



# Network Management & Monitoring

## Network Delay



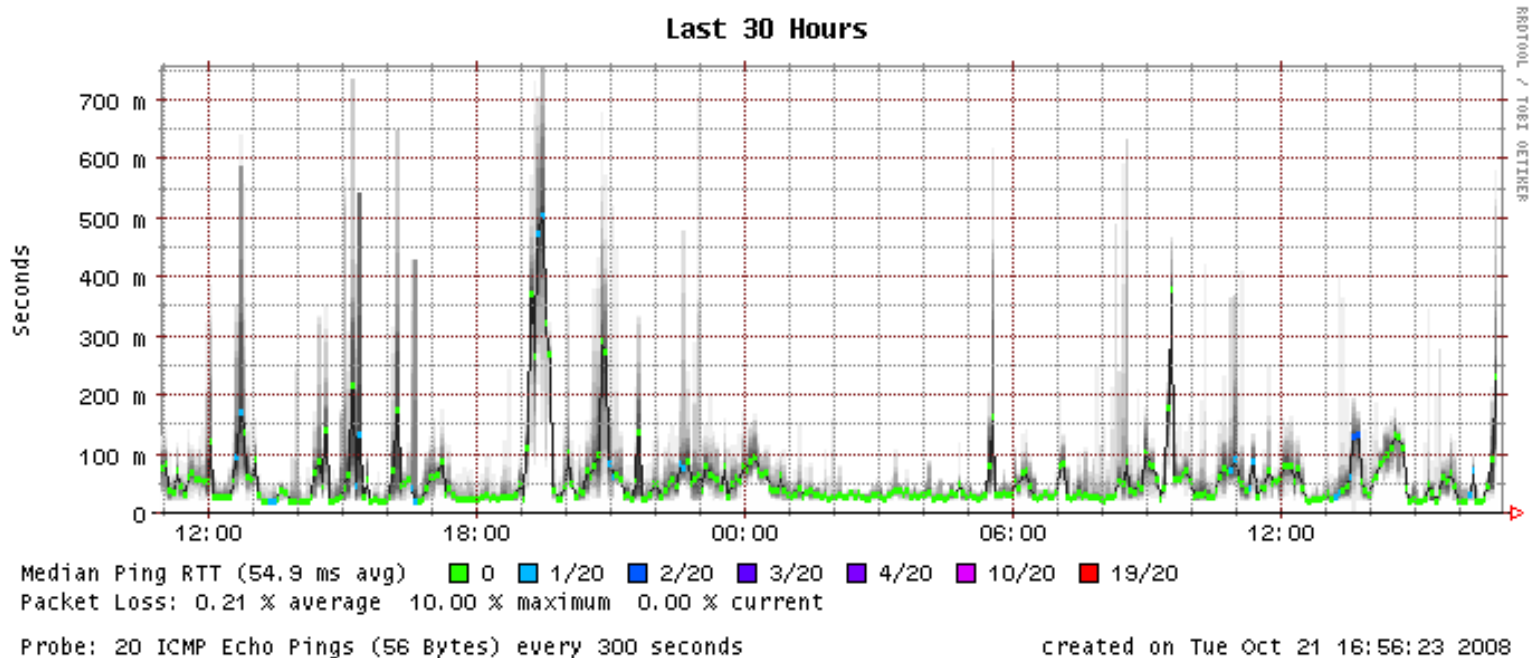
These materials are licensed under the Creative Commons *Attribution-Noncommercial 3.0 Unported* license (<http://creativecommons.org/licenses/by-nc/3.0/>)

# End-to-end Delay

**The time required to transmit a packet along its entire path**

- *Created by an application, handed over to the OS, passed to a network card (NIC), encoded, transmitted over a physical medium (copper, fibre, air), received by an intermediate device (switch, router), analyzed, retransmitted over another medium, etc.*
- The most common measurement uses *ping* for total round-trip-time (RTT).

# Historical Measurement of RTT



- What is this telling us?
- We need to understand the sources of delay

# Causes of Delay

- Processing delays
- Queuing delays
- Transmission delays
- Propagation delays

# 1. Processing Delay

Time required by intermediate routers to decide where to forward the packet, update TTL, perform header checksum calculations

(Note: most modern routers handle packet forwarding in hardware at full line rate)

plus:

Time for the far end to process the ICMP echo request and generate a response

## 2. Queuing Delay

- The time a packet is enqueued while the link is busy sending other packets
- This is a statistical function and depends on the arrival times of other packets
- QoS configurations may prioritize some types of traffic over others
- (In practice, that means multiple queues, and different packets are assigned to different queues)

# 3. Transmission Delay

The time required to push all the bits in a packet on the transmission medium in use

For  $N$ =Number of bits in packet,

$R$ =transmission rate (bits per second)

$$t = N/R$$

For example, to transmit 1500 bytes (12000 bits) using Fast Ethernet (100Mbps):

$$t = 12000/1 \times 10^8 = 0.12 \text{ milliseconds}$$

# 4. Propagation Delay

- Once a bit is 'pushed' on to the transmission medium, the time required for the bit to propagate to the other end of its physical path
  - For a given medium, the velocity of propagation is usually constant (some fraction of the speed of light)
  - The longer the path, the longer the delay
- For  $x$  = distance,  $v$  = propagation velocity

$$t = x/v$$



# Transmission vs. Propagation

Can be confusing at first

Consider this example:

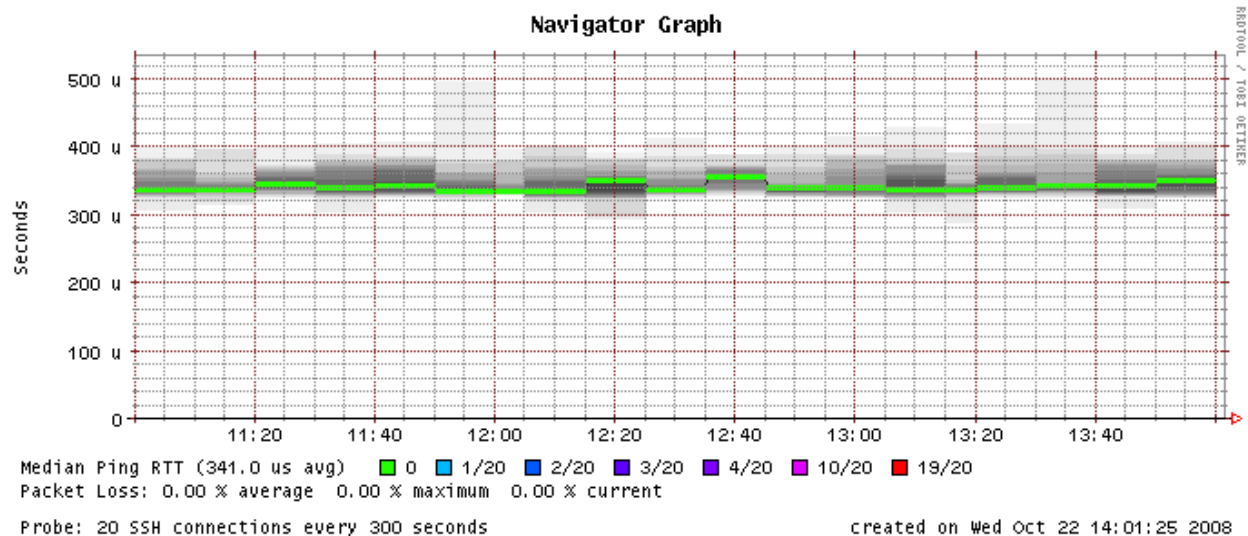
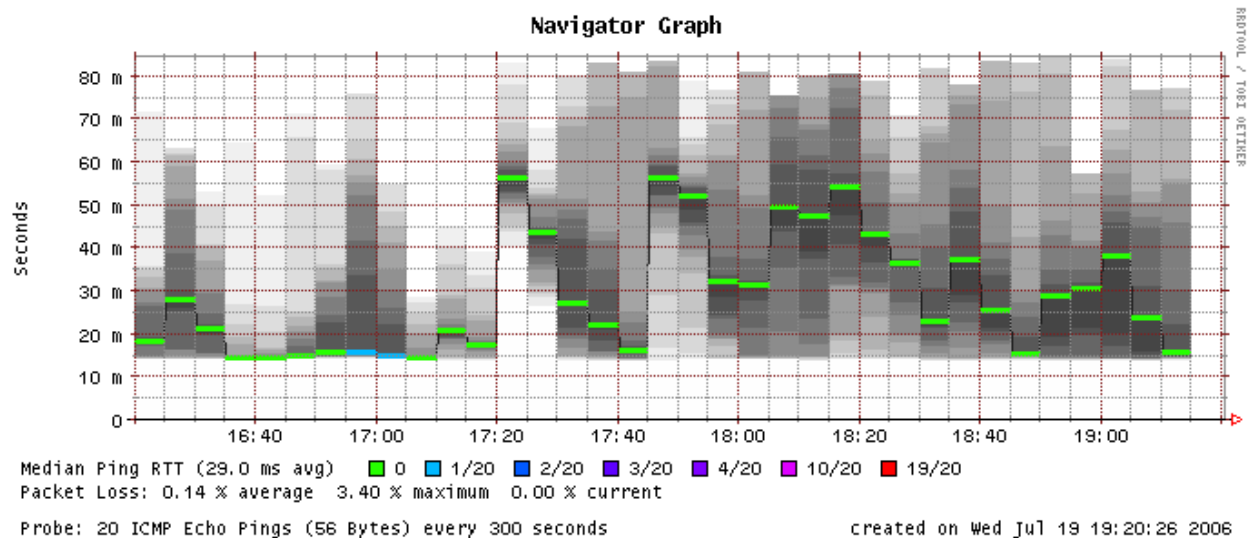
## **Two 100 Mbps circuits**

- 1 km of optic fiber
- Via satellite with a distance of 35,000 km between the base and the satellite

For two packets of the same size which will have the larger transmission delay?

Propagation delay?

# Jitter



# Questions about Jitter

- We've seen four causes of delay. Which are constant for a given path and packet size, and which are variable?
- What applications are particularly sensitive to jitter?
- Those applications may apply extra buffering to smooth out jitter – why is that additional delay a problem?

# Questions?



# Packet Loss

## Causes of packet loss:

- Transmission errors
- Queue overflow (congestion)

# 1. Transmission errors

"1" received as "0", or vice versa

- e.g. due to excess noise, poor connections, ...

Can be measured in terms of "bit error rate" (BER)

If one or more bits in a packet is corrupted, the whole packet is discarded

Retransmission of lost packets is the responsibility of higher layers (transport or application)

## 2. Queue overflow

Queues do not have infinite size

If a packet arrives when queue already full, it is dropped

Ultimately caused by insufficient capacity

However, packet loss starts to occur before the link is 100% utilized, because of random distribution of arrival times

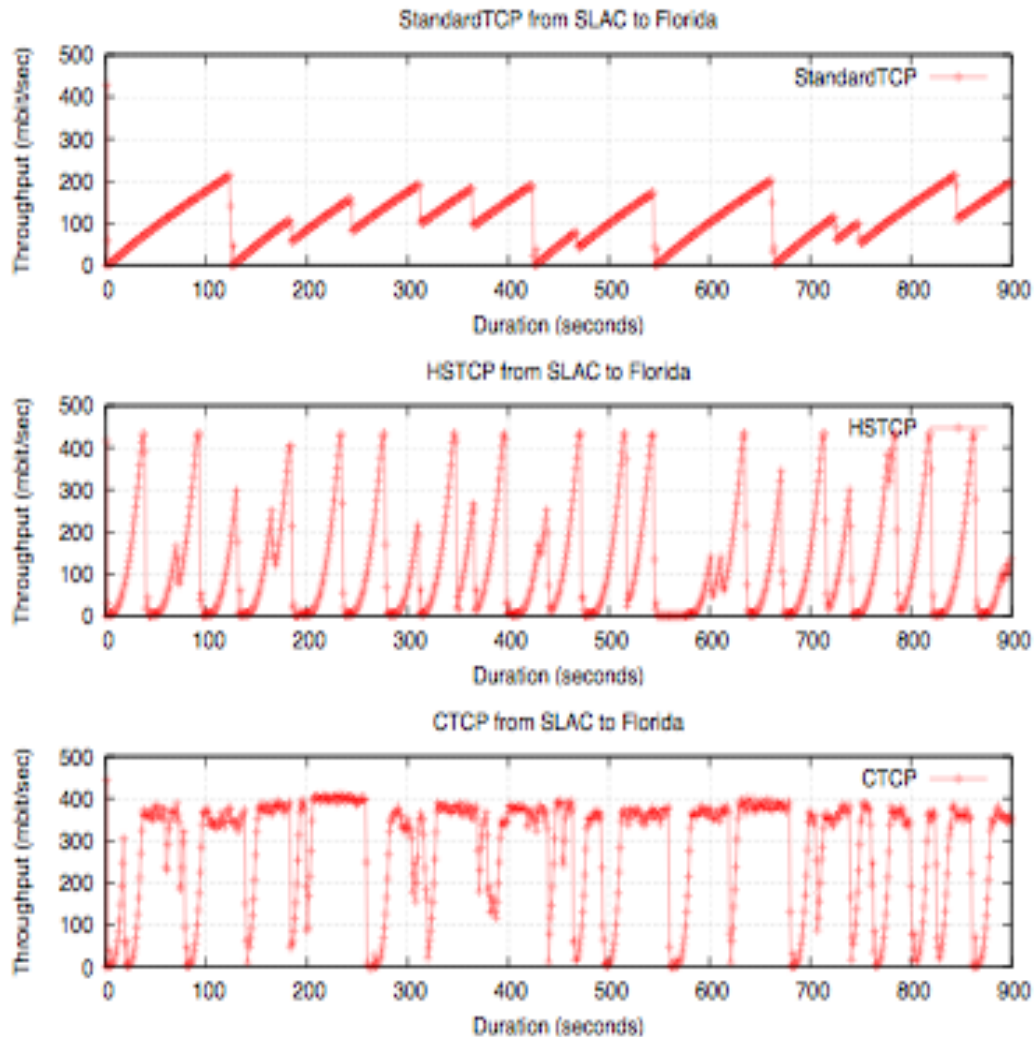
Retransmissions cause further demand and could lead to network collapse!

# TCP and Congestion Control

- TCP limits sending rate by means of a "congestion window"
- The congestion window starts small, and increases gradually while there is no packet loss
- Any detected packet loss causes the congestion window to shrink rapidly, so the sender sends more slowly



# Different TCP Congestion Control Algorithms



# Effects of TCP congestion control

- Network collapse is prevented
- "Fair sharing"
  - ✓ When there are multiple TCP streams, each one uses an approximately equal share of available bandwidth
- TCP detects congestion by observing packet loss
  - ✓ Newer TCP stacks also respond to "Explicit Congestion Notification" signals from routers: packets are marked when queues nearly full

# TCP and transmission errors

- TCP cannot tell the difference between transmission errors and queue overflows!
- Hence transmission errors cause TCP to slow down too
- Formula for maximum throughput of TCP in the presence of packet loss:

$$\frac{\text{MSS}}{\text{RTT} \cdot \sqrt{p_{\text{loss}}}}$$

# Example calculation: LAN

- MSS = 1460 bytes
- RTT = 1ms = 0.001 seconds
- Packet loss = 2% = 0.02
- $1460 / (0.001 * \sqrt{0.02})$   
 $\approx 10.3\text{MB/sec} = 82\text{ Mbps}$
- Short RTT means packet loss does not have a huge impact on local transfers

# Example calculation: WAN

- MSS = 1460 bytes
- RTT = 150ms = 0.15 seconds
- Packet loss = 0.02% = 0.0002
- $1460 / (0.15 * \sqrt{0.0002})$   
 $\approx 690\text{KB/sec} = 5.5 \text{ Mbps}$
- Loss of just *1 packet in 5,000* causes severe reduction of throughput when transferring across the Internet!

# Measurement of packet loss

- Smokeping gives a coarse measurement (20 packets every 5 minutes => 5% loss detectable, but bursts may be missed)
- For more accurate measurement you need a tool like perfsonar / owamp
  - Standard configuration sends 10 packets per second continuously
  - Can detect packet loss of 0.17% over one minute, or 0.0028% over one hour
  - Separate measurements in each direction

# Questions?

