IPCOMM2006
September 25-27 • Gaylord Opryland • Nashville, TN

The Hidden Cost of QoS: The Myth of Scarcity

Loki Jörgenson
Chief Scientist
Apparent Networks

Wednesday, September 27 | 10:15am – 11:15am

What Will Be Covered

Some forms of QoS offer selective performance degradation, guaranteeing the service of one application at the expense of others. While boosting bandwidth can lessen the need for QoS, it comes with a price tag that makes the promise of QoS seem more attractive. But QoS has big ticket impact on implementation, maintenance and support costs, while throwing people and time at break-fix issues can make QoS approaches less cost-effective than growing network capacity. Find out which is the best approach for your company’s converged network.

Outline

- QoS as deliberate performance degradation – bandwidth as an option
- QoS doesn’t fix broken networks
- Proper QoS needs to be app-specific
- Hidden costs
  - **Complexity of implementation**
  - **Support and truck rolls**
  - **Equipment costs**
- Not just technological – also economic

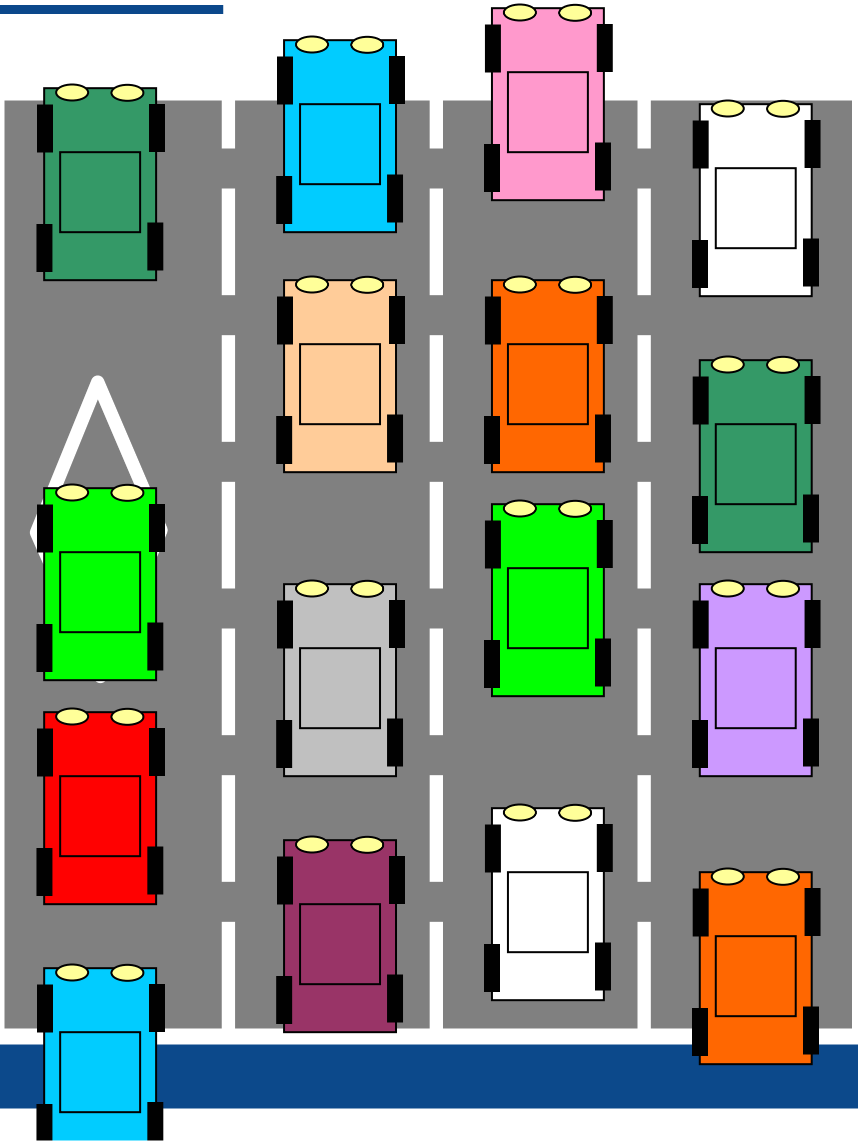
What is QoS?

“QoS isn't about treating things better (it doesn't create network capacity)---it's about treating things non-uniformly. ”
(Internet2 – QoS working group)

“The reason for QoS on the LAN side is due to buffering, **not lack of bandwidth**. For this reason, QoS tools are required to manage these buffers to minimize loss, delay, and delay variation.”
(Cisco)

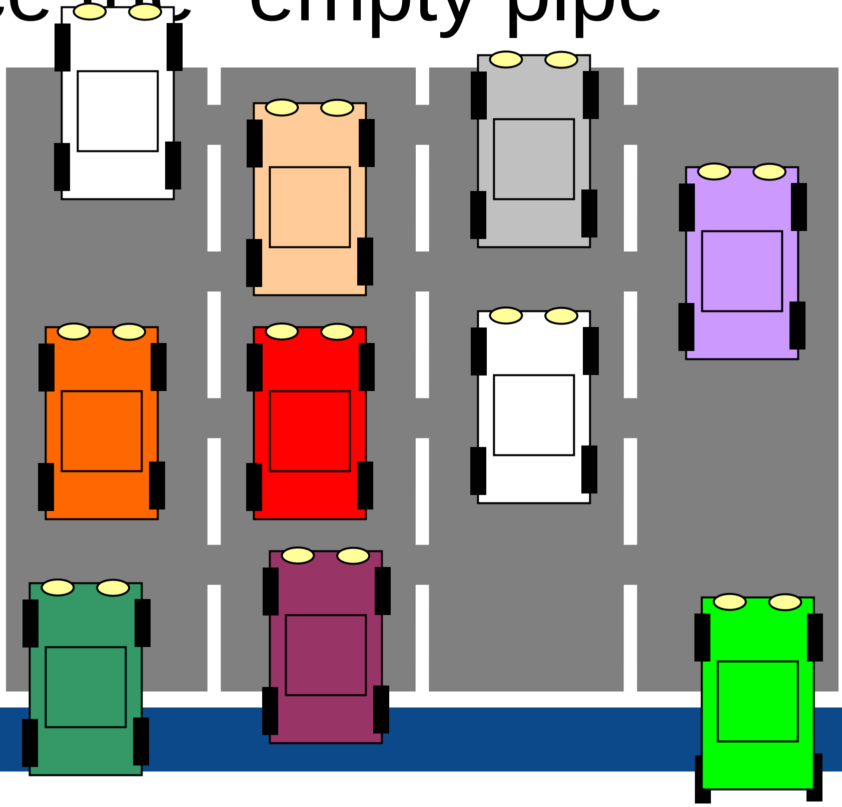
QoS as HOV

- Simple model of QoS - reserved capacity
- Experience no better than the empty road



Alternative: Over-Provisioning

- Over-provisioning avoids congestion effects by eliminating them
- All packets experience the “empty pipe”



Over-provisioning

Pros

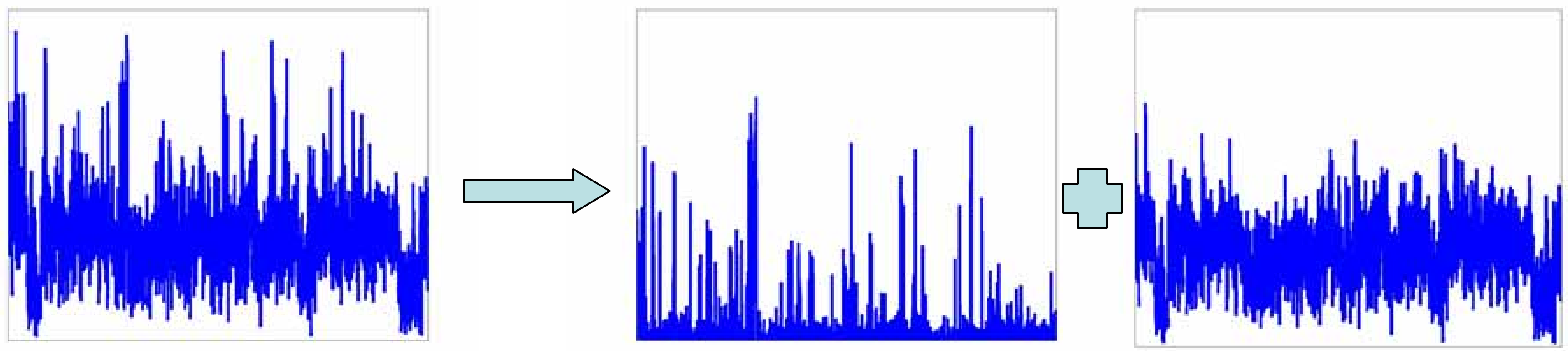
- Simple
- Scalable
- Always increasing

Cons

- Explicit price tag on bandwidth
- Other bottlenecks
- Burstiness

Bursty Traffic: Alpha and Beta

Alpha traffic is caused by **large file transmissions over high bandwidth links** and is extremely bursty (non-Gaussian). Beta traffic (residual without alpha) is Gaussian, LRD, and shares the same fractal scaling exponent as the aggregate traffic. Beta traffic is caused by both small and large file transmissions over low bandwidth links.



www-ece.rice.edu/~shri/alphabeta

Lessons Learned from Internet2

Guy Almes**, chief engineer Abilene**

"The general consensus is that it's easier to fix a performance problem by host tuning and healthy provisioning rather than reserving. But it's understood that this may change over time. [...] For example, of the many performance problems being reported by users, very few are problems that would have been solved by QoS if we'd have had it."

Lessons Learned from Internet2

- Internet2 review of QoS
  - **Gave up on QoS working group in 2002**
  - **Focused on end-to-end performance working group instead**
  - **QoS group still not reactivated**

- **Recommendation** → Ensure network health

QoS - http://abilene.internet2.edu/services/qos.html

End-to-End Initiative - http://e2epi.internet2.edu/

Outline

- QoS as deliberate performance degradation – bandwidth as an option
- QoS doesn't fix broken networks
- Proper QoS needs to be app-specific
- Hidden costs
  - **Complexity of implementation**
  - **Support and truck rolls**
  - **Equipment costs**
- Not just technological – also economic

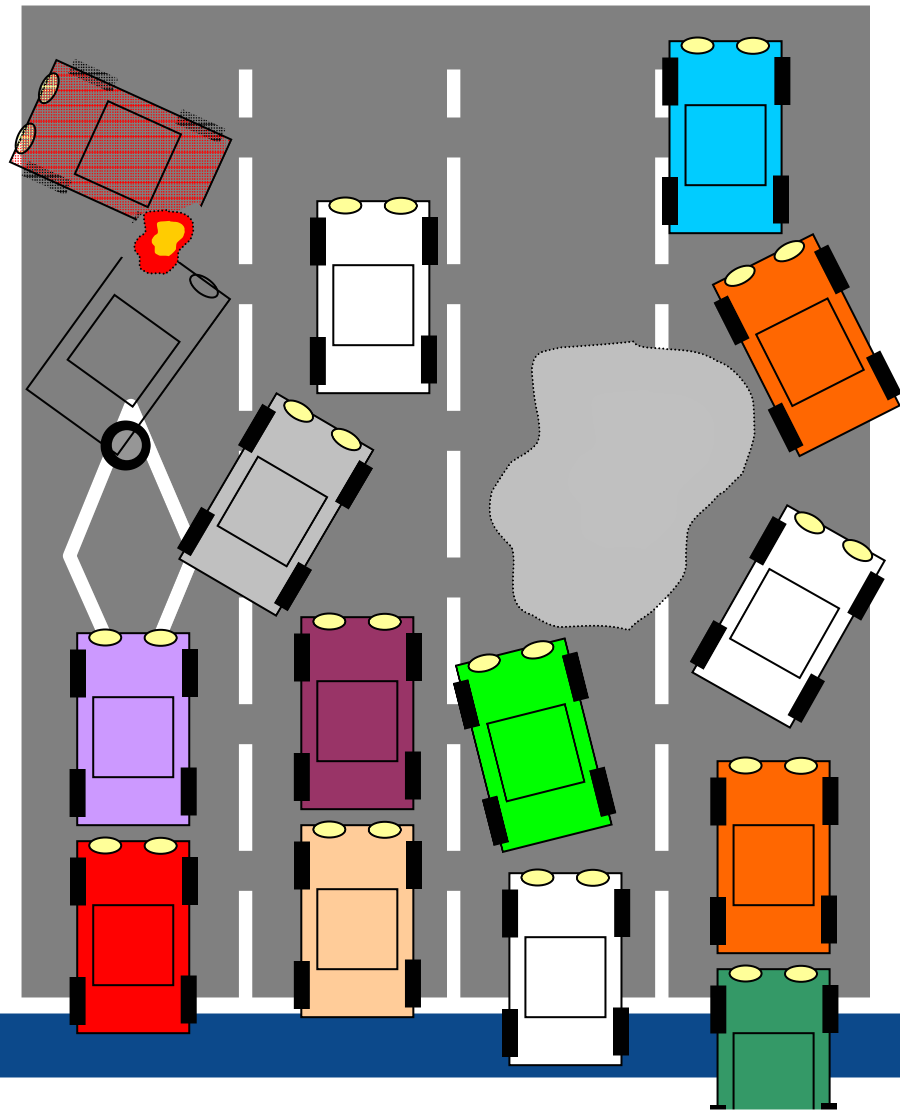
Fix the Network First

Three Steps to Performance

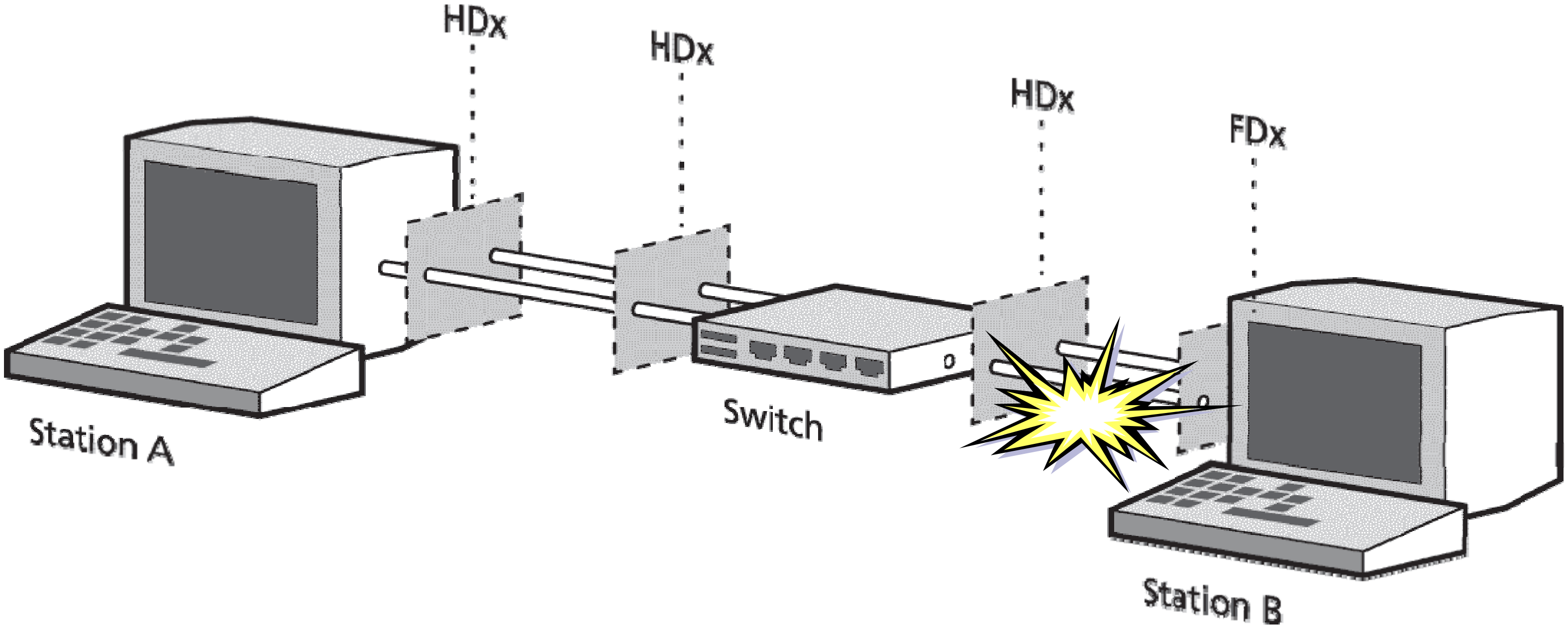
1. **Clean the network**
   a) Pre-deployment
   b) Monitoring
2. **Model traffic**
   a) Application requirements for QoS/SLA
   b) Monitoring for application performance
3. **Deploy QoS**

Performance Degradations

- QoS can't fix degradations
- Find, diagnose and repair
- Continuously monitor for recurring effects



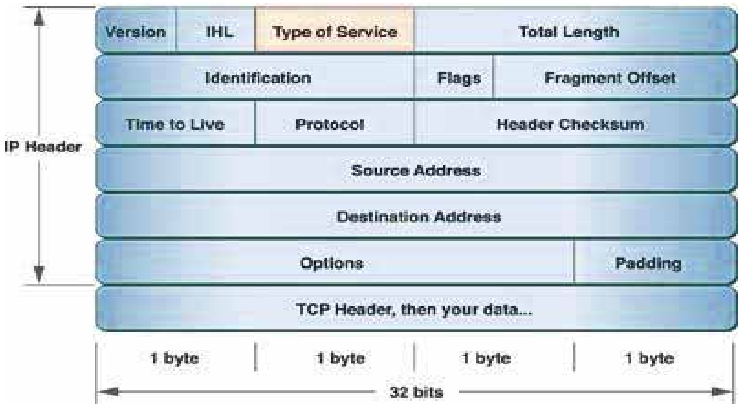
Duplex Conflict on Switch



So, what is '**auto-negotiation**'???

QoS and the Duplex Conflict

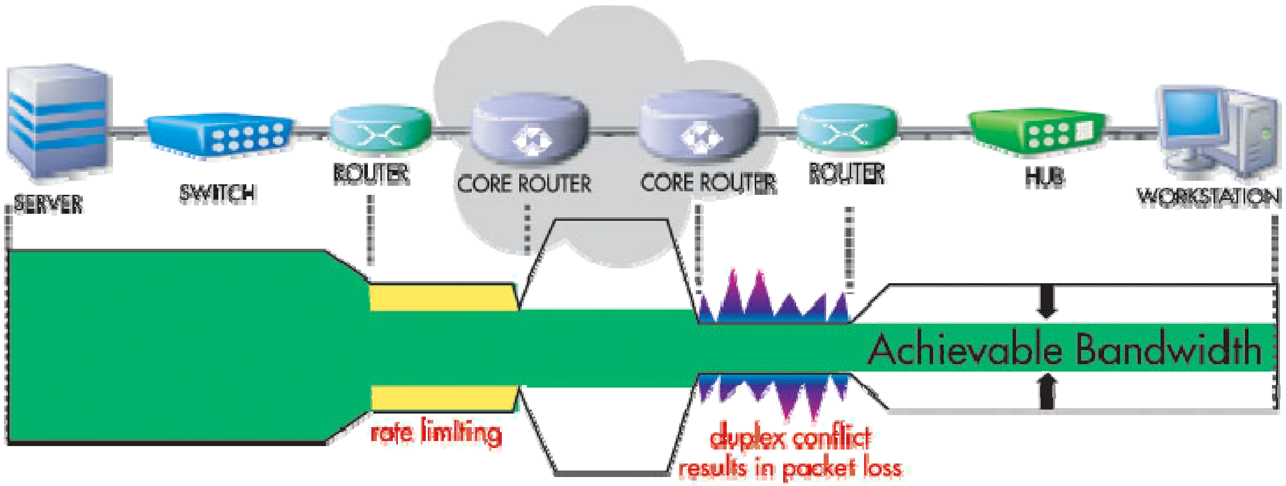
- Consider a TOS-based QoS mechanism employed on a converged network with a duplex conflict



  - **TOS bit implemented at Layer 3 – IP header bit**
  - **Conflict occurs at Layer 2 – frame collision**
  - **Packets arriving at switch port are not differentiated prior to collision**

→ QoS never has a chance to operate

Clean Networks Perform

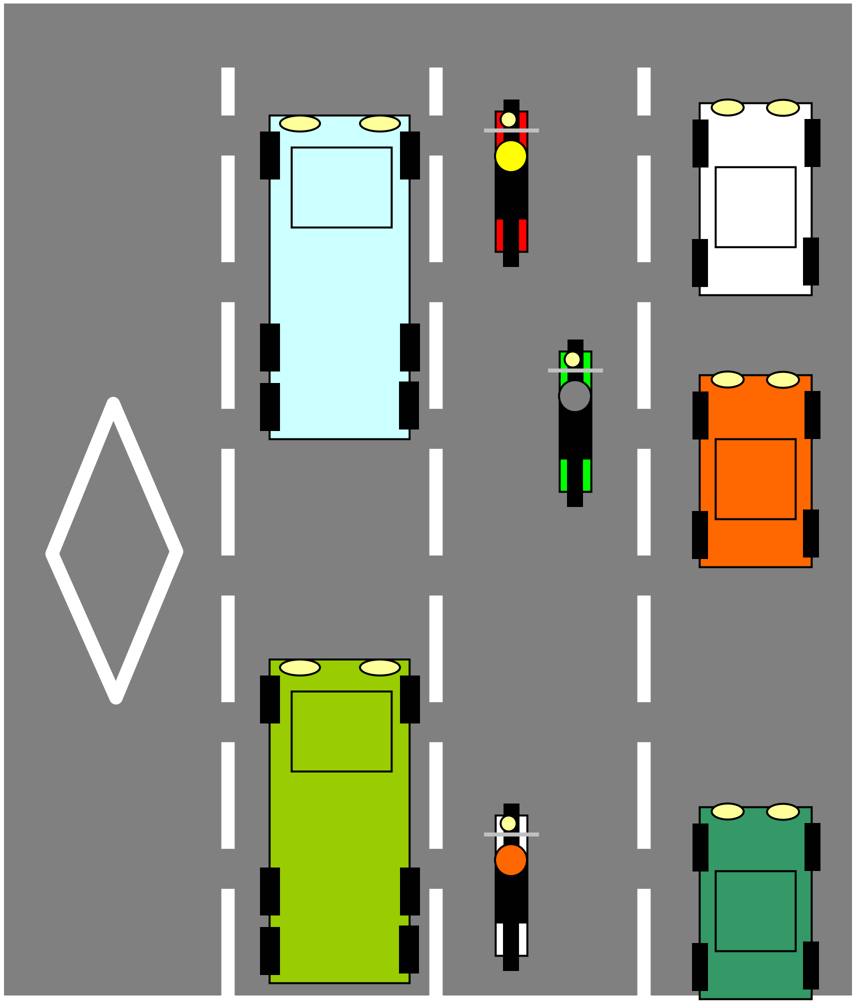


Outline

- QoS as deliberate performance degradation – bandwidth as an option
- QoS doesn’t fix broken networks
- Proper QoS needs to be app-specific
- Hidden costs
  - **Complexity of implementation**
  - **Support and truck rolls**
  - **Equipment costs**
- Not just technological – also economic

Modeling for Applications

- On a healthy road, the type of app then defines performance
- Characterize app
- Define performance required



Modeling Application Performance

Network inputs for VoIP

Latency/Loss/Jitter

↳ **Psycho-Acoustic Assessment model**

Emodel (ITU G.107)

↳ **Subjective MOS**

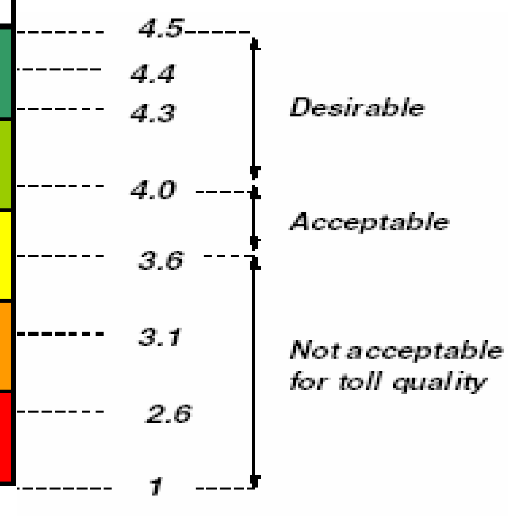
Mean Opinion Score (ITU-T P.800)

E-Model Mapping: R → MOS

E-model generated "R-value" (0-100) maps to well-known MOS score

R -value range	speech transmission quality category
90 - 100	best
80 - 90	high
70 - 80	medium
60 - 70	low
0 - 60 *	(very) poor

MOS



MOS – A Windsock in a Storm

- Mean Opinion Score (ITU-T P.800)
  - **16 people in a room have an opinion?**
  - **MOS is "3.4". Acceptable?**
  - **And so what? Not ready and still not ready….**

MOS points the way – but it isn't a solution

Application Performance: VoIP

Typical metric for determining network readiness

Packet Loss	Delay Variation	Latency	RTT	Status
< 1%	< 30ms	< 50ms	<100ms	Good
< 5%	< 60ms	< 80ms	< 160ms	Caution
> 5%	> 60ms	> 80ms	> 160ms	Bad

But not sufficient to assure performance!!

Define Application Performance

- ITU (SG 12/13) Y.1540 and Y.1541
  - **Y.1540 - Internet protocol data communication service - IP packet transfer and availability performance parameters**
  - **Y.1541 - Network performance objectives for IP-based services**
- IETF (IPPM) RFC's 2330, 2678 through 2681
  - **RFC 2330 - Framework for IP Performance Metrics**
  - **RFC 2678 - IPPM Metrics for Measuring Connectivity**
  - **RFC 2679 - A One-way Delay Metric for IPPM**
  - **RFC 2680 - A One-way Packet Loss Metric for IPPM**
  - **RFC 2681 - A Round-trip Delay Metric for IPPM**
- Application Index Alliance (www.apdex.org)

Metrics – ITU Y.1541

Table 1/Y.1541 – Provisional IP network QoS class definitions and network performance objectives							
Network parameter	**Nature of network performance objective**	**QoS Classes**					
		Class 0	**Class 1**	**Class 2**	**Class 3**	**Class 4**	**Class 5 Unspecified**
IPTD	Upper bound on the mean IPTD	100 ms	400 ms	100 ms	400 ms	1 s	U
IPDV	Upper bound on the $1 – 10^{–3}$ quantile of IPTD minus minimum	50 ms	50 ms	U	U	U	U
IPLR	Upper bound on the packet loss	$1 \times 10^{–3}$	$1 \times 10^{–3}$	$1 \times 10^{–3}$	$1 \times 10^{–3}$	$1 \times 10^{–3}$	U
IPER	Upper bound	$1 \times 10^{–4}$					U

Example - TCP Buffer Tuning

- Linux 2.6.17 has Tx/Rx auto-tuning and *default* 4 Mbyte maximum window size
  - **100 Mb/s on a 300 ms path**
  - **1 Gb/s on a 30 ms**
  - **assuming extremely loss-less network**

- MS touts 64 Kbyte window for Vista

QoS Mechanisms

- RSVP
- DiffServ
- Type of Service (ToS)/DSCP
- Class of Service (CoS)
- 802.1p
- Call

Challenges for QoS

- End-to-end
- Robust and reliable
- Converged, multi-purpose
- Shared / inter-domain
- Satisfy real-time, TCP, and best-effort
- Simple, scalable

Three Classes of Traffic Need

- TCP-based - requiring queues
- Streaming - queues contraindicated
- Scavenger – opportunistic/insensitive

Three Classes of Service Provided

- Dedicated
- Premium/preferred
- Best-effort

…. poor correspondence to needs

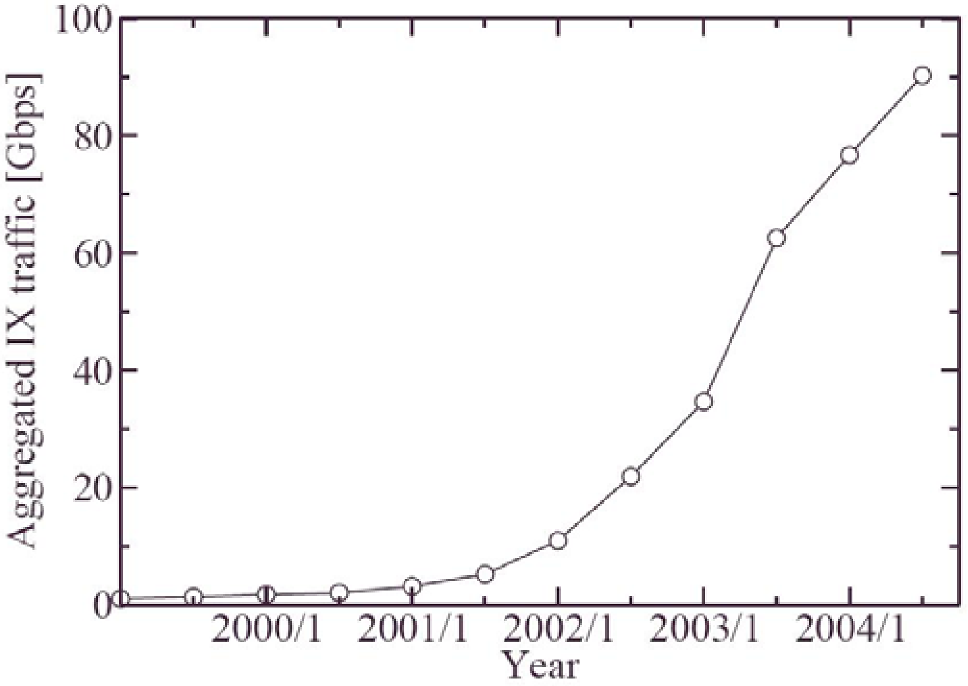
Broadband Over-subscription

- Nearly 20-to-1 as standard approach
- 1 Gbps to the home – economically feasible (Shalunov – Internet2)
- QoS as a response to artificial scarcity
- Fat pipes with competition are better for everyone (except the monopolies)

Core Congestion

"The Impact of Residential Broadband Traffic on Japanese ISP Backbones"

- Fukuda, Cho, Esaki
- SIGCOMM v35n1 2005
- 100 Mbps residential



Internet2 Back to QoS?

- Release of new auto-tuning stacks
- Heavier flows saturating the core with new Linux release?
- QoS/queues in the core for TCP flows
- Edges no longer the constraints
- But just upgraded again …..

QoS Alternatives: *Fluid* Bandwidth

- A Framework for Quality of Service Control Through Pricing Mechanisms
  Steven Shelford IEEE (NOMS 2006)

- Merkato as exemplar
  http://www.invisiblehand.net

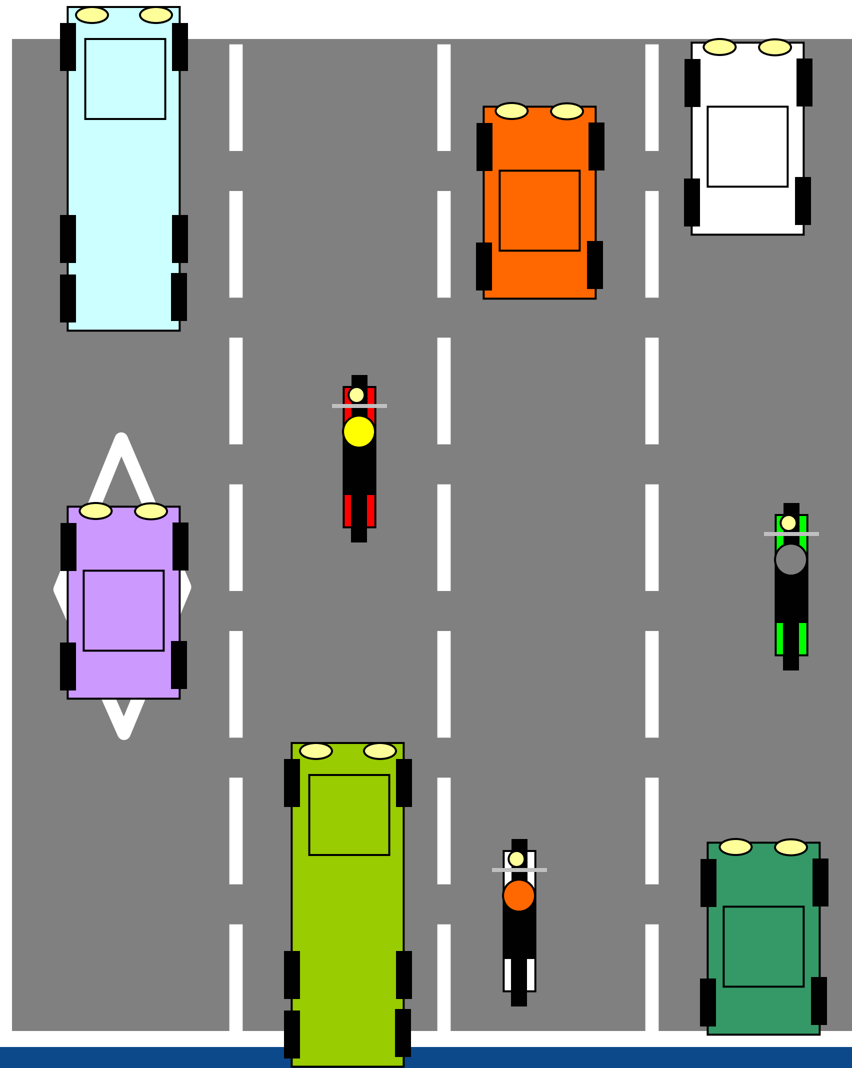
- Conceptually powerful – but feasible?
  - **Pay-per-play as function of quality**

QoS Alternatives: Scavenger

- Scavenger
  - “Worst effort” traffic
  - Marked DSCP = 8 (Precedence = 1)
  - Applied on incremental basis
  - No longer need to police DSCPs at every network boundary
  - Run pipes hotter while enjoying all the performance benefits of over-provisioning in the default class

Road Scavenger

- Traffic marked is given reduced priority
- Reduced priority offers reduced cost
- Optimization requires adaptive approach



Outline

- QoS as deliberate performance degradation – bandwidth as an option
- QoS doesn't fix broken networks
- Proper QoS needs to be app-specific
- Hidden costs
  - **Complexity of implementation**
  - **Support and truck rolls**
  - **Equipment costs**
- Not just technological – also economic

Hidden Costs of QoS

- Cost of implementation
  - **Complexity**
  - **Capital investments**
- Cost of maintenance
  - **Support center**
  - **Truck rolls**
- Cost of experimentation effort
  - **Customer churn**
  - **Changes**
- Complex inter-domain contracts

Hidden Costs: Implementation

- Converged QoS parameters
  - **Several orders of magnitude**
  - **New application type**
- High-performance TCP requires queues
- Must be pervasive
  - **All or nothing**
- Complexity required to achieve balance
  - **Emergent behaviors unpredictable**

Example - Hidden Costs: Implementation

- DSCP policing and re-marking functionality not available on every router
- When available, might not be supported on every interface
- When supported, sometimes significant performance cost (> 50% pps rate drop)
- Well-provisioned core networks can simply ignore these markings

Example - Hidden Costs: Implementation

- Police and re-mark on every boundary
  - **a practical hurdle to deployment of inter-domain QoS**
- Even properly equipped networks still suffer
  - **increase of operational complexity**

Hidden Cost: Maintenance

- Relatively fragile
- Break-fix costs
- Limited visibility for diagnostics
- Support centers expensive
- Quality of experience drives churn

Compare to the maintenance costs of adding bandwidth….

Hidden Cost: Experimentation

- “Black art” requires experts
- Hard to experiment while running operational network
- Hard to determine viability in isolation
- Not reproducible; not scalable
- Can cost customers

Hidden Cost: Inter-Domain

- Providers will make choices that suit themselves
- Cannot control what they cannot see
- It is about application/user experience QoE
- Customers will have to comply
- SLAs have always been a major drawback
- Need competition to decide if QoS is best choice – and which QoS

Outline

- QoS as deliberate performance degradation – bandwidth as an option
- QoS doesn't fix broken networks
- Proper QoS needs to be app-specific
- Hidden costs
  - **Complexity of implementation**
  - **Support and truck rolls**
  - **Equipment costs**
- Not just technological – also economic

Network Neutrality

- Introduces fragility and complexity
- Unproven – cellular experience
- Business model and economics conflict with technological model
- Contrary to the original principles of the Internet → will it still “work” if broken?
- Scavenger with over-provisioning
- QoE depends on QoS

Ask A Ninja

http://www.askaninja.com/news/2006/05/11/ask-a-ninja-special-delivery-4-net-neutrality?a=8

Key Points to Take Home

- QoS does NOT fix base performance
- Over-provisioning easier and more reliable
- QoS essential where true scarcity of capacity (e.g. wireless)
- ROI of overprovision-vs-QoS obscured
- Business models drive QoS frameworks
- Technically, QoS appropriate in Scavenger model, not premium service
- Still in doubt - ask a ninja

QUESTIONS?

Contact:
Loki Jörgenson
ljorgenson@apparentNetworks.com
(604) 433-2333

www.apparentNetworks.com

Apparent Networks