Reservations or No Reservations?

Steve Deering
Xerox Palo Alto Research Center
deering@parc.xerox.com

INFOCOM ’95 Panel Session
Boston, MA
April 6, 1995

"Reservation" (for the purpose of this debate) Does Not Mean…

- advance reservation, i.e., request for network resources for a *future* communication session
  (presumably a session with unusual requirements, e.g., surgery)
- long-term provision of performance guarantees for *particular* traffic over *particular* paths
  (e.g., private networks over shared infrastructure, or "911" service)
- Dave Clark's "aggregate provisioning"

“Reservation” (for the purpose of this debate) Means…

request for, and resulting p*rovision of*:

a particular *performance guarantee*

for a packet flow to / from:

arbitrary host(s) in the *public Internet*

at *flow-initiation time*

Reservations:
What a Network Uses to Say “No”

for normal, every-day communication, such as:

- “phoning” someone
- turning on “the TV”
- clicking on a WWW link

network must say “no” *very rarely*, for service to be acceptable to customers

Adequate Provisioning to Avoid Saying “No”

if a network is able to say “no” very rarely

=> has capacity to meet (normal) peak demand

=> reservations are unnecessary

question is simply what happens on those rare occasions when demand cannot be met:

(1) degrade service for everybody, or

(2) give “perfect” service to some, no service to others.

claim: degraded service is better than no service, for the applications for which reservations are used

Keeping Users Happy ≠ Carrying All Offered Traffic

users can impose priority ranking on *their own* packets, e.g., by:

- sender labeling of drop priority
- unicast end-to-end feedback (flow control)
- joining/leaving different multicast groups

just need to deliver "enough" packets to keep users from switching to competitors

Reservations: Not Only Unnecessary, But Also Harmful

- adds greatly to cost and complexity of routers
  - signallng protocols and admission algorithms
  - non-aggregatable state
  - billing data collection
- adds greatly to cost and complexity of service provision
  - more complicated billing systems
  - more complicated settlement procedures
  - more promises => more exposure
- results in more fragile applications that, for example:
  - fail when bandwidth limited, e.g., wireless, or emergencies
  - fail when delay is high, e.g., talking to moon base
  - fail when loss is high, e.g., mobile wireless