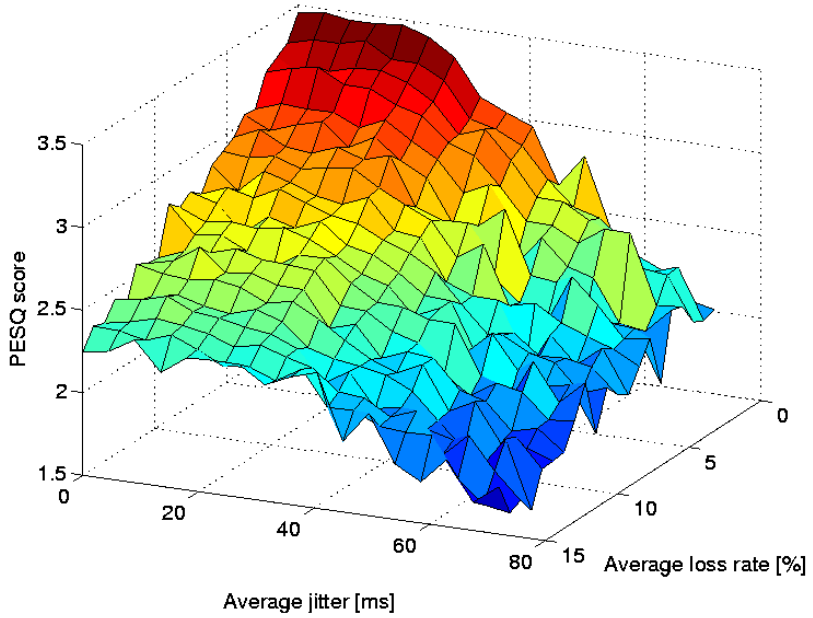
Network Emulation

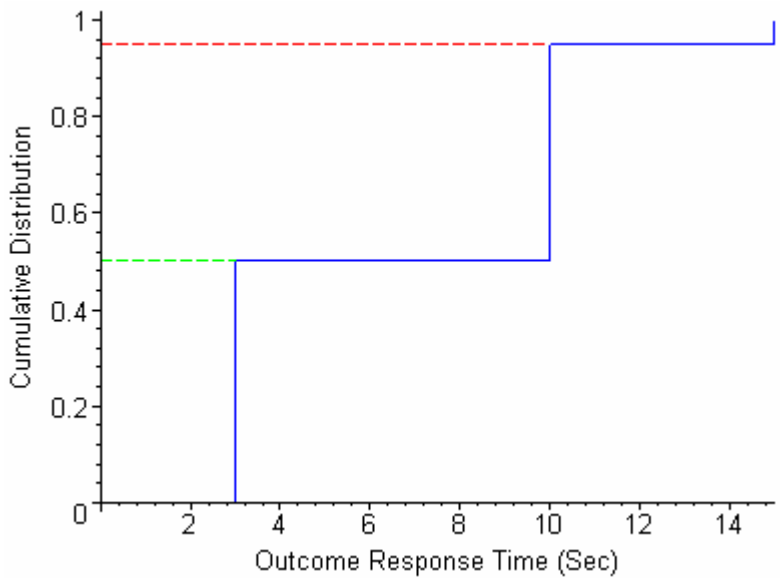
Mihai Ivanovici
Dr. Razvan Beuran
Dr. Neil Davies

Quality of Service

- What is “quality” for a network / application
  - Quality is “finite”
- How is it
  - Shared (between different traffic flows)
  - Managed (at the contention points)
  - Delivered (to the application / user)
  - Perceived (by the user)
- View points
  - Local view (at a point in network)
  - Global view (end to end)

Application outcomes





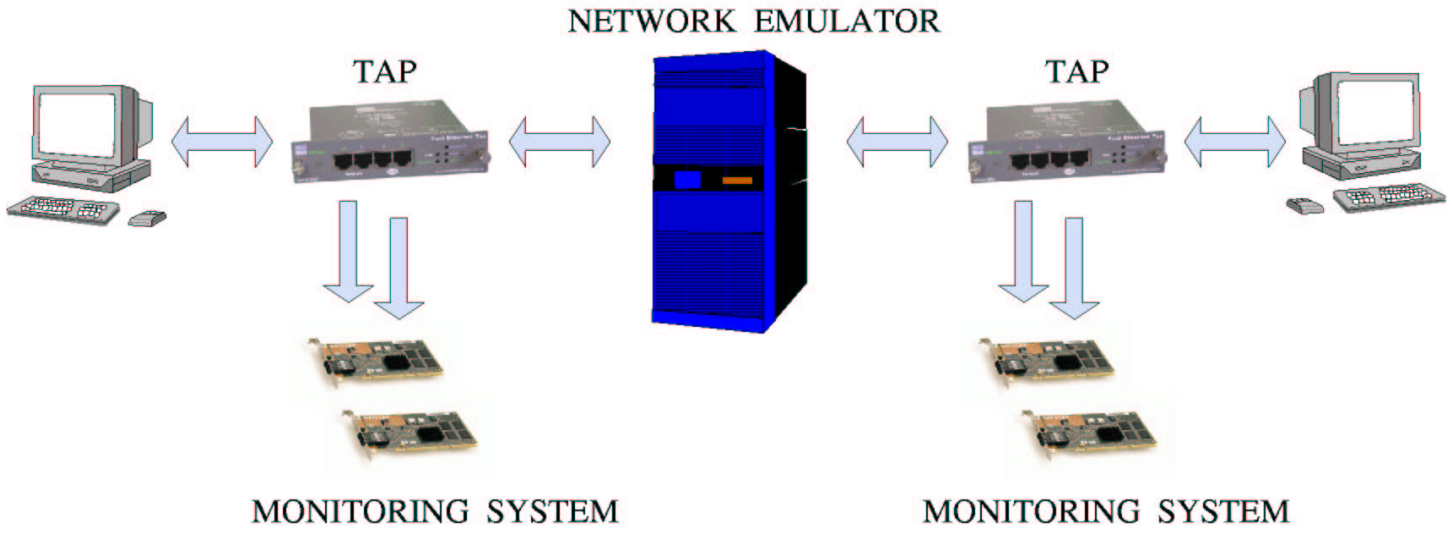
The judgment metric: “application outcomes”

Needed elements

- Model of interaction between application and delivered quality from network
- Environment for evaluation of existing applications
- Capture process to apply to existing / proposed developments

→ Combine to create a well-defined methodology for assessing application performance

Assessing application performance



- In parallel
  - Measure the network QoS parameters
  - Assess the UPQ for the application under test

Current emulators

- Do exist
  - Mainly software based
- Packet by packet systems
  - Independent loss and delay applied to packets – unrealistic behaviour
  - False packet reordering
  - No intra-stream contention modelled

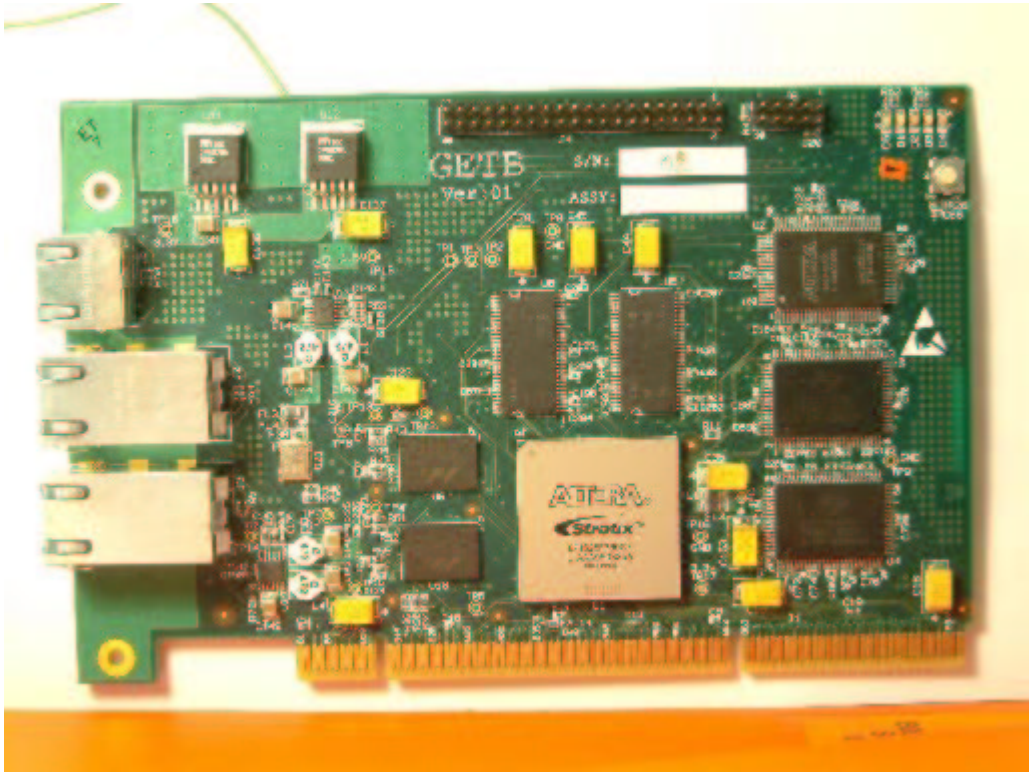
Why another emulator?

- More realistic scenarios
  - Intra-stream and inter-stream contention
    - Correlated loss and delay, natural induced jitter
  - Phase / mode changes in network
    - Topology or environment changes (e.g. wireless)
- More flexibility and control on the degradation models

Why another emulator? (II)

- Safety critical / mission critical viewpoint
  - How the systems operate under various network conditions
    - It's not just about "normal" situations
  - How and when applications fail
- Speeds up to 1 Gbps

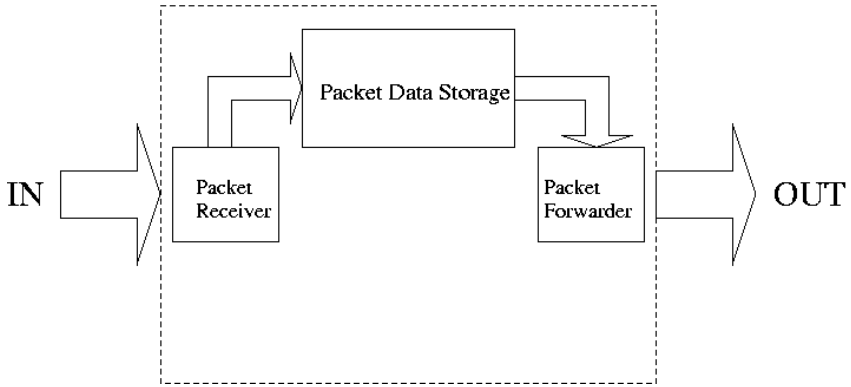
The hardware platform



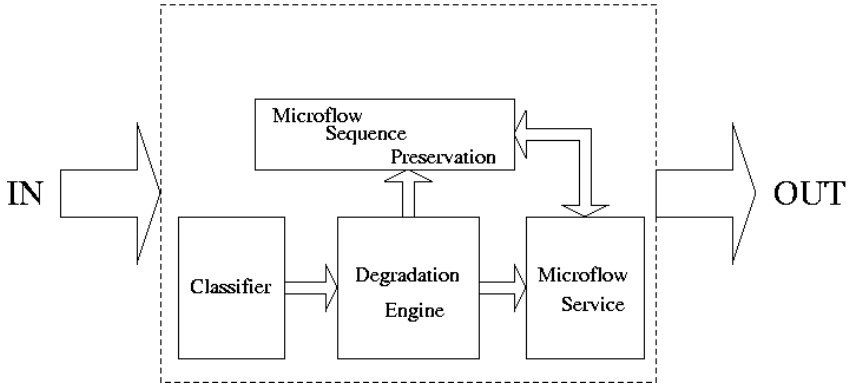
How we do it

The packet processor

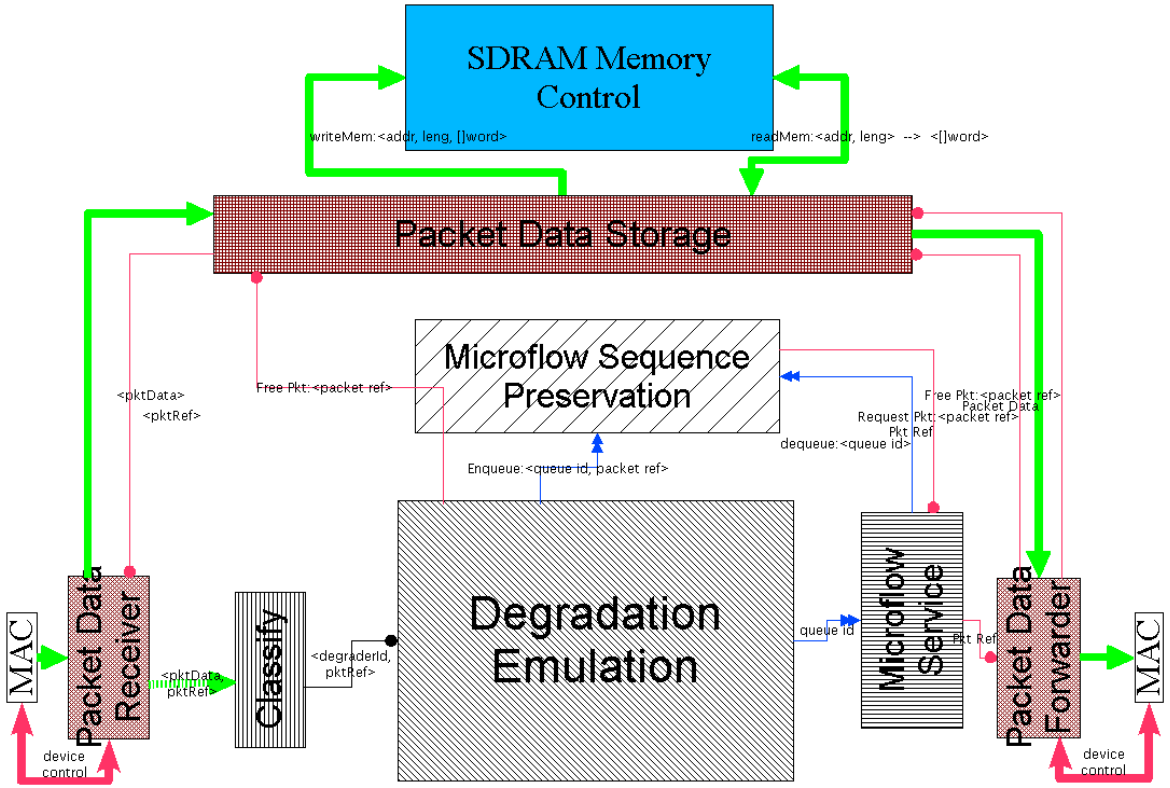
The packet path



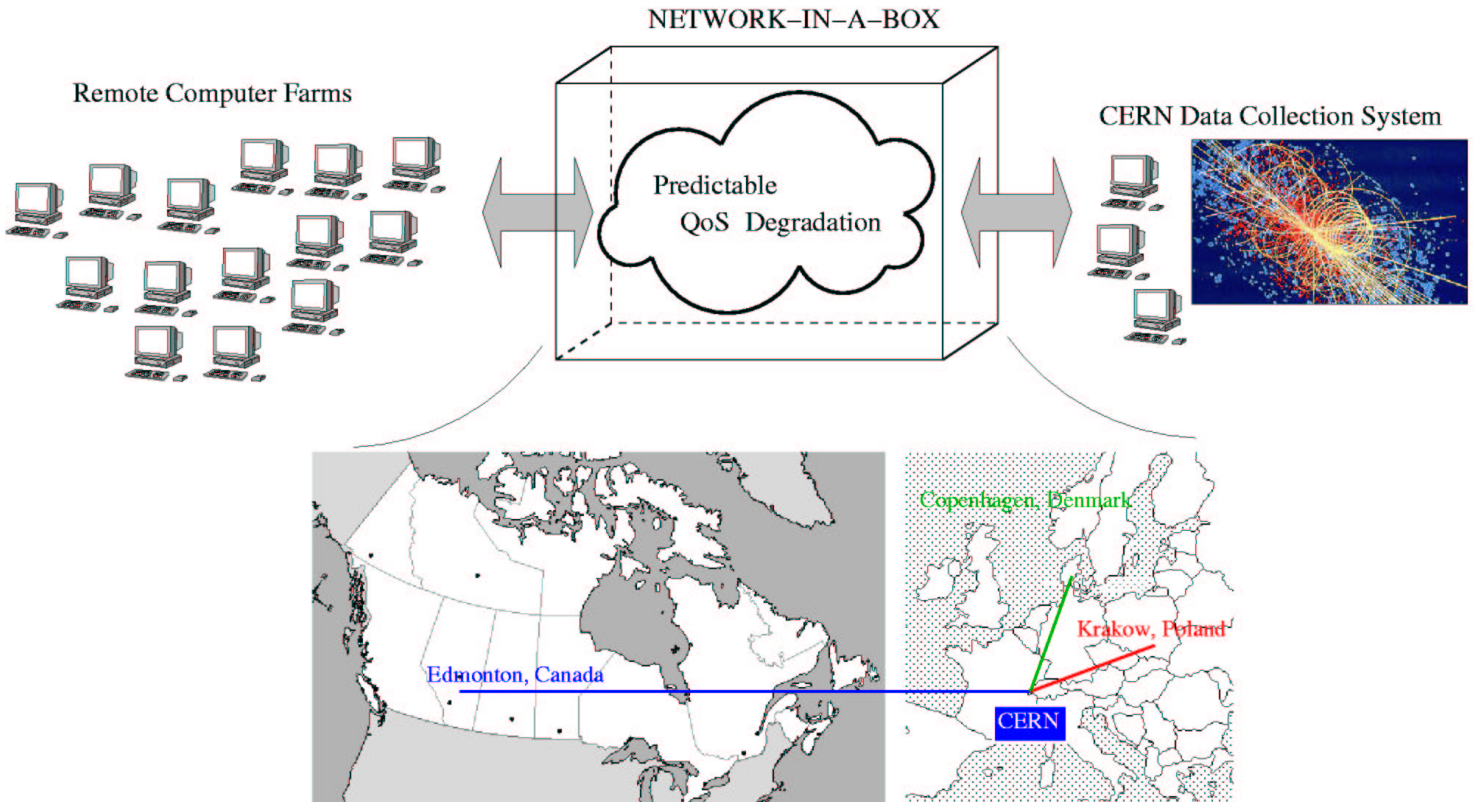
The control path



The architecture

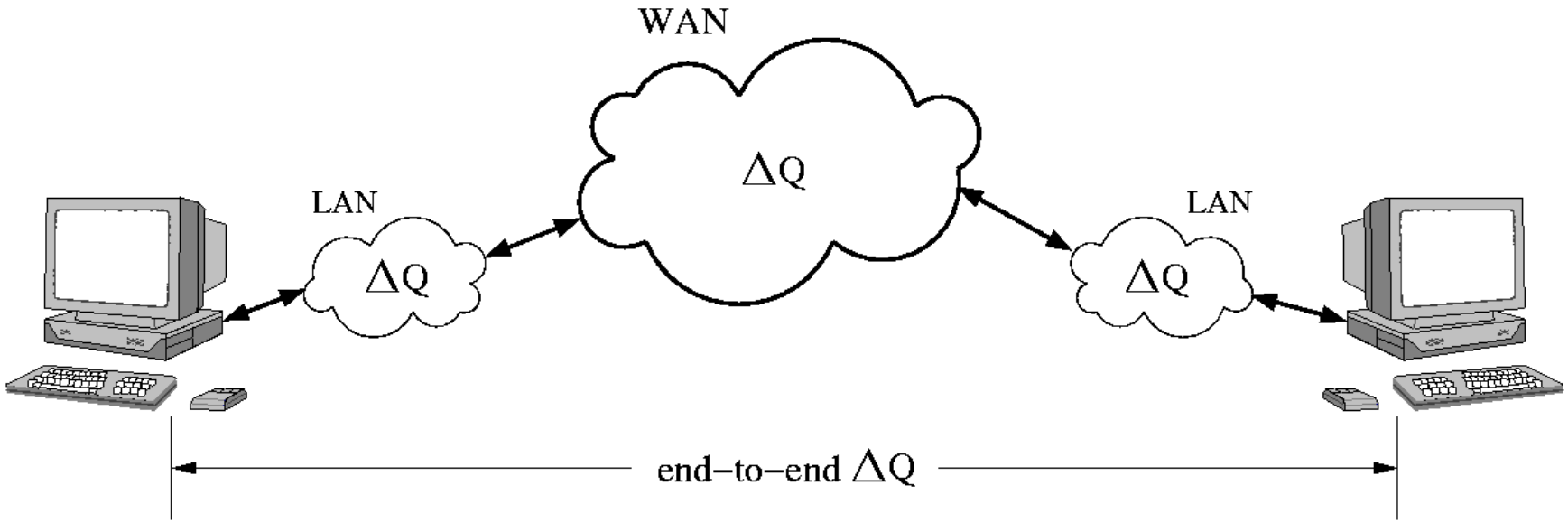


Network in a box



- Predictable = reproducible network behaviour and application errors
- Regression testing

The “ΔQ” concept



Total degradation in the network = aggregation of the degradation induced by each sub-network and network element on the way

Intra-stream contention

- Competition for resources
  - Sharing the connection → THROUGHPUT
  - Entering the queues → LOSS
  - Leaving the queues → DELAY
- How applications react to quality degradation
- Model the effects of application behaviour on quality degradation
  - e.g. burst loss behaviour on TCP/IP

Inter-stream contention

- The total amount of degradation is shared between different streams
  - By use of scheduling mechanisms (e.g. SP, WRR)
- Differentially treat the traffic to achieve the best application outcomes

First versions

- Fixed delay + one queue
  - Constant service time
  - Bandwidth limitation
- Fixed delay + one queue
  - The effect of other traffic flows sharing the same network/path emulated as server vacations

The next step

- More sophisticated network models for the “server with vacations”
- Aggregation of simple models of “queues” and “wires”

Conclusions

- We propose a methodology for assessing application performance
- Network emulation
  - Allows a hybrid test technique
  - Combines the advantages of simulation and real application/networks testing