely Problem Areas



Likely Problem Area



Lessons From This Example

- This testing can be done quickly if perfSONAR is already deployed
- Huge productivity
  - Reasonable hypothesis developed quickly
  - Probable administrative domain identified
  - Testing time can be short – an hour or so at most
- Without perfSONAR cases like this are very challenging
- Time to resolution measured in months
- In order to be useful for data-intensive science, the network must be fixable quickly, because it ***will*** break

Conclusions

- Lower level tools can help diagnose what the higher level tools are seeing
- Low throughput is the result of all network problems when using TCP
  - Out of Order Packets (OOP)
  - Loss
  - Retransmissions/Timeouts