

Back to the Future Part 4: The Internet*

Simon S. Lam

Department of Computer Sciences
The University of Texas at Austin
lam@cs.utexas.edu

Slide 1

Back to the Future Part 4: The Internet

Simon S. Lam
Department of Computer Sciences
The University of Texas at Austin

Sigcomm 2004 Keynote
S. S. Lam 1

I thank the SIGCOMM Awards committee for this great honor. I would like to share this honor with my former and current students, and colleagues, who collaborated and co-authored papers with me. Each one of them has contributed to make this award possible. To all my coauthors, I thank you.

It is most gratifying for me to accept this award at a SIGCOMM conference because my relationship with the conference goes back a long way. Over 21 years ago, I organized the very first SIGCOMM conference held on the campus of the University of Texas at Austin in March 1983. Let me tell you a little story about the first SIGCOMM.

Back in 1982–1983, the SIGCOMM chair was David Wood. The vice chair was Carl Sunshine. One day during the summer of 1982, Carl called me up on the phone. He said, “Simon, SIGCOMM is broke. We have a lot of red ink. How would you like to organize a SIGCOMM conference to make some money for us?” I said, “Okay, I will do it.” Then, we talked a bit. Towards the end of our conversation, Carl said, “By the way, Simon, as I said, SIGCOMM is broke. So if you lose any money on the conference, you will have to take care of the loss yourself.”

*This is an edited transcript of the author’s SIGCOMM 2004 keynote speech given on August 31, 2004. This work was sponsored in part by Nation Science Foundation grants ANI-0319168 and CNS-0434515 and Texas Advanced Research Program grant 003658-0439-2001.

I was much younger then. The great thing about young people is that they have no fear. I don’t remember that I ever worried about losing money. I arranged to use the Thompson conference center on campus which charged us only a nominal fee. We had a great lineup of speakers. The pre-conference tutorial on network protocol design was by David Clark, a former SIGCOMM Award winner. The Keynote session had three speakers: Vint Cerf, another former SIGCOMM Award winner, John Shoch of Xerox who was in charge of Ethernet development at that time, and Louis Pouzin, yet another former SIGCOMM Award winner. We had 220 attendees, an excellent attendance for a first-time conference. We ended up making quite a bit of money. End of story.

Slide 2

IP won the networking race

- Many competitors in the past
 - SNA, DECnet, XNS
 - X.25, ATM, CLNP
- IP provides end-to-end delivery of datagrams, best-effort service only
- IP can use any link-layer technology that delivers datagrams (packets)

```
graph TD; FTP[FTP] --- TCP[TCP]; SMTP[SMTP] --- TCP; HTTP[HTTP] --- TCP; RTP[RTP] --- UDP[UDP]; DNS[DNS] --- UDP; TCP --- IP[IP]; UDP --- IP; IP --- LINK1[LINK1]; IP --- LINK2[LINK2]; IP --- LINKn[LINKn];
```

Sigcomm 2004 Keynote
S. S. Lam 2

IP won the networking race for data communications. However, as recently as ten years ago, before widespread use of the World Wide Web, it was not clear that IP would be the winner. The original ARPANET in the 1970s and later the Internet had many competitors in the past.

In industry, the competitors included SNA, IBM’s Systems Network Architecture, DECnet with the Digital Network Architecture, and XNS which stands for Xerox Network systems. SNA was developed around 1975. At that time, IBM was the 500-pound gorilla in the computer industry. IBM already had networking products used by numerous customers. Those early networks had a tree topology

with a mainframe computer connected to data concentrators and terminals. But SNA was designed to have a new architecture with a mesh topology similar to ARPANET's architecture. Back in 1975, very few people would have predicted that the small, experimental ARPANET, rather than SNA, would emerge 25 years later as winner of the networking race. In fact, when SNA was first developed, SNA was the acronym for Single Network Architecture. IBM had very ambitious goals. However, by the time SNA was announced to the public, the name was changed to Systems Network Architecture, possibly due to antitrust concerns.

IP also had strong competitors in the standards world. First, there was X.25 in the 1970s and 1980s. Then there was ATM in the 1990s. Also, there were ISO protocols such as CLNP (Connectionless Network Protocol).

IP is a very simple protocol. It provides end-to-end delivery of datagrams, also called packets. It provides no service guarantee to its users. In turn, IP expects no service guarantee from any link-layer technology it uses. Therefore, any network can connect to IP.

Ten years ago, when Internet applications were primarily email, ftp, and web, IP's simplicity was its greatest strength in fighting off competitors. In the future, however, IP's simplicity is possibly a liability because the requirements of Internet's future applications will be more demanding, particularly the requirements of interactive multimedia applications.

Slide 3

IP's underlying model is a network of queues

- Revolutionary change from the *circuit* model (Kleinrock 1961)
 - Each packet is routed independently using its destination IP address
 - No concept of a flow between source and destination, no flow state in routers
- Unreliable channels, limited buffer capacity
- Prone to *congestion collapse*

Sigcomm 2004 Keynote 3
S. S. Lam

IP's underlying model is a network of queues. Packets are transmitted from one node to another, leaving one queue and joining the next queue as they travel from source to destination. Each packet travels independently. There is no concept of a flow of packets belonging to the same session between source and destination processes.

I give credit to Len Kleinrock for proposing the network of queues model for data communications as an alternative to the circuit model in telephone networks. Some might argue that Kleinrock did not invent the name "packet switching." But he was definitely the first to propose the network of queues model in his 1961 Ph.D. dissertation proposal.

In a real network, channels are unreliable. More importantly, buffer capacity for queuing is limited. Therefore,

packets may be discarded because of buffer overflow. As a result, a network of queues is prone to congestion collapse, as illustrated in this figure, where system throughput decreases rapidly as offered load becomes large. This is a weakness of IP that we should keep in mind.

Slide 4

Congestion collapse—ALOHA channel

- The **ALOHA System** (Abramson 1970)
- Poisson process assumption
 - pure ALOHA throughput
 $S = G e^{-2G}$ (Abramson)
 - slotted ALOHA throughput
 $S = G e^{-G}$ (Roberts)
- **ARPANET Satellite System** (1972)

Sigcomm 2004 Keynote 4
S. S. Lam

Congestion collapse was a subject of great interest to me when I was a Ph.D. student at UCLA. Congestion collapse of the ALOHA channel was the inspiration of my dissertation research. In the next several slides, I will show you a little bit of my early work in the 1970s. My early work had a lot of influence on my thinking, and may explain why I hold certain opinions later on in this talk.

I went to UCLA as a graduate student in 1969 when the first IMP of the ARPANET was being installed there. The next year, Norm Abramson presented his seminal paper on the ALOHA System in the Fall Joint Computer Conference. Abramson made use of a Poisson process assumption and obtained the well-known formula for the throughput of pure ALOHA.

The throughput formula for slotted ALOHA was an observation by Larry Roberts in 1972. By then, the ARPANET was already up and running. Roberts was looking for something else to do. Motivated by the ALOHA System, he started the ARPANET Satellite System project, later renamed the Packet Satellite project. I was one of the first graduate students to work on the project. Others include Bob Metcalfe who was then at Xerox PARC and Dick Binder who was working on the ALOHA System.

Slide 5

Congestion collapse was not a possibility discussed by ALOHA System researchers. In their early presentations, the ALOHA System was always said to work well. This is because the ALOHA channel was vastly under-utilized. In today's terminology, it was over-provisioned. One of my opinions is that over-provisioning covers up problems.

Also, with the Poisson process assumption, Abramson abstracted away in his analysis the need for a backoff algorithm. So the first thing Len Kleinrock and I did for slotted ALOHA was to introduce a realistic backoff algorithm for

Congestion collapse—ALOHA channel (cont.)

- The early ALOHA System was vastly *under-utilized*
- *Backoff* algorithm for slotted ALOHA (Kleinrock-Lam 1973)
 - Retransmit a collided packet randomly into one of K future time slots
 - Poisson process assumption implies $K \rightarrow \infty$

Sigcomm 2004 Keynote 5
S. S. Lam

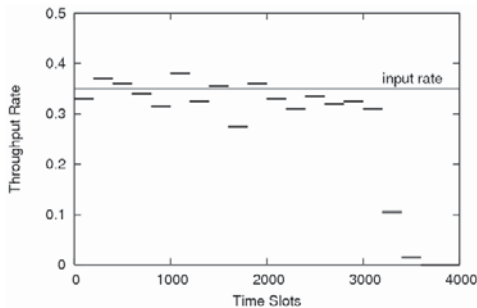
collided packets. In our algorithm each collided packet is retransmitted randomly into one of K future time slots, where K is a parameter. Our analysis produced some interesting observations. First, the Poisson process assumption implies that K is infinity. In hindsight, this is obvious—it is not possible to get independent arrivals unless collided packets are retransmitted into the infinite future. Second, for a given channel throughput there is a K value that minimizes average delay.

Slide 6

Congestion collapse—ALOHA channel (cont.)

- ASS Note 48 (Lam 1973)

$K = 15$



Sigcomm 2004 Keynote 6
S. S. Lam

In early 1973, I ran a simulation to confirm my conjecture that ALOHA channels are unstable. I waited a long time to do this because, in those days, running a simulation was not as easy as today. Obviously, we didn't have ns. In fact, there was no simulation package of any kind to use. We had to write simulation programs from scratch.

My simulation results were disseminated as ASS Note 48, where ASS stands for ARPANET Satellite System. The experiment shown here was run for a fixed value of K equal to 15. The channel input rate was 0.35 which is less than the theoretical maximum value of 0.368 for slotted ALOHA. Here I have plotted channel throughput as a function of time in slots. Notice that congestion collapse occurred precipi-

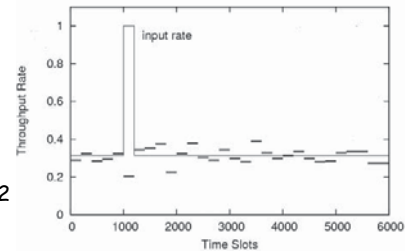
tously after about 3,000 time slots. I believe that this was the first experimental demonstration of congestion collapse and ALOHA instability.

Slide 7

Congestion collapse—ALOHA channel (cont.)

- *Adaptive backoff* algorithm (Lam 1974): Retransmit a packet with m previous collisions into $K(m)$ slots, where $K(m)$ is monotonically nondecreasing in m

$K(1) = 10$
 $K(m) = 150, m \geq 2$



Sigcomm 2004 Keynote 7
S. S. Lam

Subsequently, I developed a Markov chain model that explains ALOHA instability. I also proposed and investigated several adaptive backoff algorithms. This particular heuristic might look familiar to you. The idea is to use the number of previous collisions of a packet as an indication of current load. A packet with m previous collisions is retransmitted into an interval of $K(m)$ slots, where $K(\cdot)$ is a monotonically nondecreasing function of m . In particular, $K(m)$ increases rapidly as m increases from 1. Exponential backoff is a special case of this heuristic.

For the experimental results shown here, I used $K(1) = 10$, which is very close to optimal for a wide range of channel throughput. I used $K(m) = 150$ for $m \geq 2$. There was no need for a larger K value because I found by analysis that $K = 150$ was sufficient to stabilize a slotted ALOHA system with 400 users and this particular simulation had 400 users.

I have plotted channel throughput as a function of time in slots. In this experiment, the input rate was about 0.32 and it jumped up to 1 for an interval of 200 time slots. As we can see, the adaptive backoff algorithm handled the input pulse easily. This is because the jump in K value from 10 to 150 was actually faster than exponential.

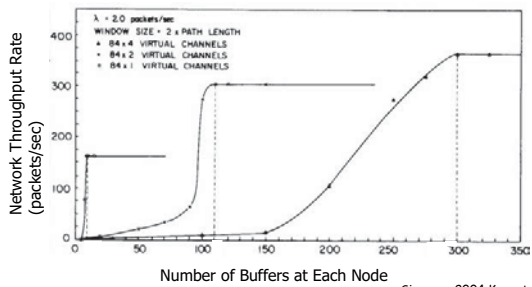
Slide 8

As a graduate student, I was also very interested in the possibility of congestion collapse in the ARPANET. We knew that a network of queues models with limited buffer capacity would be prone to congestion collapse. So the real question was this: What about packet networks with window flow control? The original ARPANET had flow control using a message called RFNM and the subsequent TCP had window flow control. From Little's law, we knew that window flow control provides congestion control to some extent.

My conjecture was that static window flow control would not be sufficient to avoid congestion collapse unless each network node had a very large buffer capacity. I had to wait until 1979 before I had enough computing resources to run

Congestion collapse—packet networks

- Network of queues with limited buffers
 - *Static window flow control in TCP not sufficient (Lam, late 1970s)*



meaningful simulation experiments to validate this conjecture. Here are some of my simulation results for a 7-node network. Each node is a router. The vertical axis is network throughput rate. The horizontal axis is the number of buffers in each router. There are three curves, the left for a network with 84 flow-controlled sessions, the middle for a network with 168 flow-controlled sessions, and the right for a network with 336 flow-controlled sessions.¹ The window size for each session was fixed at twice the number of hops between the session's source and destination. Each point on a curve represents the network throughput of a simulation experiment that ran for 150 seconds of simulated time. So points with low network throughput indicate that the networks were experiencing congestion collapse.

There are two ways to look at these results. First, for a fixed number of flow-controlled sessions (consider the curve on the right for a network with 336 flow-controlled sessions) if the buffer capacity at each node is too small, congestion collapse would occur quickly. Second, consider a network with a fixed amount of buffer capacity, say 150 buffers per node. If the number of sessions is 168, the network will most likely not incur congestion collapse. But if the number of sessions increases to 336, then congestion collapse will occur quickly.

Slide 9

Even though I knew that the TCP window size needs to be adaptively controlled, I was not smart enough to design an effective adaptive algorithm. So several years went by and after the Internet experienced real congestion collapse, the problem was solved by Van Jacobson, Raj Jain, and others. Van Jacobson's algorithms for TCP congestion control are generally acknowledged as the main reason for stability of the current Internet where more than 90% of the traffic are carried by TCP.

Nowadays, UDP, rather than TCP, is the transport protocol preferred by voice and video applications because UDP does not perform congestion control. If the Internet has to carry more and more UDP traffic, its stability will be affected. One possible way to address this concern is to entice designers of multimedia applications to use TCP-friendly

¹Flow-controlled sessions are labeled as "virtual channels" in the figure.

Internet congestion control

- Van Jacobson's algorithms for TCP congestion control (late 1980s)
 - *main reason for stability of the current Internet*
- UDP does not perform congestion control
 - *preferred by voice and video applications*
- More and more voice and video traffic will impact Internet stability
 - *I believe in differentiating voice and video flows as well as flow admission control*

Sigcomm 2004 Keynote
S. S. Lam 9

congestion control. This is advocated by Sally Floyd and others. This approach may work for the current Internet which has only small amounts of voice and video traffic. However, in the future, if voice and video traffic become very large components of Internet's traffic mix, I believe that differentiating voice and video flows together with the use of flow admission control will be a better approach.

Slide 10

Efforts to extend/replace IPv4 (past 15 years)

- IP multicast
- QoS support - IntServ, RSVP, DiffServ
- Active Networks research program of DARPA
- IPsec - retrofitting IP with security
- IPv6 - replacing IPv4
 - 128 bit IP address
 - flow concept to support QoS
- Mobile IP
- ...

Sigcomm 2004 Keynote
S. S. Lam 10

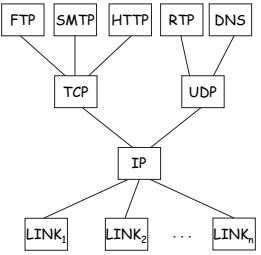
The work of Van Jacobson was the start of a new era of Internet research beginning from the late 1980s by a new generation of researchers. In SIGCOMM '88, Steve Deering's paper was the beginning of a large body of research on IP multicast. The fair queueing paper (by Demers, Keshav, and Shenker) in SIGCOMM '89 was probably the beginning of an even larger body of research on quality of service. Throughout the 1990s, the research community was most interested in improving, extending, or even replacing IPv4. These efforts also include the Active Networks research program funded by DARPA, IPsec with the objective of retrofitting IP with security measures, IPv6 with the objective of replacing IPv4, Mobile IP with the objective of supporting mobility, etc. For this audience, it is unnecessary for me to elaborate on any of these topics. It suffices to say

that numerous protocols have been designed, and thousands of papers have been written and published, as well as Internet drafts and RFCs. The research community worked very hard on these topics. But as we know, very few of these efforts have been deployed to a significant extent. Of the ones on this list, I would say that DiffServ and IPsec have meaningful deployment on the Internet, perhaps, because they can be quite useful even when deployed incrementally.

Slide 11

Don't mess with IP ?

- In recent years, the research community has moved on to other areas
 - P2P overlays
 - Wireless (ad hoc networks, sensor networks, satellite networks)
 - Measurements
- But the IP foundation, currently relying on over-provisioning, still needs work



Sigcomm 2004 Keynote 11
S. S. Lam

The research community seems to have decided not to mess with IP any more. In recent years, many researchers have redirected their efforts to the application layer for multicast support. Some are trying to use the transport layer for QoS support. Most of the current Internet research efforts are concerned with either the application layer at the top, such as research on various P2P systems, the link layer at the bottom, such as research on sensor networks, wireless ad hoc networks, and satellite networks, or methodology for Internet measurements.

It is often said that with recent advances in DWDM (dense wavelength division multiplexing), the Internet core is over-provisioned. Therefore, there is no need to worry about IP any more. Actually, I disagree with that statement. I think that while the Internet core is over-provisioned in the near future, we should take advantage of this window of opportunity to do research that will strengthen IP's foundation.

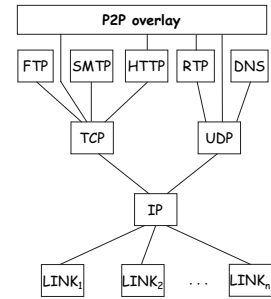
Slide 12

Let me digress and express a few opinions about P2P systems since there is so much research interest in P2P overlays. I think that P2P research is an exciting research area and P2P overlays will enable the Internet to support many new distributed applications. However, P2P overlays are somewhat inefficient in their use of underlying Internet resources and they do not directly address IP's stability and QoS issues.

Imagine that the Internet protocol stack is a house with IP as its foundation. Adding another floor to the house will provide more space for new functions. However adding another floor will not address foundational issues.

Is P2P overlay a panacea ?

- P2P overlay supports many *new* distributed applications
- P2P overlay is *inefficient* in its use of underlying Internet resources
- P2P overlay does not directly address IP's *foundational issues* (stability, QoS)



Sigcomm 2004 Keynote 12
S. S. Lam

Slide 13

IP as the future universal interface

- Payload in the form of IP packets to enable new applications across different telecom networks
 - analog → digital → packet**
- Recent news
 - Microsoft unveiled plans for developing IPTV (October 2003)
 - SBC to invest \$6 billion on fiber to home for TV services (June 2004)

Sigcomm 2004 Keynote 13
S. S. Lam

Why should we be concerned with IP's foundation?

I believe that IP is on track to become the universal interface² for telecommunications. Like the previous technology transformation from analog to digital transport, we are in the midst of a technology transformation from digital to packet transport. There is no alternative to IP to serve as this universal packet interface.

The migration of voice traffic from telephone networks to IP networks has already begun. The next step is the migration of television services. You may have read about these two news items. In October 2003, Microsoft unveiled major plans for developing IPTV. More recently I read that SBC would invest \$6 billion dollars on fiber to home for television services to compete with cable companies.

Slide 14

While more than 90% of the current Internet traffic is TCP traffic, which is responsive to congestion, I believe the network traffic mix will change substantially in the future. Here are a few numbers to consider.

²More specifically, the IP packet format.

Changing network traffic mix

- Much more voice and video traffic
- Current traffic of a major telecommunications carrier
 - Circuit switched voice 1.2 petabytes/day
 - Internet traffic 1.5 petabytes/day
- Television services over IP
 - Back of the envelope calculation:
4-8 Mbps, 10,000 seconds/day, 10^8 TV sets
→ 500-1000 petabytes/day to end users

Sigcomm 2004 Keynote 14
S. S. Lam

A major telecommunications carrier currently delivers about 1.2 petabytes of circuit-switched voice per day and about 1.5 petabytes of Internet traffic per day. Indeed, the amount of data is larger than the amount of voice traffic. But if voice continues its migration to the Internet, the voice traffic component would be significant.

What about television services? I don't have real data. But here are some back of the envelope calculations: I used 4-8 Mbps for a high quality video signal, such as MPEG-2 or MPEG-4, a round number of 10,000 seconds per day per television set (this is about two and three-quarter hours), and 100 million television sets. I got between 500 to 1,000 petabytes per day to end users. This amount will have to be distributed over a large number of networks, and I have not taken into account savings from using multicast. Still the point here is that the amount of television traffic will be large. With the addition of both telephony and television traffic, the network traffic mix will be substantially different from today's traffic mix.

Slide 15

A pragmatic approach

1. *Learn from the evolution of Ethernet*
Ethernet technology today is very different from Ethernet technology 20 years ago.
 - Transmission rates: 10 Mbps, 100 Mbps, 1 Gbps, 10 Gbps
 - Switching protocols:
 - CSMA/CD on a cable,
 - CSMA/CD on a hub,
 - collision-free switching,
 - full-duplex point-to-point, both WAN and LAN
 - A variety of coding techniques and mediaOnly the *Ethernet frame interface* remains the same.

Sigcomm 2004 Keynote 15
S. S. Lam

For IP to become the universal interface there is a pragmatic approach. First, we can learn from the evolution of Ethernet. Ethernet technology today is very different

from Ethernet technology twenty years ago. In particular, 10 Gbps Ethernet is very different from 10 Mbps Ethernet. Only the Ethernet frame interface remains the same.

All kinds of technologies now co-exist under the Ethernet interface. Ethernet was originally designed for a large population of bursty users. For many years, Ethernet was synonymous with CSMA/CD. Ethernet is now also used for high-throughput point-to-point full-duplex communications, not only within a local area but also for wide-area communications.

Slide 16

A pragmatic approach (cont.)

2. *Accept in some form a competing idea that has been vanquished again and again by IP but refuses to die and go away:*

virtual circuit packet switching

- X.25 1970s - 1990s [L. Roberts]
- Frame Relay 1980s - present
- ATM early 1990s - present
- Label switching, MPLS late 1990s - present

Sigcomm 2004 Keynote 16
S. S. Lam

Second, we should accept *in some form* a competing idea that has been vanquished again and again by IP but it refuses to die and go away, namely, the virtual circuit packet switching idea. The virtual circuit idea first appeared in X.25 which was developed by a company called Telenet in the mid 1970s and it became the first international standard for packet networks.

The CEO of Telenet was Larry Roberts. This is the same Larry Roberts who built the original ARPANET several years earlier. In 1973, when the ARPANET was still in its infancy, Larry left ARPA to be CEO of Telenet. Telenet would be the first public network to provide packet switching services, somewhat like today's ISPs. Larry was a visionary. But as an entrepreneur he was about 20 years too early.

X.25 was also used in other public packet networks, such as Datapac in Canada and Transpac in France. X.25 is gone now. But the virtual circuit idea reappeared in frame relay, which is a streamlined version of X.25. It reappeared again in ATM, and more recently in MPLS. All three technologies are working as link-layer technologies under IP.

Slide 17

In addition to virtual circuits, we also have real circuit switching running under IP. IP over SONET is one example. Some time in the future, GMPLS will be another one. In GMPLS, forwarding can be based upon a particular time slot in a TDM frame, a particular wavelength, or a particular optical port. Therefore a label-switched path in GMPLS is a real circuit.

Real circuit switching also under IP

- IP over SONET
- GMPLS forwarding based on TDM time slot, wavelength, or optical port
 - Nesting of label switched paths (LSPs) reminiscent of multiplexing/demultiplexing hierarchy in telephone networks

Sigcomm 2004 Keynote 17
S. S. Lam

The nesting of label-switched paths is essentially the same concept as the multiplexing/demultiplexing hierarchy in telephone networks.

So, virtual and real circuits, these old ideas from the telephone world, are alive and doing quite well in the link layer under IP. This is because they are useful. More specifically, they are used by carriers to provision Virtual Private Networks and to provision other carriers. In business, the customer is always right. If there is a need for something, it will not go away.

Slide 18

IP should be a "big tent"

- To "rule" the world of communications, IP has to attend to the needs of new constituents (voice, video)
 - *multiple services to support diverse applications*
- For the research community, *over-provisioning* should be considered a temporary fix, not a permanent solution
- While the core is over-provisioned, access paths to the core are not

Sigcomm 2004 Keynote 18
S. S. Lam

For IP to eventually become the universal interface for telecommunications, i.e., to rule the world of communications, the IP layer itself will need to evolve to provide services that attend to the needs of new constituents, namely, voice and video traffic. When packet switching was first proposed in the 1960s, it was justified with the observation that data traffic is bursty, with a very high peak-to-average ratio, unlike voice traffic. But if the network traffic mix changes, with the addition of large amounts of voice and video traffic, we should be open minded about adopting a multiservice approach for IP.

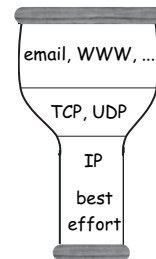
Over-provisioning the core is currently a good solution

for network operators. However, it should not be considered a solution by the research community. Furthermore, the access paths from end systems to the core are not over-provisioned and will require a better solution than best effort in order for IP to provide QoS end to end.

Slide 19

The hourglass shape reconsidered

- Although link-layer technologies are diverse, including virtual and real circuits, only best effort service is available to Internet applications

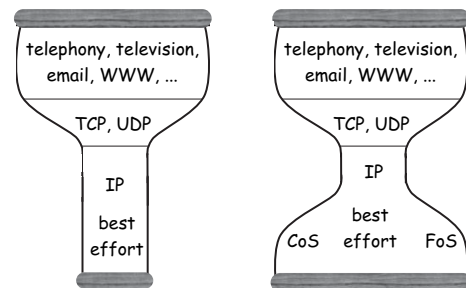


Sigcomm 2004 Keynote 19
S. S. Lam

To summarize what I said in the last few slides, let's reconsider the hour-glass shape of the Internet protocol stack. Internet's link-layer technologies are diverse, including different kinds of virtual and real circuits. However, only the best effort service is available to protocols and applications above IP. With this perspective, the shape of the Internet protocol stack is more like a drinking glass than an hour-glass.

Slide 20

IP needs a broader base



CoS *Class-oriented Service*

FoS *Flow-oriented Service*

Sigcomm 2004 Keynote 20
S. S. Lam

In the future, when telephony and television applications generate substantial amounts of traffic, the Internet protocol stack will become top heavy and it will require a broader base for stability.

I have suggested two services in addition to the best-effort service for IP: The first is CoS, a class-oriented service,

which may be DiffServ as it exists now or a revised version.

The second is FoS, a flow-oriented service targeting voice and video traffic. The flow-oriented service should be a topic of investigation and discussion in the near future.

Slide 21

What is flow-oriented service?

- **Dynamic signaling, flow admission control, flow state, quantitative QoS metric**
 - *Flows subject to admission control rather than random packet drop*
 - *Make use of virtual or real circuits in link layer*

- **Am I reviving IntServ?**
Not exactly

- **Per-flow state and dynamic signaling are not scalable!**
I know

Sigcomm 2004 Keynote 21
S. S. Lam

In my mind, a flow-oriented service targeting voice and video traffic should have these four elements: dynamic signaling, flow admission control, flow state, and a quantitative QoS metric. Both voice and video flows are relatively long in duration. They should be subject to admission control rather than controlled by random packet drop. In particular, they should be differentiated from elastic traffic.

I also envision that a flow-oriented service in IP can make use of virtual or real circuits in the link layer if they are available along a flow's end-to-end path.

Am I reviving IntServ? Not exactly. I am trying to keep alive some useful ideas that are in IntServ. I am also fully aware that per-flow state and dynamic signaling are not scalable.

Slide 22

Two good engineering ideas for voice and video

1. Flow aggregation

- **Current examples**
Virtual paths in ATM, Label stacks in MPLS, RSVP aggregation
- **Each flow is routed along a sequence of "virtual channels" each of which carries a flow aggregate**
- **Flow aggregation reduces state information and signaling overhead thus improving scalability**
 - *In the extreme case, a router has just two flow aggregates (voice and video) for each outgoing channel*

Sigcomm 2004 Keynote 22
S. S. Lam

So what do we do? There are two good engineering ideas

(these are old ideas, not new ideas) that should be further investigated because, I believe, they are appropriate for voice and video.

The first idea is flow aggregation. This idea is old; it goes back to the virtual path concept in ATM. It reappeared in the nesting of label-switched paths in MPLS. Now RSVP also allows aggregation of reservations.

With aggregation, a flow travels from its source to destination along a sequence of "virtual channels." Each virtual channel carries a flow aggregate.

Flow aggregation can be used to reduce both state information and signaling overhead in routers, thus improving scalability. The question is whether the improvements will be enough to make a flow-oriented service commercially attractive. I believe that there is potential for very substantial improvements. Ideally, a router may only need to keep track of the number of flows in a flow aggregate. In the extreme case, a router has just two flow aggregates for each outgoing link, one for voice and the other for video.

Even though the flow aggregation idea has been around for a long time, it is not yet well understood as a research issue. I believe that it is a promising approach. But I don't think we know how to make it work yet.

Slide 23

Two good engineering ideas (cont.)

2. Statistical guarantee

- **A natural service guarantee for a voice or video flow is the flow's loss probability for a given packet delay bound**
$$\text{Prob}[\text{packet delay} > x] < \epsilon$$

Very hard problem! Substantial research in the past but needs much more work to be applicable.
- **Statistical multiplexing gain from flow aggregation**

Sigcomm 2004 Keynote 23
S. S. Lam

The second good engineering idea, I believe, is statistical guarantee. For a voice or video flow, a natural service guarantee is the flow's loss probability for a given packet delay bound

$$\text{Prob}[\text{packet delay} > x] < \epsilon$$

For many years, we, as researchers, fell in love with deterministic guarantees because they have such elegant mathematics and their solution is more or less complete. But network operators are more practical than us researchers and they don't care about elegant mathematics.

Statistical guarantee is a much harder problem with rather complicated mathematics. There has been substantial research in the past, but it is still far from having as clean, and as complete a solution as the one for deterministic guarantees.

The two ideas, flow aggregation and statistical guarantee, have synergy. In particular, flow aggregation provides

statistical multiplexing gain. For a given statistical guarantee, the bandwidth needed to provision a flow aggregate is less than the total bandwidth needed to provision individual flows.

Slide 24

Major research issues

- How to derive service guarantee of a flow from the service guarantee provided to a flow aggregate?
- Dynamic configuration and provisioning of virtual channels for flow aggregates.
- How to efficiently compute the end-to-end statistical guarantee to a flow, under practical assumptions?
 - design, modeling, analysis
 - approximation methods
 - measurement-based techniques

Sigcomm 2004 Keynote 24
S. S. Lam

To apply the ideas of flow aggregation and statistical guarantee, there are some major research issues to be addressed. How to derive the service guarantee of a flow from the service guarantee provided to a flow aggregate? How to dynamically configure and provision virtual channels for flow aggregates? How to efficiently compute the end-to-end statistical guarantee to a flow under practical assumptions?

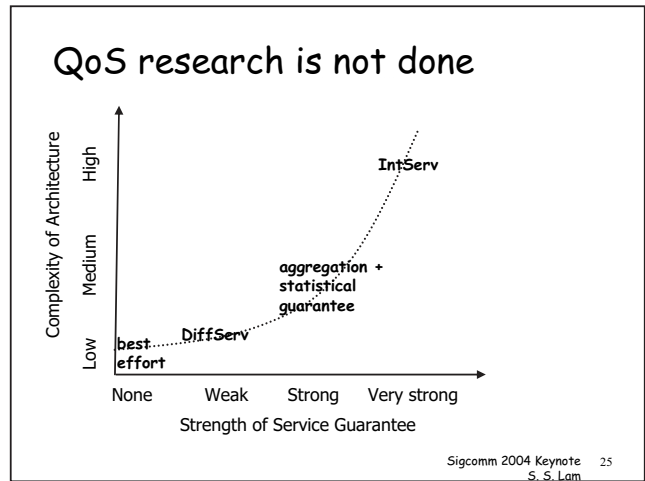
There are various models in the literature with different assumptions for deriving statistical guarantees. In my opinion, the modeling and analysis work is not yet ready for application. For efficient application, we should also investigate the use of approximation methods as well as measurement-based techniques.³

Slide 25

The idea of this simple graph is borrowed from a paper in IEEE Communications Magazine (June 2003) by Nicolas Christin and Jörg Liebeherr, but I have modified it. The vertical axis represents the complexity of architecture, while the horizontal axis represents the strength of service guarantee provided. The three QoS architectures for IP, best effort, DiffServ, and IntServ, with increasing complexity and strength of guarantee, are shown in the figure.

The fact that IntServ has not gained acceptance in the marketplace because of its high complexity should not deter us, as a research community, from another attempt at a flow-oriented service. The scalability issue can be addressed by flow aggregation and by targeting voice and video flows. A design based upon flow aggregation and statistical guarantee will benefit from statistical multiplexing that was not

³An afterthought: While a mathematical model for computing statistical guarantees is useful to have for provisioning, it is perhaps not necessary for implementing the ideas of flow aggregation and statistical guarantee. An efficient measurement technique for checking service conformance is more important.



exploited in IntServ. Using these ideas, I believe it is possible to find good solutions with medium complexity and a fairly strong, quantitative service guarantee.

Slide 26

Inter-provider QoS is a major challenge

- Business and legal issues
- Framework for competitive ISPs to cooperate
 - A quantitative QoS metric for inter-provider agreement
 - A small set of standardized traffic specs for voice and video
 - ...
- It would be nice to have a **de facto standard!**

Sigcomm 2004 Keynote 26
S. S. Lam

Providing quality of service across multiple providers is a major challenge because it involves business and legal issues in addition to technical issues. While the research community cannot directly address business and legal issues, we can indirectly facilitate the resolution of such issues with an appropriate framework for end-to-end QoS deployment. I don't have a framework to present today. As a start, I suggest that the framework should include these two elements.

First, I believe that a quantitative QoS metric is important for inter-provider agreement. When a provider offers a premium service to a customer or another provider, the quality of the premium service should be measurable. Second, I believe that the number of possible traffic specs for voice and video should be limited and standardized. A small number of traffic specs will also reduce complexity and improve scalability.

It would be nice if a de facto standard emerges in the future. So I started thinking: How do we get de facto standards?

Slide 27

How to get one?

- The SSL model
 - *Application-driven*—the need to secure web transactions
 - Now SSL used for other applications as well
- The FedEx model
 - *Profit-driven*—someone takes risk

Possible scenario: Some large ISP takes risk and provides QoS services to a large part of the Internet. Success leads to universal global coverage and a de facto standard.

Sigcomm 2004 Keynote 27
S. S. Lam

I thought of two examples from the past.

First, there is the SSL model. In this case, the de facto standard, namely, the SSL protocol, was application-driven. There was an urgent need to secure web transactions. The web's success led to widespread adoption of SSL as a de facto standard. Now SSL is used for other applications as well.

Second, there is the FedEx model. I know that FedEx is not a protocol. Nevertheless it is still a great example of a new business model created by someone who took risk, motivated by the potential of profit, and succeeded.

Before FedEx, our primary mail delivery service was the US Postal Service, which is a best effort service just like IP. Then someone by the name of Frederick Smith founded FedEx more than 30 years ago to provide guaranteed overnight delivery service. FedEx charged a lot more than the Postal Service. Yet business people are willing to pay for guaranteed delivery. FedEx took a large risk because it had to build a complete infrastructure for end-to-end delivery (i.e., a large fleet of planes and trucks, together with people).

A similar scenario could play out for the Internet: Some ISP takes risk and provides QoS service to a large part of the Internet. If successful, it will lead to universal global coverage and a de facto standard.

Slide 28

Here are my conclusions:

First, over-provisioning covers up problems. We, as the networking research community, should not consider over-provisioning of the core as a solution.

Second, IP won the networking race for data communications. However, to become the universal telecom interface, IP needs to be a big tent, as in politics. To address the needs of new constituents, namely, voice and video, a flow-oriented service is needed to support telephony and television services in the future.

Third, QoS research is not done. I suggest flow aggregation and statistical guarantee as two good engineering ideas that merit further investigation.

Conclusions

- For the research community, over-provisioning should not be considered a solution
- To become the universal telecom interface, IP needs to be a "big tent"
 - *A flow-oriented service needed to support television and telephony services*
- QoS research is not done
 - *Flow aggregation and statistical guarantee merit further investigation*

Sigcomm 2004 Keynote 28
S. S. Lam

Slide 29

Conclusions (cont.)

- From history
 - *Internet*—almost 30 years from initial research to commercial deployment
 - *Packet radio*—about 25 years
 - *QoS research* began in late 1980s

Widespread commercial deployment of QoS within 10 years!

Sigcomm 2004 Keynote 29
S. S. Lam

Lastly, we as researchers need to be more patient. History shows that it takes many years to nurture and develop a new idea until widespread commercial deployment. The Internet took almost 30 years, from the mid 1960s to the mid 1990s. Packet radio took about 25 years, from 1973 to the late 1990s.

Even though packet voice research started in the late 1970s, I believe that QoS research as we know it now began in the late 1980s, about 15 years ago. Therefore to reach 25 years, widespread commercial deployment of QoS is not due for another 10 years.

Acknowledgments

The author thanks Ken Calvert, Thomas Woo, Geoff Xie, and Yin Zhang for their constructive comments during the preparation of this talk. The author also thanks X. Brian Zhang for his help in preparing slides and Min Sik Kim for his help in formatting this paper.