Advanced Features of Network Emulators for Sync Applications

Chip Webb, CTO – Anue Systems

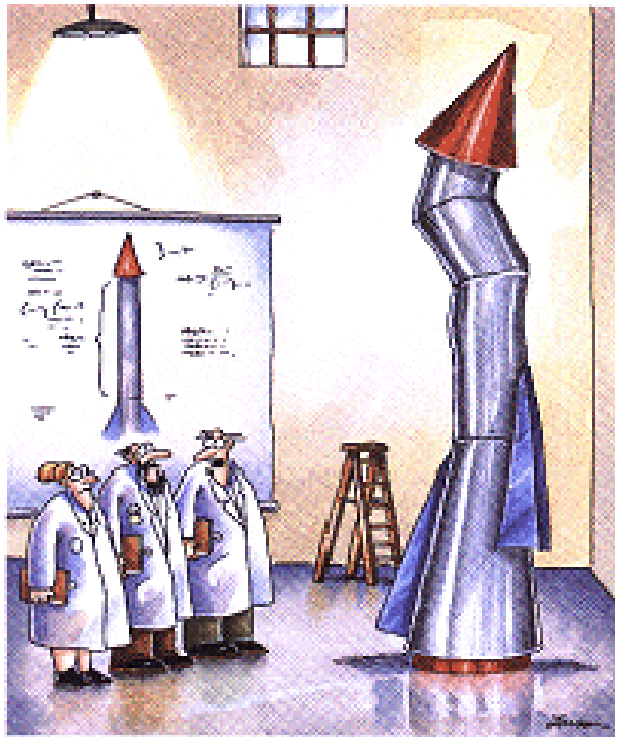
ITSF 2007

www.anuesystems.com

Timing Over Packet. *Issues*

Issues:

- Behavior of packet networks is still poorly understood.
- Service providers are reluctant to provide real-world, worst-case performance data for their networks (and always will be).
- Sync performance may depend on network characteristics that a provider either doesn't control or is unaware of.
- Sync performance may depend on “secret sauce” in a vendor’s CES Inter-Working Function.



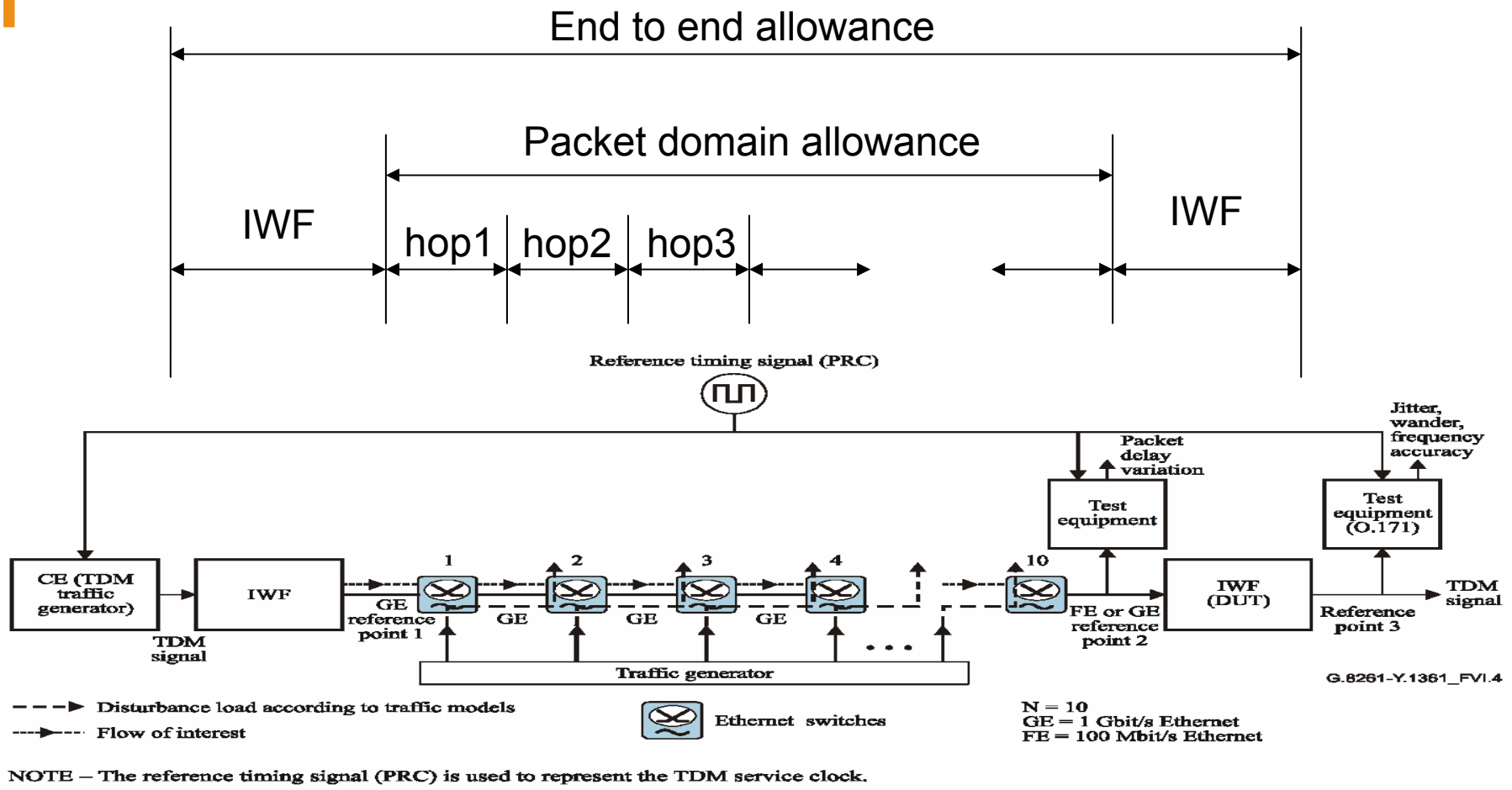
Testing is crucial to Sync Development

- Test new Sync techniques & algorithms
  - Find the breaking point of a system
  - Determine robustness
  - Must be flexible and realistic
  - Directly control parameters of interest
- Build an impairment (loss/wander) budget
  - Impact on application performance
  - Build a link budget, hop-by-hop
  - How much is allowable at each hop?
- Perform interoperability tests
  - Go/No-Go tests against predefined limits
  - Does vendor X meet performance spec?
  - Does vendor X work with vendor Y?

Develop Hop-by-Hop Wander Budget

- System designers are left to specify their own idea of "worst-case."
- If it is too conservative (pessimistic)
  - Leads to over designed systems that cost too much
  - And may unnecessarily delay deployment (and revenue)
- If it is too aggressive (optimistic)
  - System designers may miss key aspects of network behavior.
  - Deployed systems may not work as expected
  - Fiasco.

Packet domain wander budget



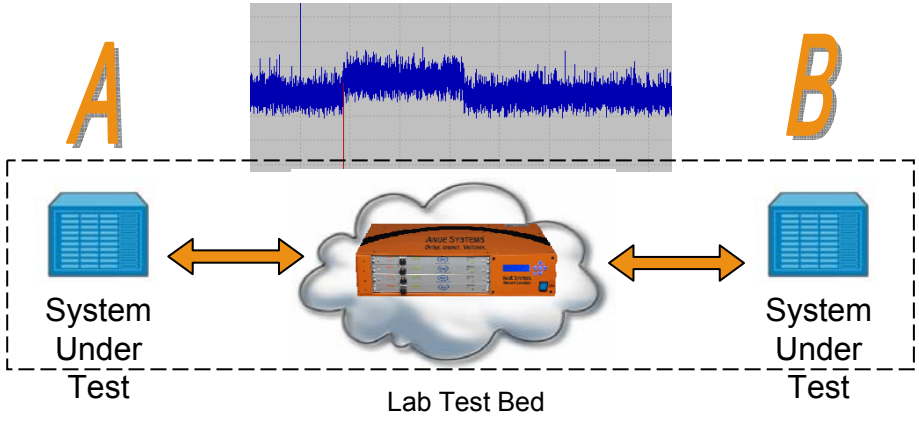
NOTE – The reference timing signal (PRC) is used to represent the TDM service clock.

How to test?

- Use a high-performance network emulator
  - Deterministic & Repeatable
  - But how to control it?
- But not all network emulators are suitable
  - Must have high precision
  - Must have realistic worst-case scenarios for testing
- There isn't one universal idea of what that worst-case scenario should be.
- Until then, network emulators should provide a general purpose facility to control packet delay from user-defined table

Network Impairment Playback

- Control the emulator on a packet-by-packet basis
- For long test durations
- Need to import and play back *custom network scenarios*
- This boils down to table-based impairment control
  - One table for A->B and another for B->A.
- Each table entry sets the impairment for one packet
- Fully dynamic control with ns precision and repeatability
- Extremely large tables (e.g. model diurnal variation)



Constructing realistic test cases

- Two sources of realistic performance data:
- Actual network measurements
  - Measure packet delays between “A” and “B”
  - Record for days or weeks. Save to a file
  - Replay these scenarios in the lab, over and over
  - Can magnify impairment to mimic worst-case (margining)
  - Can focus replay just on significant network “events”
- An analytical model
  - Use above measurements to formulate a “*golden*” model of network behavior
  - Can be abstract and parameterized
  - Run the model for various scenarios of interest
  - E.g. 10 hops with 10 to 30% load and QoS.

Applications

- Test new technologies, protocols or applications with virtually limitless control for:
  - Any dynamic or bursty network scenario
  - Extremely high frequency jitter or low frequency wander can be easily simulated
  - Allows custom statistical impairment models (ex: load dependent jitter)
- Stress *Timing over Packet* (PTP, NTP, RTP) clock recovery algorithms with maximum precision of delay variation – *per packet delay control*
- Develop a library of test scenarios

Conclusion

- Table-driven "*Network Impairment Playback*" is a key feature for testing Sync applications.
- Allows the user to play back very long impairment scenarios with packet-by-packet precision.
- Gives complete control to the user
- Scenarios can be based on
  - Actual network measurements or
  - Synthetic sequences from a "golden" model.

Questions?

- Contact me!
  - Chip Webb
    Anue Systems
    9111 Jollyville Rd
    Suite 100
    cwebb@anuesystems.com (*preferred*)
    +1 (512) 527-0453 x102

Thank You!