Accuracy and Expressiveness in Adaptive Bandwidth Measurements

Prepared for the:
BEst Workshop
La Jolla, CA
December 9-10, 2003

Marc Pucci
+1 973 829 4340
marc@research.telcordia.com

Joint work with
Jim Alberi, Allen McIntosh, Tom Raleigh

An SAIC Company

What's So Hard About Bandwidth Estimation?

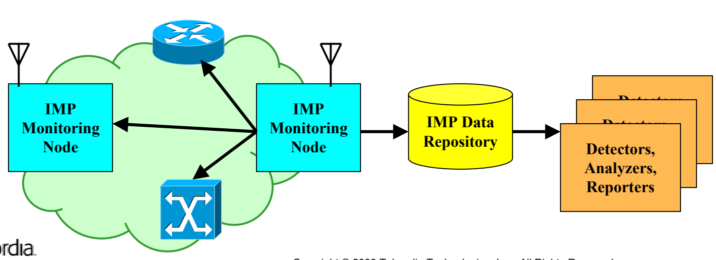
- Aside from the mathematical modeling …
- Bandwidth Estimation techniques stress the capabilities of monitoring platforms:
  - They require accurate collection and accurate emission of packets
    - Accurate collection can be achieved via kernel instrumentation and GPS clocking
    - Accurate emission requires tighter control on generation
  - More so than basic delay/loss/jitter measurements, they require feedback control between the collecting and emitting process to adapt algorithmic behavior.
  - It is also unclear that a single bandwidth estimator will be sufficient for all link speeds, all applications to be measured, and all types of cross traffic.
    - Available bandwidth of a link vs. available bandwidth for a particular traffic class (application) on a link?
    - Do we need the ability to select / switch between multiple techniques as part of an estimation process?

Telcordia Internet Monitoring Platform (IMP)

- Improving the accuracy of the measurement process
  - Better than busy-wait, but without specialized hardware
  - Low cost, easily deployed infrastructure
- Simplify the introduction of new measurement techniques
  - Common plumbing and control
  - Dynamic emission of packets
  - No need to install custom emitters across the network
- Provide a measurement platform for network monitoring systems
  - Accurate measurements for operational support
    - Management of Quality of Service
    - Identification of service-affecting conditions
  - Service Level Agreement support
    - Delay, loss, delay variance and availability are basic requirements
  - Validation of Service Level Agreement claims
    - How does an enterprise know if their agreement is being met?

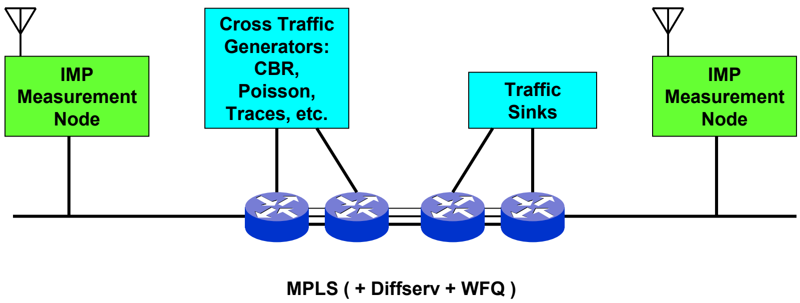
Telcordia IMP: Internet Monitoring Platform

- A Software Platform to Support Network Traffic Measurements
  - Active measurements of network characteristics
    - One-way end-to-end delay, delay variance, loss, reordering, available bandwidth
  - **Probe description language**
    - Frequency, spacing, size, contents, (e.g., a customized VoIP probe)
  - SNMP MIB and Command Line Interface Data directly from NEs
  - Extremely accurate time stamps (GPS-based) and **probe generation**
  - Commercial Off-The-Shelf components + custom software
  - Graphing, reporting, alerting, anomaly detection (wavelet, change point)



Test Bed Environment

- Multiple Cross Traffic Generators
- Multiple Bandwidth Estimators



Telcordia Technologies

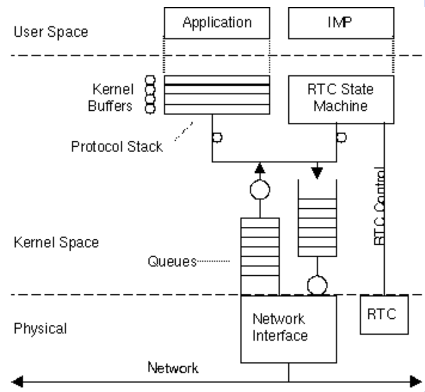
Techniques for Accurate Packet Emission

- Packet emission can be supported at several layers
  - Application Software Based
    - Busy waits and program loops
    - Easiest to deal with, but subject to high variance
  - Kernel Interrupt Driven
    - Significantly greater accuracy
    - Can be difficult to use
      - Unless a platform provides the details
    - Still interruptible
  - Exploitation of Network Interface Card characteristics
    - Even greater accuracy for higher speeds
    - No interrupts
    - Requires per-NIC driver – difficult to use
  - Modification of NIC firmware
  - Dedicated Hardware
    - More costly to deploy
    - But, you get what you pay for

Telcordia Technologies

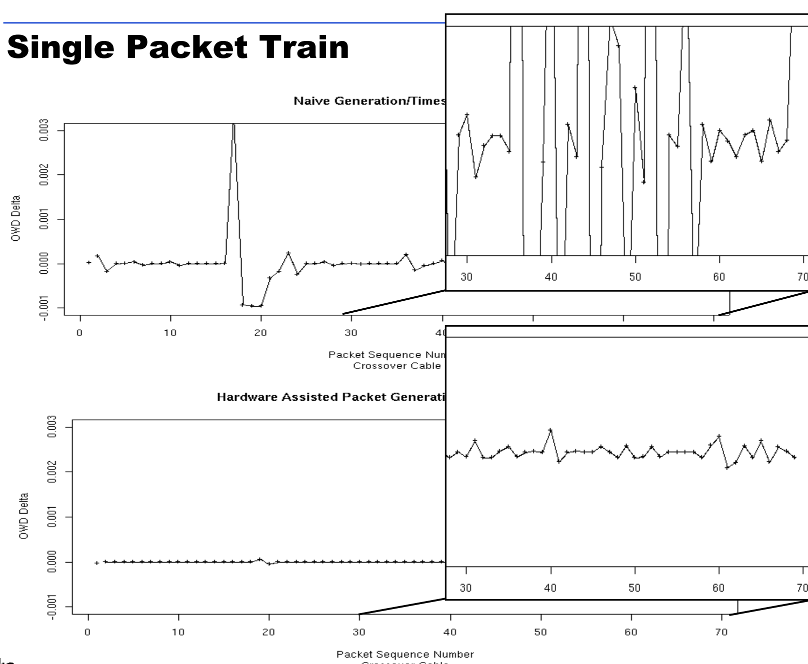
Kernel Interrupt Driven Packet Generation

- Common hardware – custom software
  - Real-time clock with 122 μsec granularity
  - Network interface card
  - No impact on standard functions
- Design components
  - RTC interrupt-driven state machine used to schedule IMP packet transmission
  - IMP packets bypass the kernel protocol stack
  - IMP packets formatted via an array of *P-Spec Packet Descriptors* copied into kernel space (/proc).
- 10 and 100 Mbit links



Telcordia Technologies

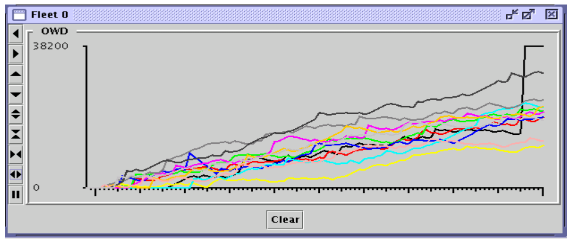
Single Packet Train

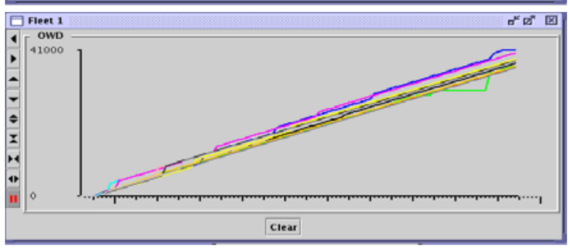


Telcordia Technologies

IMP: Measurement Plots

- Send 12 trains of 100 packets through network
- Simple Java implementation vs. kernel support
  - (Most code the same)

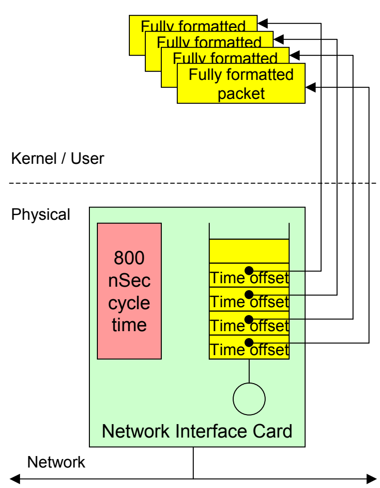




Telcordia Technologies

NIC FIFO Queue Driven Packet Generation

- Common hardware – custom driver
  - Internal NIC clock with 800 nSec cycle time access to packet data
  - Dedicated NIC, separate data path from IP stack packets
  - Still based on IMP packets formatted via an array of *P-Spec Packet Descriptors*
- Gbit links



Telcordia Technologies

Packet Probe Profile Objectives

- Should be independent of the packet generation technique
  - Whether busy wait, interrupt, NIC, dedicated HW
- Allow the receiver to control the behavior of the packet sender.
- Improve the expressiveness of the measurement request.
  - Short but descriptive specifications
- Allow a single packet emitter to generate multiple forms of packet probe profiles.
  - Minimizes the deployment / update problem
- A programmable specification, but without the overhead and variance caused by program execution.
- The mechanism should be small and efficient enough to be kernel resident.

P-SPEC Language

Packet variables	p0 - pN	Sets the values of the first N words of a packet
Assignment operators	=, *=, etc.	Binary operators for generating simple progressions such as sequence numbers or varying delays or sizes
Synchronization commands	wait, sync, delay,	Creates a synchronization point in the stream for communicating back to the measurement collector
Synchronization variables	arg0 - argN	Control arguments set by “start” or “continue” commands from the collector
Packet descriptor variables	size, gap, tos,	Define packet constraints such as size and gap before next packet
Packet generator	packet	Forces a Packet Descriptor to be generated with the current set of values for content, size, etc.
Looping control	N:(…)	Repeat internal instructions N times; for generating repeated subsequences of packets
General purpose variables	a - z	General purpose variables for processing, but which have no special meaning to the packet generator.

- The Packet statement does not emit a packet, but creates a packet descriptor for later use by the kernel emitter.
- Synchronization statements are used to flush accumulated packet descriptors and possibly await further control information.

P-SPEC – Specification Language for Packet Sequences

- P-Spec gives control over
  - Packet content
  - Packet size
  - Inter-packet gap
  - Inter-packet-group delay
  - Synchronization data sent back to local requester
  - Control arguments sent from local requester to remote emitter
- Example P-Spec for an Adaptive Dispersion Technique packet sequence (pathload)
- The probe sends fleets of packets, with the packet size, inter packet gap and inter burst delay adjusted by the collecting process according to the analysis algorithm.

```
fleets_per_burst=3              probe parameters
packets_per_fleet=20
burst=3
*:(                             infinite generation
   WAIT(fleet++)                send fleet index to collector
                                     and await argN response
   gap=arg0                     reset new values for gap,
   size=arg1                         size and delay
   delay=arg2

   p1=0
   p3=packets_per_fleet
   fleets_per_burst:(           generate multiple packet fleets
        p0=burst                packet word 0: burst number
        p2=0                    packet word 2: packet in fleet
        packets_per_fleet:(
             PACKET()
             p2+=1
        )
        p1+=1                   packet word 1: fleet in burst
        DELAY(delay)            inter-fleet delay
   )
)
```


Platform Architecture