
*http://fasterdata.es.net/performance-testing/2019-2020-data-mobility-workshop-and-exhibition/*

# CC* Data Movement Workshop and Exhibition – TCP & Performance

Jason Zurawski, Eli Dart
zurawski@es.net, dart@es.net
ESnet / Lawrence Berkeley National Laboratory

Dr. Jennifer M. Schopf
jmschopf@indiana.edu
Indiana University International Networks

*CC*/CICI PI Meeting Pre-Workshop*
*September 22nd 2019*



*National Science Foundation Award #1826994*

*https://epoc.global*

# Outline

- ***TCP Basics***
- Demo
- Science DMZ Design Context (e.g. Buffering)
- Monitoring

EPOC
Engagement and Performance Operations Center

# TCP – Ubiquitous and Fragile

- Networks provide connectivity between hosts – how do hosts see the network?
  - From an application's perspective, the interface to "the other end" is a socket
  - Communication is between applications – mostly over TCP
- TCP – the fragile workhorse
  - TCP is (for very good reasons) timid – packet loss is interpreted as congestion
  - Packet loss in conjunction with latency is a performance killer
  - Like it or not, TCP is used for the vast majority of data transfer applications (more than 95% of ESnet traffic is TCP)

EPOC
Engagement and Performance Operations Center

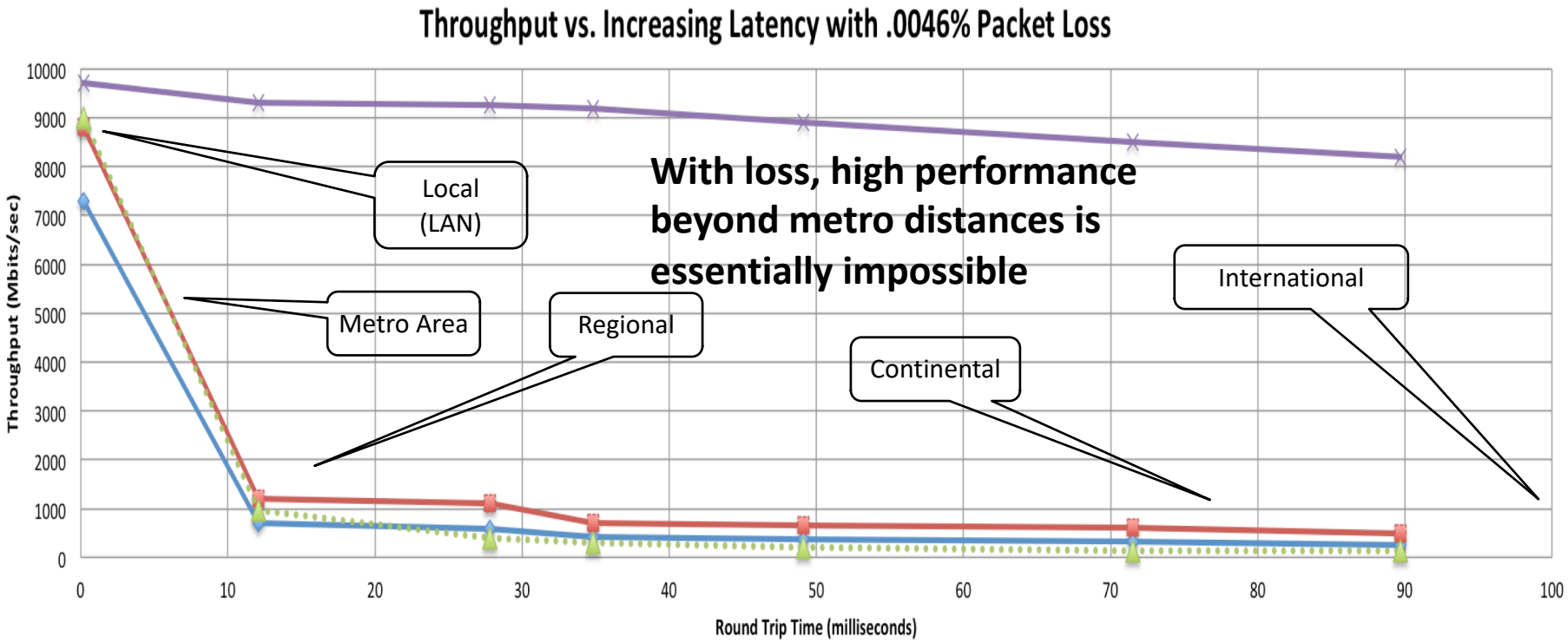
# Packet/Data Loss?!

- ***"Wait a minute! I thought TCP was a reliable protocol? What do you mean 'packet loss', where is the data going!?"***

- We are going to talk about this a lot.
  - The data isn't lost forever, its dropped somewhere on the path.
  - Usually by a device without enough buffer space to accept it, or by someone who thinks the data is corrupted and they won't send it
- Once its dropped, we have a way of knowing its been dropped.
  - TCP is reliable, each end is keeping track of what was sent, and what was received.
  - If something goes missing, its resent.
  - Resending is what takes the time, and causes the slowdown.

EPOC
Engagement and Performance Operations Center

# A small amount of packet loss makes a huge difference in TCP performance



# Outline

- TCP Basics
- ***Demo***
- Science DMZ Design Context (e.g. Buffering)
- Monitoring

EPOC
Engagement and Performance Operations Center

# Breaking a Network

- Disk PT Hosts (10G)
  - east-dc-pt1.es.net (New York, NY)
  - lbl-pt1.es.net (Berkeley, CA)
- Path
  - ~70ms RTT

```
[zurawski@lbl-pt1 ~]$ tracepath east-dc-pt1.es.net
 1?: [LOCALHOST]                                         pmtu 9000
 1:  lblmr2-lblpt1.es.net                                  0.266ms asymm  2
 1:  lblmr2-lblpt1.es.net                                  0.210ms asymm  2
 2:  sacrcr5-ip-a-lblmr2.es.net                            2.935ms
 3:  denvcr5-ip-a-sacrcr5.es.net                          24.421ms
 4:  kanscr5-ip-a-denvcr5.es.net                          34.469ms
 5:  chiccr5-ip-a-kanscr5.es.net                          45.426ms
 6:  washcr5-ip-a-chiccr5.es.net                          62.543ms
 7:  eqxashcr5-ip-a-eqxchicr5.es.net                      62.006ms
 8:  washcr5-ip-c-eqxashcr5.es.net                        62.428ms asymm  6
 9:  east-dc-pt1.es.net                                   69.388ms !H
     Resume: pmtu 9000
```

EPOC
Engagement and Performance Operations Center

# Forcing Bad Performance (to illustrate behavior)

- Add 10% Loss to a specific host

```
sudo /sbin/tc qdisc delete dev eth0 root
sudo /sbin/tc qdisc add dev eth0 root handle 1: prio
sudo /sbin/tc qdisc add dev eth0 parent 1:1 handle 10: netem loss 10%
sudo /sbin/tc filter add dev eth0 protocol ip parent 1:0 prio 3 u32 match ip dst 198.129.254.78/32 flowid 1:1
```

- Add 10% Duplication to a specific host

```
sudo /sbin/tc qdisc delete dev eth0 root
sudo /sbin/tc qdisc add dev eth0 root handle 1: prio
sudo /sbin/tc qdisc add dev eth0 parent 1:1 handle 10: netem duplicate 10%
sudo /sbin/tc filter add dev eth0 protocol ip parent 1:0 prio 3 u32 match ip dst 198.129.254.78/32 flowid 1:1
```

- Add 10% Corruption to a specific host

```
sudo /sbin/tc qdisc delete dev eth0 root
sudo /sbin/tc qdisc add dev eth0 root handle 1: prio
sudo /sbin/tc qdisc add dev eth0 parent 1:1 handle 10: netem corrupt 10%
sudo /sbin/tc filter add dev eth0 protocol ip parent 1:0 prio 3 u32 match ip dst 198.129.254.78/32 flowid 1:1
```


- Reorder packets: 50% of packets (with a correlation of 75%) will get sent immediately, others will be delayed by 75ms.

```
sudo /sbin/tc qdisc delete dev eth0 root
sudo /sbin/tc qdisc add dev eth0 root handle 1: prio
sudo /sbin/tc qdisc add dev eth0 parent 1:1 handle 10: netem delay 10ms reorder 25% 50%
sudo /sbin/tc filter add dev eth0 protocol ip parent 1:0 prio 3 u32 match ip dst 198.129.254.78/32 flowid 1:1
```

- Reset things

```
sudo /sbin/tc qdisc delete dev eth0 root
```

EPOC
Engagement and Performance Operations Center

# Outline

- TCP Basics
- Demo
- ***Science DMZ Design Context (e.g. Buffering)***
- Monitoring

EPOC
Engagement and Performance Operations Center

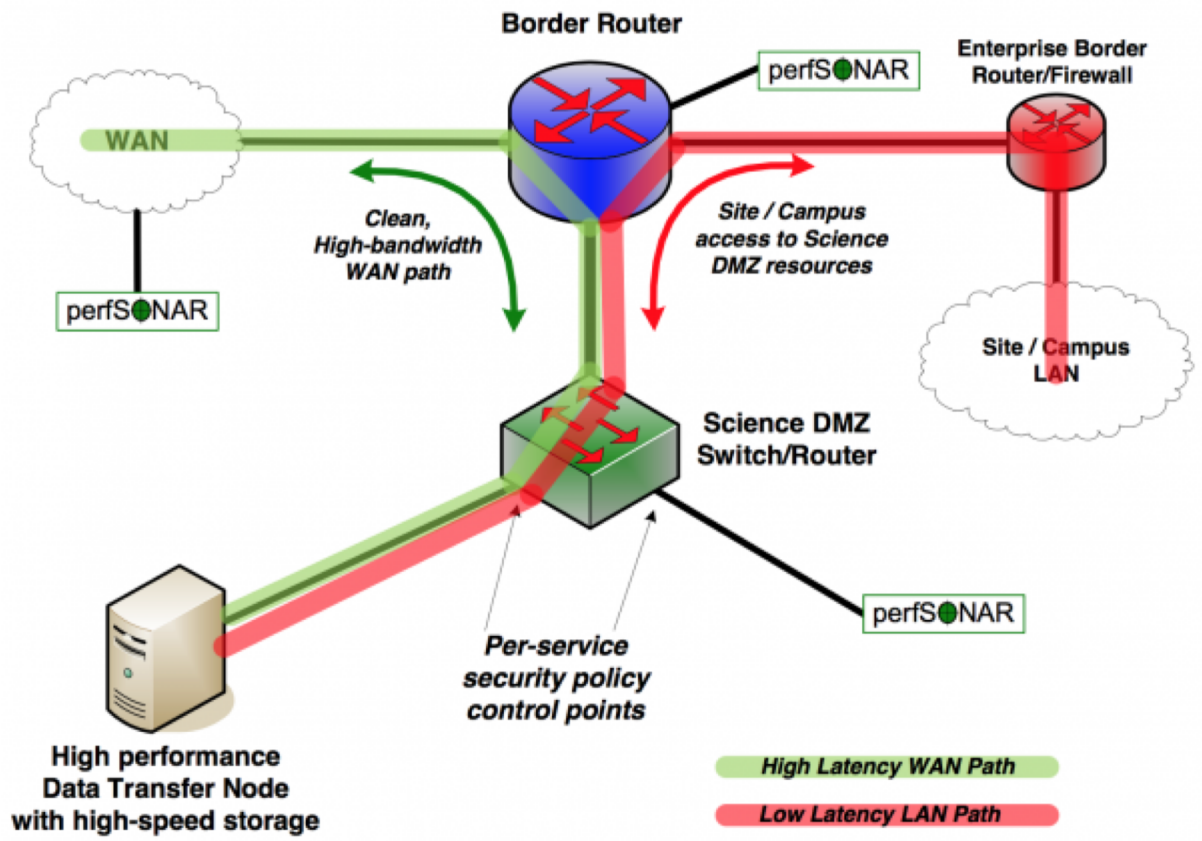
# How Do We Accommodate TCP?




- High-performance wide area TCP flows must get loss-free service
  - Sufficient bandwidth to avoid congestion
  - Deep enough buffers in routers and switches to handle bursts
    - Especially true for long-distance flows due to packet behavior
    - No, this isn't buffer bloat
- Equally important – the infrastructure must be verifiable so that clean service can be provided
  - Stuff breaks
    - Hardware, software, optics, bugs, …
    - How do we deal with it in a production environment?
  - Must be able to prove a network device or path is functioning correctly
    - Regular active tests should be run - perfSONAR
  - Small footprint is a huge win
    - Fewer the number of devices = easier to locate the source of packet loss

EPOC
Engagement and Performance Operations Center

# A better approach: simple Science DMZ




EPOC
Engagement and Performance Operations Center

# Common Threads

- Accommodation of TCP
  - Wide area portion of data transfers traverses purpose-built path
  - High performance devices that don't drop packets
- Ability to test and verify
  - When problems arise (and they always will), they can be solved if the infrastructure is built correctly
  - Small device count makes it easier to find issues
  - Multiple test and measurement hosts provide multiple views of the data path
    - perfSONAR nodes at the site and in the WAN
    - perfSONAR nodes at the remote site

EPOC
Engagement and Performance Operations Center

# Rant Ahead

N.B. You are entering into rant territory on the matter of switch buffering. If you are going to take away anything from the next section:

1. Under buffered network devices are the ***single greatest threat*** to data intensive use of the network. You can make hosts, operating systems, and application choices perform better for 'free', it will cost $$$ to fix a crappy switch or router
2. You will be steered toward non-optimal choices when you talk with the vendor community because they don't understand simple math (but by the end of this, you will).
3. A 1U/2U data center/racklan network device should never be in the path of your data intensive network use case.
4. Non-passive (e.g. stateful) security devices are the same for buffering, and are actually worse due to the processing overhead.
5. Anytime you jump around the OSI stack – add friction (e.g. routing when you don't need to, application layer inspection, etc.)

EPOC
Engagement and Performance Operations Center

# All About That Buffer (No Cut Through)

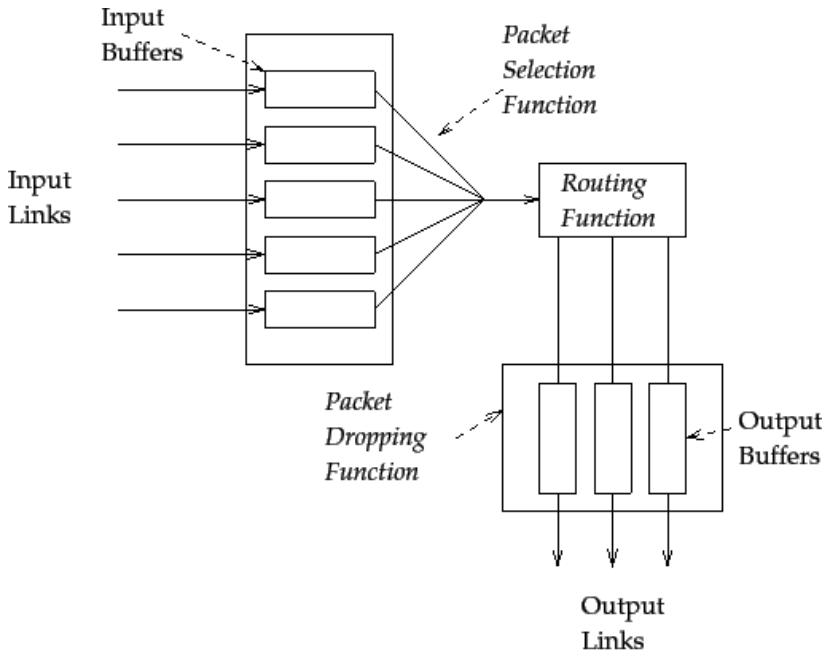



Figure 1: Basic Router Architecture

EPOC
Engagement and Performance Operations Center

# All About That Buffer (No Cut Through)

- Data arrives from multiple sources

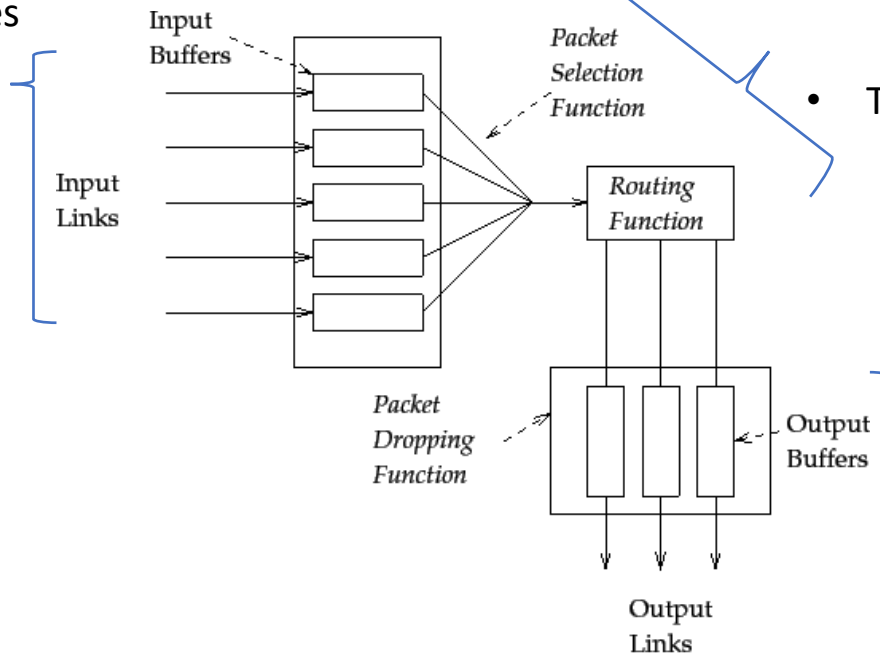
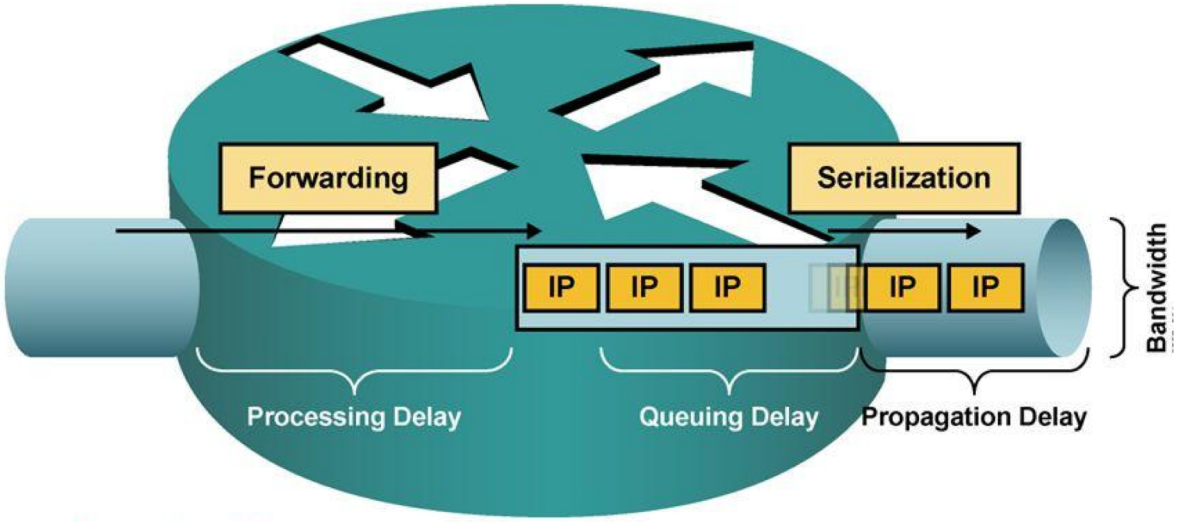



Figure 1: Basic Router Architecture

- Buffers have a finite amount of memory
  - Some have this per interface
  - Others may have access to a shared memory region with other interfaces
- The processing engine will:
  - Extract each packet/frame from the queues
  - Pull off header information to see where the destination should be
  - Move the packet/frame to the correct output queue
- Additional delay is possible as the queues physically write the packet to the transport medium (e.g. optical interface, copper interface)

EPOC Engagement and Performance Operations Center

# All About That Buffer (No Cut Through)




- **Processing delay:** The time it takes for a router to take the packet from an input interface, examine it, and put it into the output queue of the output interface.
- **Queuing delay:** The time a packet resides in the output queue of a router.
- **Serialization delay:** The time it takes to place the "bits on the wire."
- **Propagation delay:** The time it takes for the packet to cross the link from one end to the other.

EPOC
Engagement and Performance Operations Center

# All About That Buffer (No Cut Through)

- ***The Bandwidth Delay Product***
  - The amount of "in flight" data for a TCP connection (BDP = bandwidth * round trip time)
- Example: 10Gb/s cross country, ~100ms
  - 10,000,000,000 b/s * .1 s = 1,000,000,000 bits
  - 1,000,000,000 / 8 = 125,000,000 bytes
  - 125,000,000 bytes / (1024*1024) ~ ***125MB***
- Ignore the math aspect: its making sure there is memory to catch and send packets
  - As the speed increases, there are more packets.
  - If there is not memory, we drop them, and that makes TCP sad.

EPOC
Engagement and Performance Operations Center

# All About That Buffer (No Cut Through)

- Buffering isn't as important on the LAN (this is why you are normally pressured to buy 'cut through' devices)
  - Change the math to make the Latency 1ms = ***1.25MB***
  - 'Cut through' and low latency switches are designed for the data center, and can handle typical data center loads that don't require buffering (e.g. same to same speeds, destinations within the broadcast domain)

- Buffering *MATTERS* for WAN Transfers
  - Placing something with inadequate buffering in the path reduces the buffer for the entire path. E.g. if you have an expectation of 10Gbps over 100ms – don't place a 12MB buffer anywhere in there – your reality is now ~10x less than it was before (e.g. 10Gbps @ 10ms, or 1Gbps @ 100ms)

- Ignore the math aspect, its really just about making sure there is memory to catch packets. As the speed increases, there are more packets. If there is not memory, we drop them, and that makes TCP sad.
  - Memory on hosts, and network gear

EPOC
Engagement and Performance Operations Center

# All About That Buffer (No Cut Through)

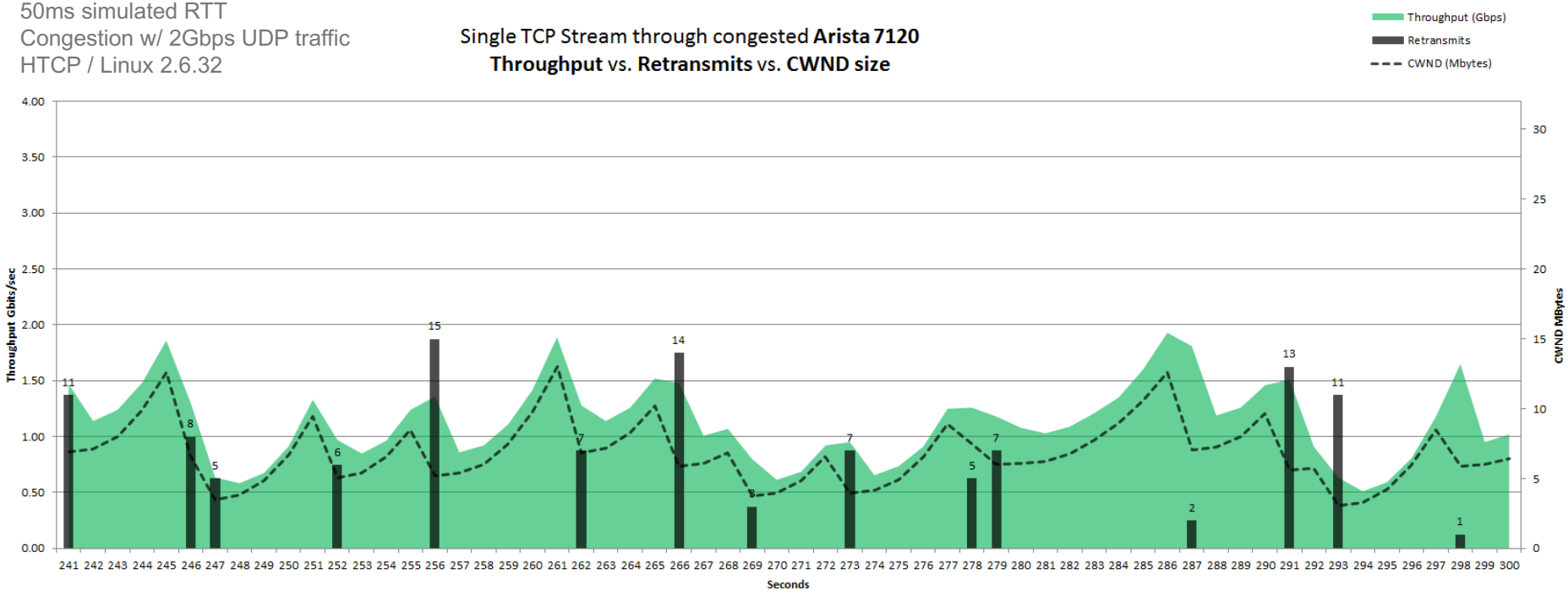
- What does this “look” like to a data transfer? Consider the test of iperf below
  - See TCP ‘ramp up’ and slowly increase the window
  - When something in the path has no more space for packets – a drop occurs. TCP will eventually react to the lost packet, and ‘back off’
  - In the example, this first occurs when we reach a buffer of around 6-8MB. Then after backoff the window is halved a couple of times
  - This happens again later – at a slightly higher buffer limit. This could be because there was cross traffic the first time, etc.

| [ ID] | Interval | | Transfer | Bandwidth | Retr | Cwnd |
|---|---|---|---|---|---|---|
| [ 14] | 0.00-1.00 | sec | 524 KBytes | 4.29 Mbits/sec | 0 | 157 KBytes |
| [ 14] | 1.00-2.00 | sec | 3.31 MBytes | 27.8 Mbits/sec | 0 | 979 KBytes |
| [ 14] | 2.00-3.00 | sec | 17.7 MBytes | 148 Mbits/sec | 0 | ***5.36 MBytes*** |
| ***[ 14]*** | ***3.00-4.00*** | ***sec*** | ***18.8 MBytes*** | ***157 Mbits/sec*** | ***214*** | ***1.77 MBytes*** |
| [ 14] | 4.00-5.00 | sec | 11.2 MBytes | 94.4 Mbits/sec | 0 | ***1.88 MBytes*** |
| [ 14] | 5.00-6.00 | sec | 10.0 MBytes | 83.9 Mbits/sec | 0 | ***2.39 MBytes*** |
| [ 14] | 6.00-7.00 | sec | 16.2 MBytes | 136 Mbits/sec | 0 | 3.63 MBytes |
| [ 14] | 7.00-8.00 | sec | 23.8 MBytes | 199 Mbits/sec | 0 | 5.50 MBytes |
| [ 14] | 8.00-9.00 | sec | 38.8 MBytes | 325 Mbits/sec | 0 | 8.23 MBytes |
| [ 14] | 9.00-10.00 | sec | 57.5 MBytes | 482 Mbits/sec | 0 | 11.8 MBytes |
| [ 14] | 10.00-11.00 | sec | 81.2 MBytes | 682 Mbits/sec | 0 | ***16.2 MBytes*** |
| ***[ 14]*** | ***11.00-12.00*** | ***sec*** | ***50.0 MBytes*** | ***419 Mbits/sec*** | ***35*** | ***3.93 MBytes*** |
| [ 14] | 12.00-13.00 | sec | 15.0 MBytes | 126 Mbits/sec | 0 | ***2.20 MBytes*** |
| [ 14] | 13.00-14.00 | sec | 11.2 MBytes | 94.4 Mbits/sec | 0 | ***2.53 MBytes*** |
| [ 14] | 14.00-15.00 | sec | 13.8 MBytes | 115 Mbits/sec | 1 | 1.50 MBytes |
| [ 14] | 15.00-16.00 | sec | 6.25 MBytes | 52.4 Mbits/sec | 5 | 813 KBytes |
| [ 14] | 16.00-17.00 | sec | 5.00 MBytes | 41.9 Mbits/sec | 0 | 909 KBytes |
| [ 14] | 17.00-18.00 | sec | 5.00 MBytes | 41.9 Mbits/sec | 0 | 1.37 MBytes |
| [ 14] | 18.00-19.00 | sec | 10.0 MBytes | 83.9 Mbits/sec | 0 | 2.43 MBytes |
| [ 14] | 19.00-20.00 | sec | 17.5 MBytes | 147 Mbits/sec | 0 | 4.22 MBytes |

EPOC
Engagement and Performance Operations Center

# TCP's Congestion Control

50ms simulated RTT
Congestion w/ 2Gbps UDP traffic
HTCP / Linux 2.6.32




Slide from Michael Smitasin, LBLnet

# Decoding Specifications

- *"The buffering behaviors of the switches and their operating system, such as behavior under memory stress, are typically proprietary information and not well documented"* http://www.measurementlab.net/blog/traffic-microbursts-and-their-effect-on-internet-measurement/
- "Even if you know ***how much*** packet buffer is in the switch, assumptions on ***how it is deployed*** that are not backed up by testing can lead to unhappiness. What we like to say is that is ***the job of the network engineers to move bottlenecks around***."
  - *Jim Warner*
- *http://people.ucsc.edu/~warner/buffer.html*

EPOC
Engagement and Performance Operations Center

# Decoding Specifications

- So lets say the spec sheet says this:

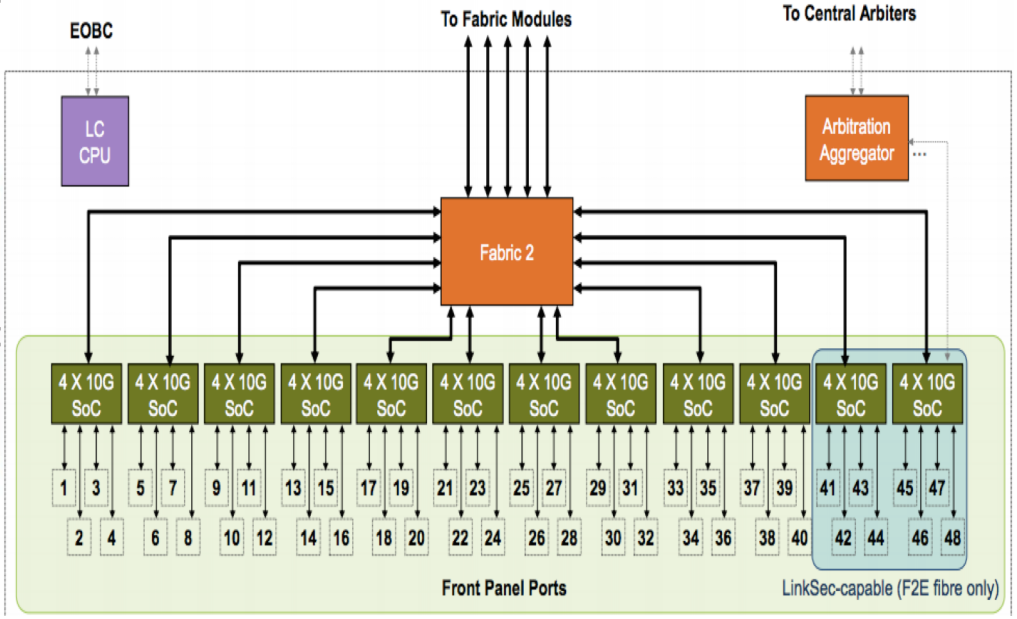
| | |
|---|---|
| **VOQ buffer** | 72 MB per module |

- What does ‘module’ mean?
  - Typically this means the amount of memory for the entire switch (if it’s a single unit) or a blade (if the chassis supports more than one.
  - BUT … this memory can be allocated in a number of different ways:
    - ***Shared between all ports***
    - ***Dedicated (smaller) amounts per-port***
    - ***Shared between ASICS, which control a bank of ports***

EPOC
Engagement and Performance Operations Center

# Decoding Specifications

- Consider this architecture
  - 48 Ports
    - 12 ASICS
    - 4 Ports per ASIC
  - ***72MB*** total
    - ***6MB per ASIC***
    - If all ports are in use – expect that eac port has access to ***1.5MB***. If only one in use, it can use 6MB
  - Additional memory is often availabl in a 'burst buffer' in the fabric




***ASIC = application-specific integrated circuit, think 'small routing engine'***

EPOC
Engagement and Performance Operations Center

# Decoding Specifications

- Recall: https://www.switch.ch/network/tools/tcp_throughput/
- What does 6MB get you?
  - 1Gbps @ <= 48ms (e.g. ½ needed for coast-to-coast)
  - 10Gbps @ <= 4.8ms (e.g. metro area)
- What does 1.5MB get you?
  - 1Gbps @ <= 12ms (e.g. regional area)
  - 10Gbps @ <= 1.2ms (e.g. data center [or more accurately, rack or row)]
- In either case – remember this assumes you are the only thing using that memory … congestion is a more likely reality

EPOC
Engagement and Performance Operations Center

# Outline

- TCP Basics
- Demo
- Science DMZ Design Context (e.g. Buffering)
- ***Monitoring***

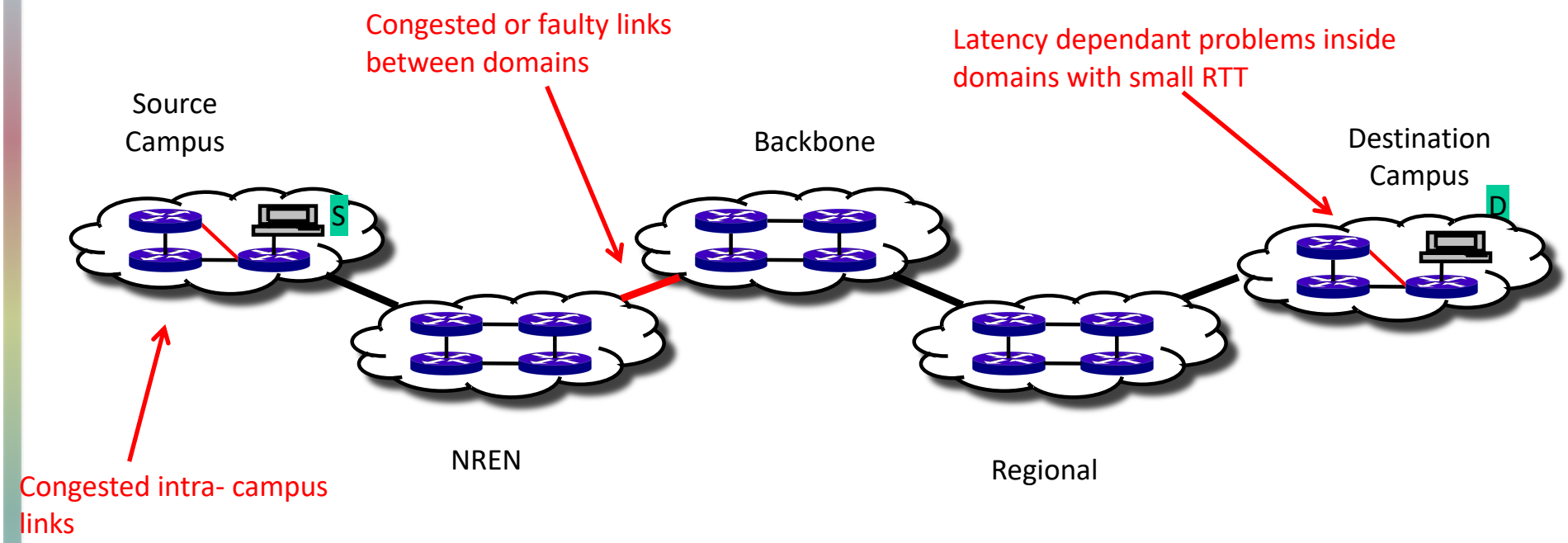
EPOC
Engagement and Performance Operations Center

# Test and Measurement – Keeping the Network Clean

- The wide area network, the Science DMZ, and all its systems can be functioning perfectly
- Eventually something is going to break
  - Networks and systems are built with many, many components
  - Sometimes things just break – this is why we buy support contracts
- Other problems arise as well – bugs, mistakes, whatever
- We must be able to find and fix problems when they occur
- Why is this so important?  Because we use TCP!

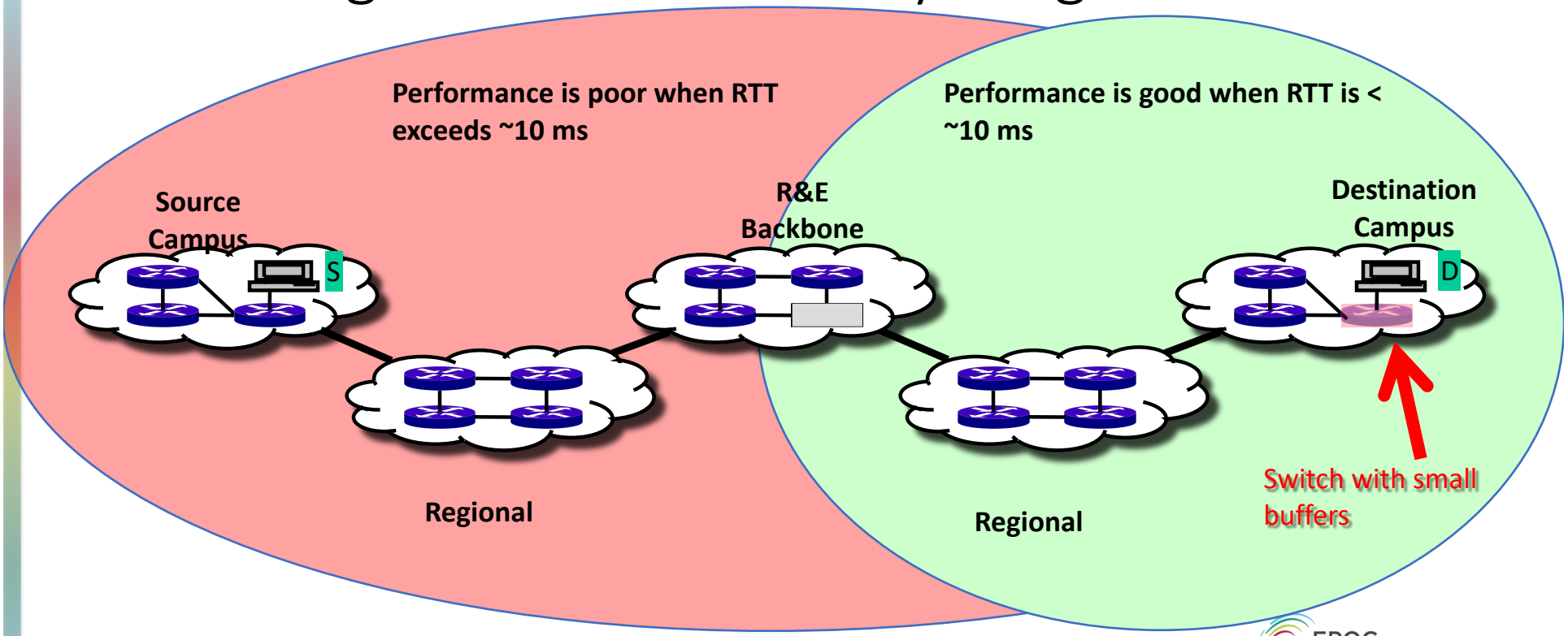
EPOC
Engagement and Performance Operations Center

# Where Are The Problems?




EPOC
Engagement and Performance Operations Center

# Local Testing Will Not Find Everything




EPOC
Engagement and Performance Operations Center

# Soft Network Failures

- Soft failures are where basic connectivity functions, but high performance is not possible.
- TCP was intentionally designed to hide all transmission errors from the user:
  - “As long as the TCPs continue to function properly and the internet system does not become completely partitioned, no transmission errors will affect the users.” (From IEN 129, RFC 716)
- Some soft failures only affect high bandwidth long RTT flows.
- Hard failures are easy to detect & fix
  - soft failures can lie hidden for years!
- One network problem can often mask others

EPOC
Engagement and Performance Operations Center

# Network Monitoring

- All networks do some form monitoring.
  - Addresses needs of local staff for understanding state of the network
    - Would this information be useful to external users?
    - Can these tools function on a multi-domain basis?
- Beyond passive methods, there are active tools.
    - E.g. often we want a ‘throughput’ number. Can we automate that idea?
    - Wouldn’t it be nice to get some sort of plot of performance over the course of a day? Week? Year? Multiple endpoints?
- perfSONAR = Measurement Middleware

EPOC
Engagement and Performance Operations Center

# What is perfSONAR?

- perfSONAR is a tool to:
  - Set network performance expectations
  - Find network problems (“soft failures”)
  - Help fix these problems
  - All in multi-domain environments
- These problems are all harder when multiple networks are involved
- perfSONAR provides a standardized package to publish active and passive monitoring data
  - This data is interesting to network researchers as well as network operators

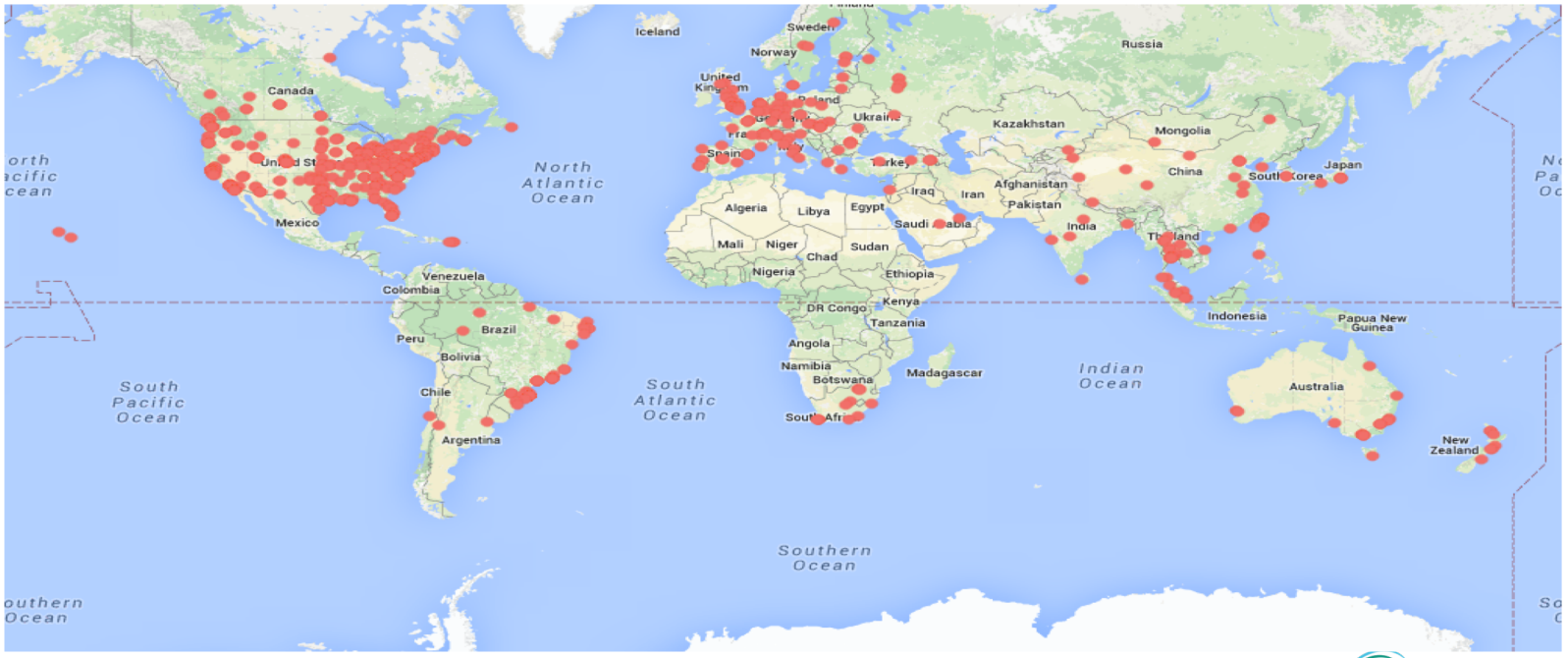
EPOC
Engagement and Performance Operations Center

# Host Considerations

- http://docs.perfsonar.net/install_hardware.html
- Dedicated perfSONAR hardware is best
  - Server class is a good choice
  - Desktop/Laptop/Mini (Mac, Shuttle) can be problematic, but work in a diagnostic capacity
- Other applications running may perturb results (and measurement could hurt essential services)
- Running Latency and Throughput on the Same Server
  - If you can devote 2 interfaces do.
  - If you can't, note that Throughput tests can cause increased latency and loss (latency tests on a throughput host are still useful however)
- Virtual Machines do not always work well as perfSONAR hosts (adding an extra layer of uncertainty)
  - Clock sync issues are a bit of a factor
  - throughput is reduced significantly for 10G hosts
  - VM technology and motherboard technology has come a long way, YMMV
  - NDT/NAGIOS/SNMP/1G BWCTL are good choices for a VM, OWAMP/10G BWCTL are not

EPOC
Engagement and Performance Operations Center

# http://stats.es.net/ServicesDirectory/ -




EPOC
Engagement and Performance Operations Center

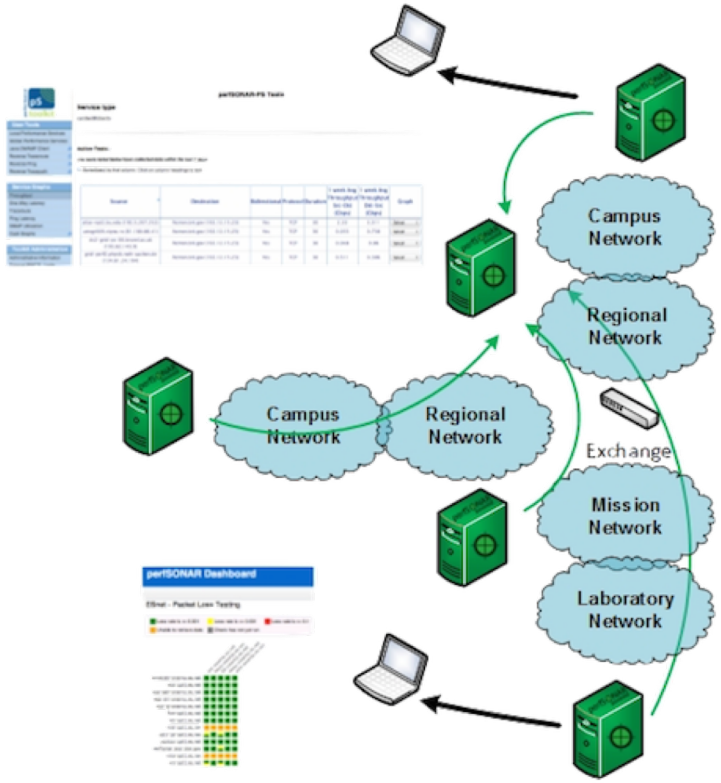
# Importance of Regular Testing

- 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
  - End to End path change
- 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

EPOC
Engagement and Performance Operations Center

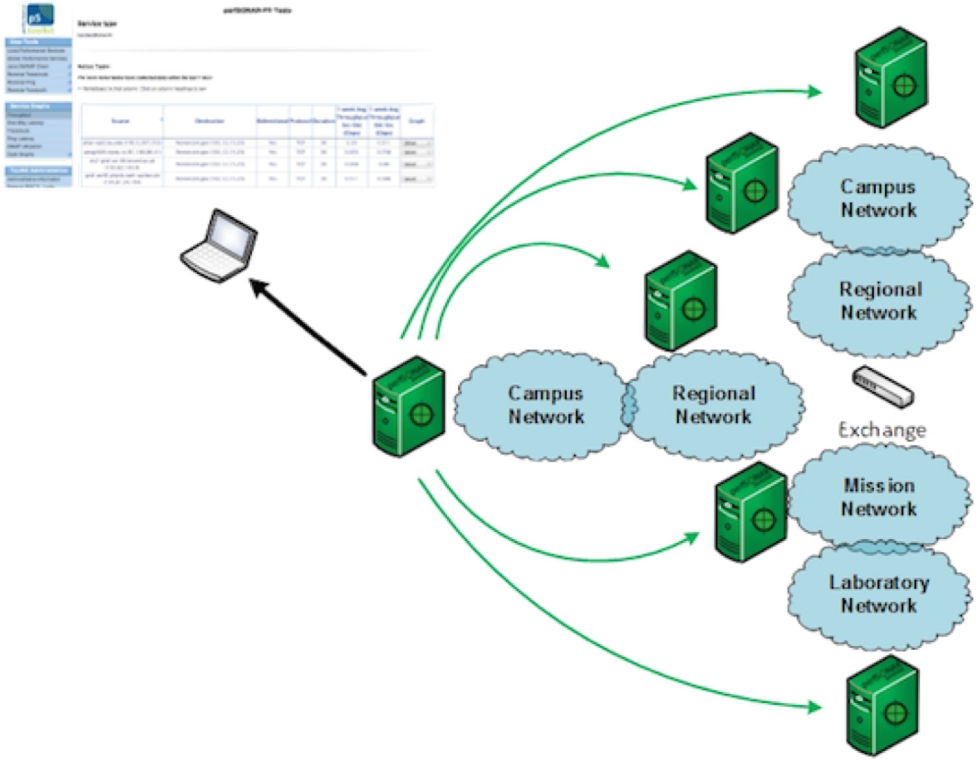
# Regular Testing - Beacon

- The beacon setup is typically employed by a network provider (regional, backbone, exchange point)
  - A service to the users (allows people to test into the network)
  - Can be configured with Layer 2 connectivity if needed
  - If no regular tests are scheduled, minimum requirements for local storage.
  - Makes the most sense to enable all services (bandwidth and latency)



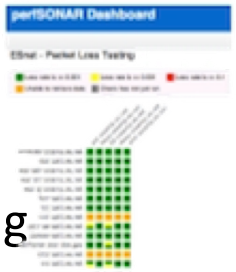
# Regular Testing - Island

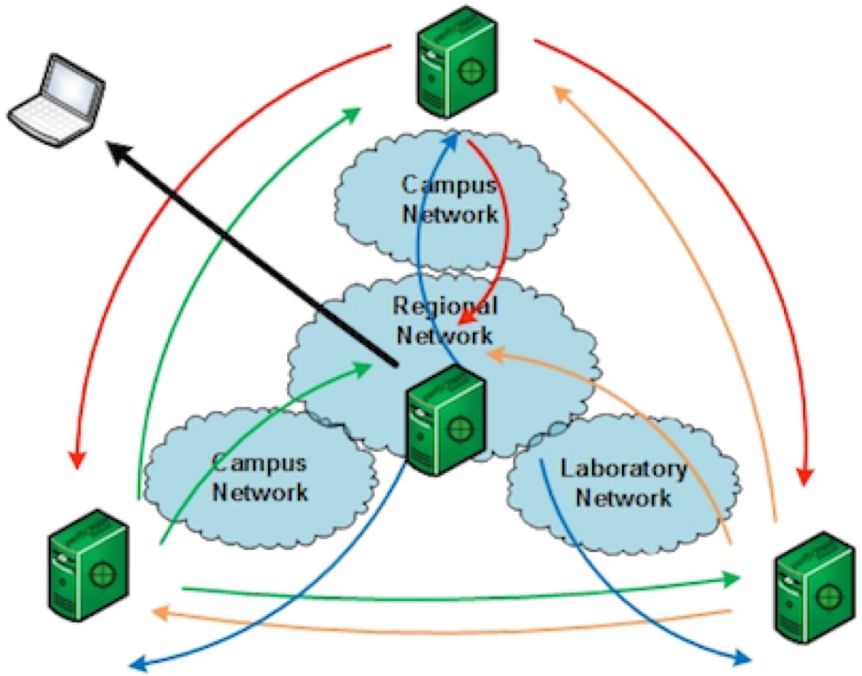
- The island setup allows a site to test against any number of the 2000+ perfSONAR nodes around the world, and store the data locally.
  - No coordination required with other sites
  - Allows a view of near horizon testing (e.g. short latency – campus, regional) and far horizon (backbone network, remote collaborators).
  - Throughput will not be as valuable when the latency is small



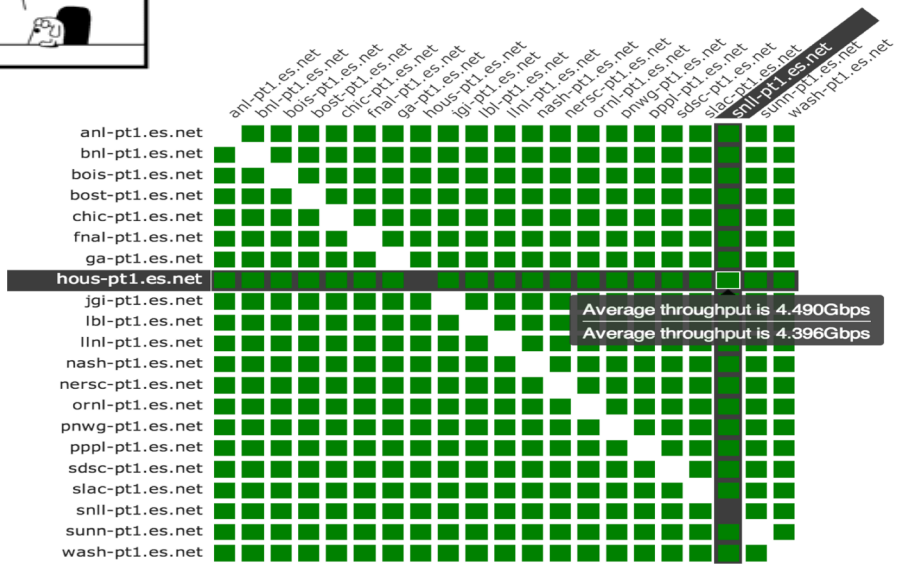
# Regular Testing - Mesh

- 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 Dashboard: http://ps-dashboard.es.net


EPOC
Engagement and Performance Operations Center

# Develop a Test Plan

- What are you going to measure?
  - Achievable bandwidth
    - 2-3 regional destinations
    - 4-8 important collaborators
    - 4-8 (*more if you are willing, especially to start*) times per day to each destination
    - 20-30 second tests within a region, longer across oceans and continents
  - Loss/Availability/Latency
    - OWAMP: ~10-20 collaborators over diverse paths
  - Interface Utilization & Errors (via SNMP)
- What are you going to do with the results?
  - NAGIOS Alerts
  - Reports to user community
  - Dashboard

EPOC
Engagement and Performance Operations Center

# perfSONAR Deployment Locations

- Critical to deploy such that you can test with useful semantics
- perfSONAR hosts allow parts 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


*http://fasterdata.es.net/performance-testing/2019-2020-data-mobility-workshop-and-exhibition/*

# CC* Data Movement Workshop and Exhibition – TCP & Performance

Jason Zurawski, Eli Dart
zurawski@es.net, dart@es.net
ESnet / Lawrence Berkeley National Laboratory

Dr. Jennifer M. Schopf
jmschopf@indiana.edu
Indiana University International Networks

*CC*/CICI PI Meeting Pre-Workshop*
*September 22nd 2019*



*National Science Foundation Award #1826994*

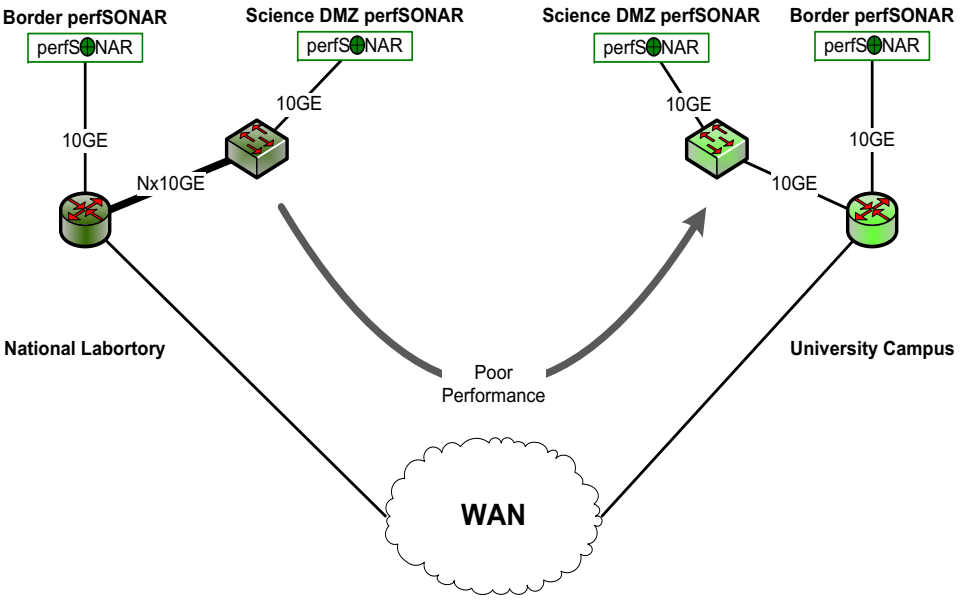
*https://epoc.global*

# WAN Test Methodology – Problem Isolation

- Segment-to-segment testing is unlikely to be 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
- 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

EPOC
Engagement and Performance Operations Center

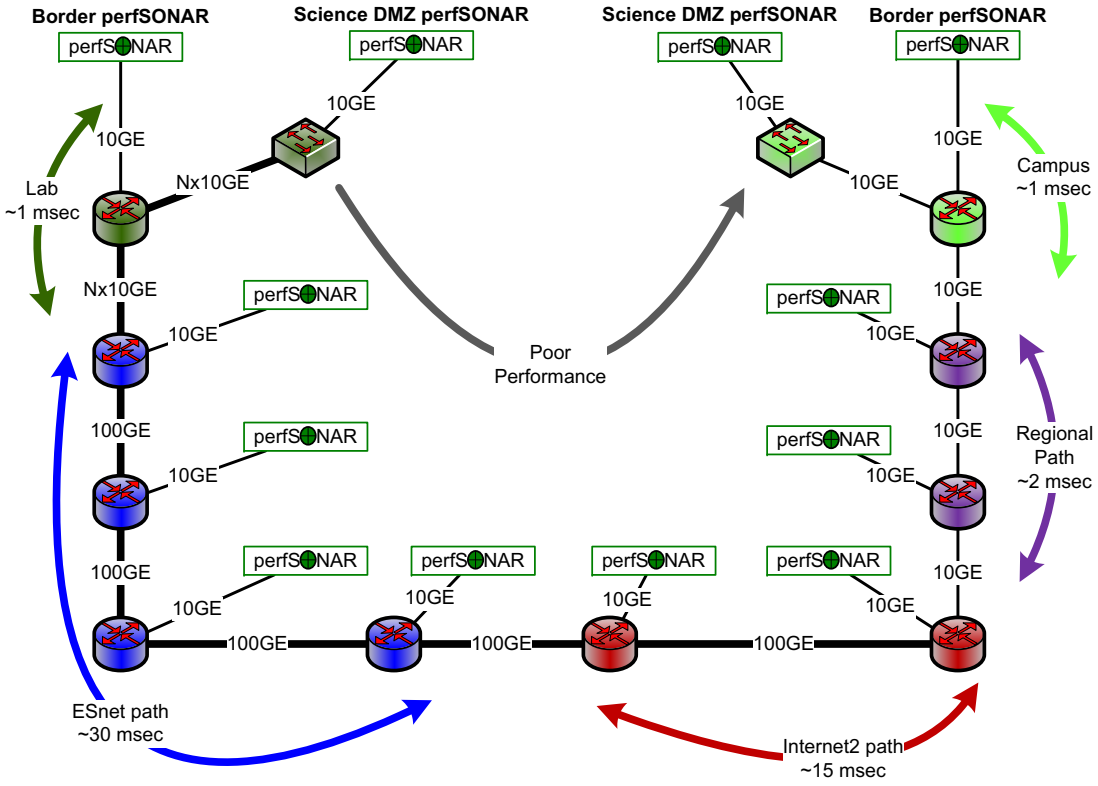
# Network Performance Troubleshooting Example




Full details and more text:
http://fasterdata.es.net/performance-testing/perfsonar/perfsonar-success-stories/long-distance-troubleshooting/

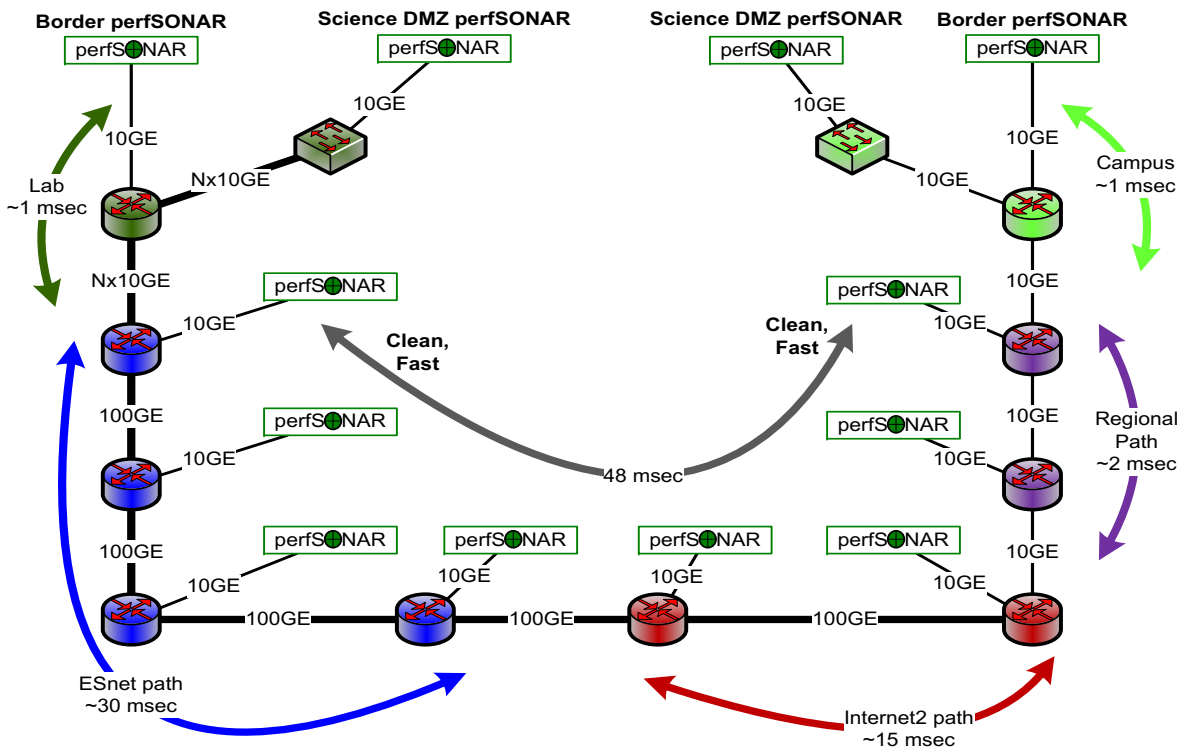
EPOC
Engagement and Performance Operations Center

# Wide Area Testing – Full Context




EPOC
Engagement and Performance Operations Center

# Wide Area Testing – Long Clean Test




EPOC
Engagement and Performance Operations Center

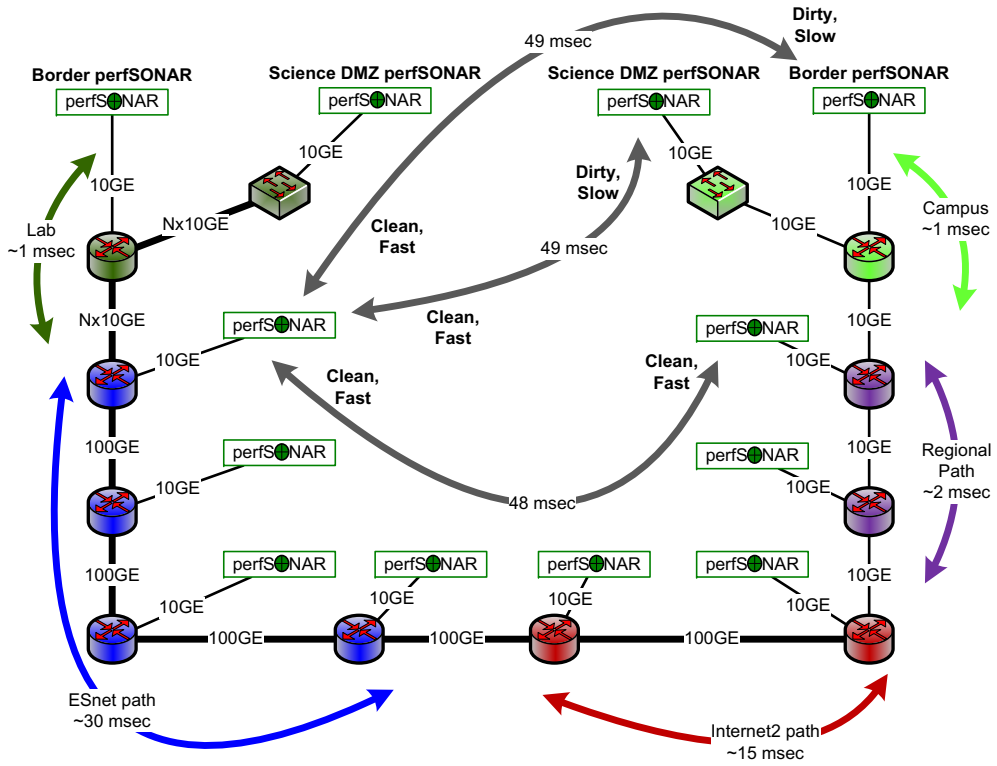
# 3rd party testing

```
pscheduler task throughput –dest x.x.x.x –source x.x.x.x
```

- From a host that has a pScheduler client, I can launch a test to two “other” hosts.
- Negotiation on timeslots and test parameters occurs between the 3 parties
- Results are returned to the host that initiated the test
- Why do we need this?
  - Helps to debug our problem space. I can sit on any host and launch a test from any of the above

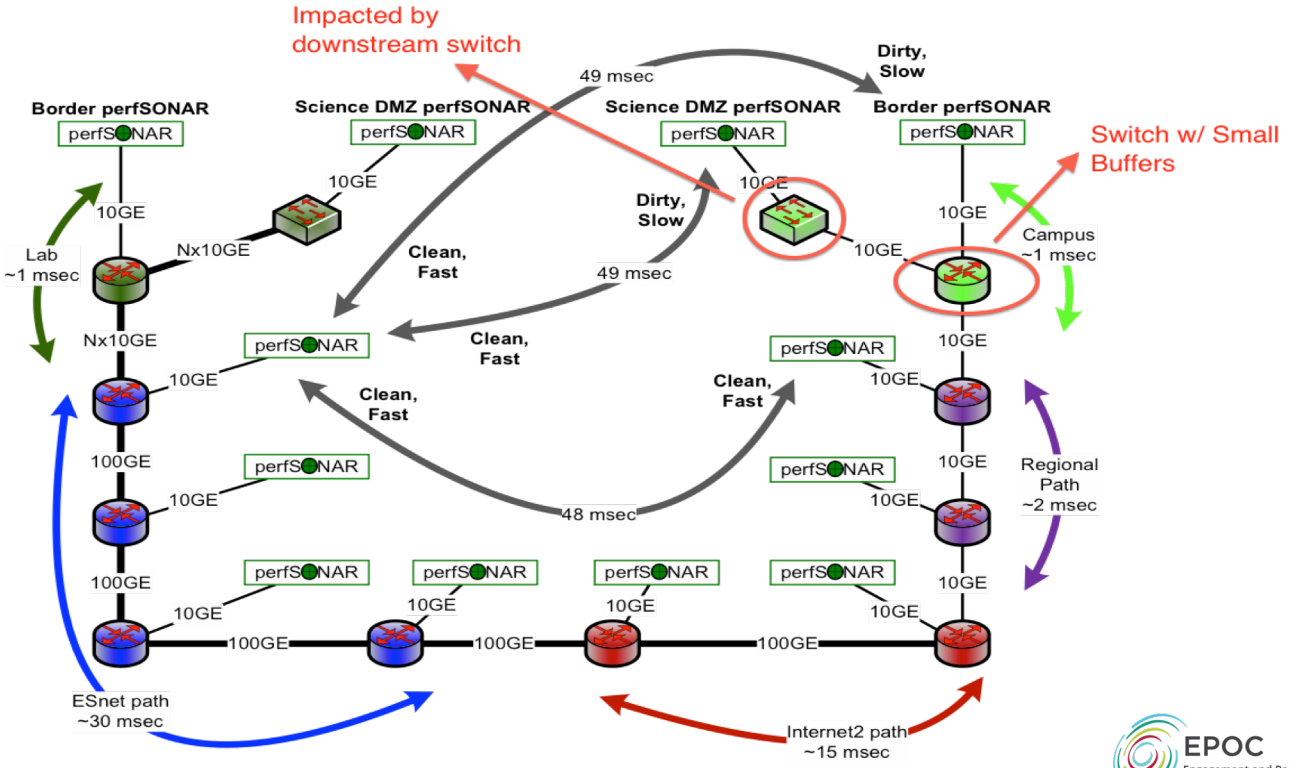
EPOC
Engagement and Performance Operations Center

# Wide Area Testing – Poorly Performing Tests Illustrate Likely Problem Areas




EPOC
Engagement and Performance Operations Center

# Likely Problem Area




EPOC
Engagement and Performance Operations Center