Asynchronous multicast

goal: leverage multicast while at same time allowing asynchronous access by clients

exploit: client workahead buffering, ability receive several unicast/multicast channels simultaneosly

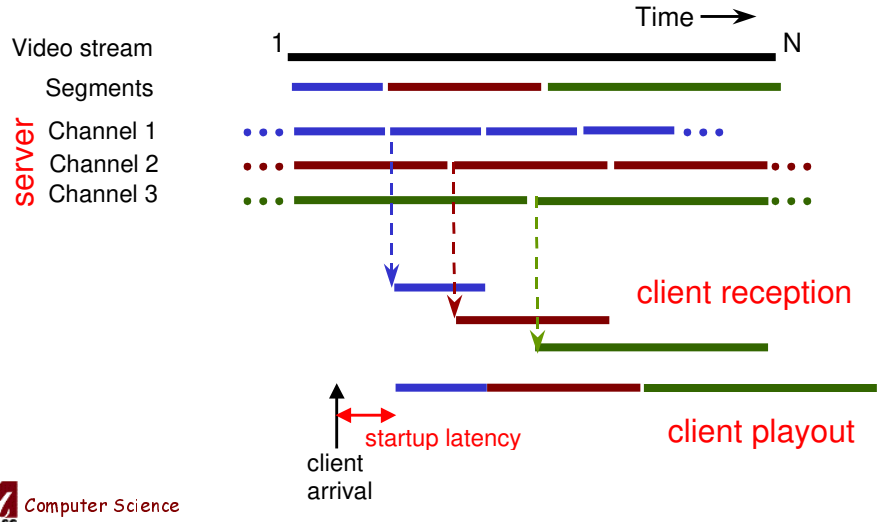
Broadcasting:

- cleverly map video segments to multicast groups
- clients "tune in" to groups

Patching, stream tapping:

- client taps into existing transmission to other client(s)
- server unicasts "patch" of missing data

Periodic Broadcast: Background

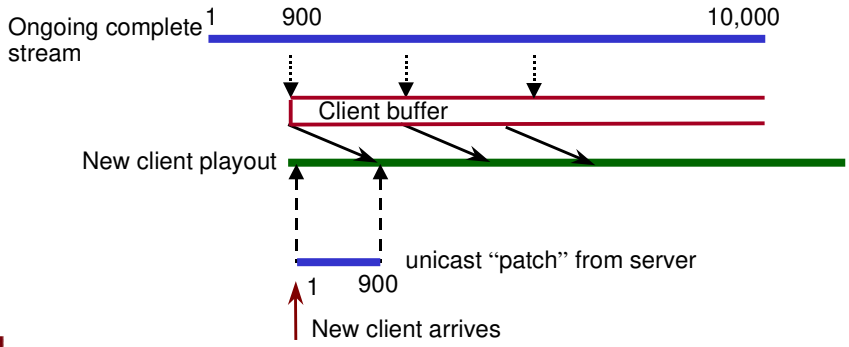


Periodic Broadcast:

- Beautiful algorithmic work: Viswanathan96, Aggarwal96, Hua97, Gao98, Eager98
- L. Gao, J. Kurose, D. Towsley, "Efficient Schemes for Broadcasting Popular Videos," *Proc. IEEE NOSSDAV'98:*
  - lower bound on number channels needed to realize given startup latency for any scheme
  - GDB(i) Greedy Disk-Conserving-Broadcast: periodic broadcast segmenting scheme improving on Skyscraper broadcasting

Video Patching

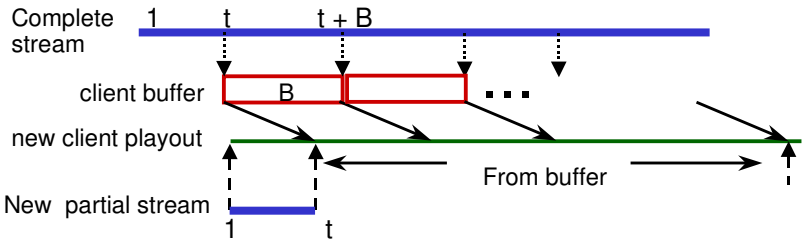
- client "taps" into ongoing stream(s)
- client sent missing initial segment
  - zero startup delay (unlike periodic broadcast)
- [Carter97,Hua97,Gao99]
- Advantage: Server, Network bw reduction w/o startup delay



Periodic Buffer Reuse (PBR) Patching

Q: how to use client buffer of size B ?

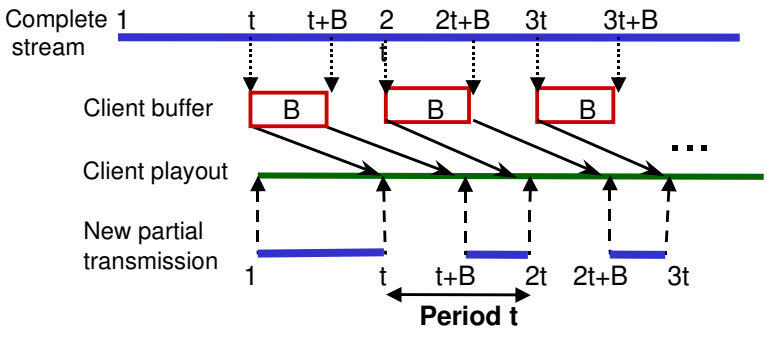
Case t ≤ B: First t frames from proxy, remainder from complete stream



Computer Science UMASS

PBR Patching (Continued)

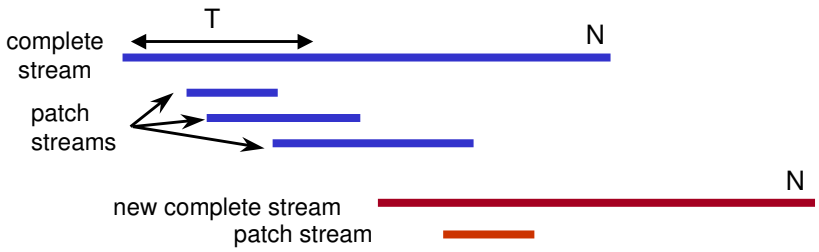
Case t > B :



Computer Science UMASS

PBR Patching uses Thresholding

- threshold, T
- client arrives >= T after start of complete stream:
  - start new complete stream
  - else patch into most recently started stream



L. Gao, D. Towsley, "Supplying instantaneous VoD services using controlled multicast," ICMCS 99.

Tradeoffs

- Advantages
  - Bandwidth savings for popular videos (v/s unicast)
  - Patching: initiation latency same as unicast
- Disadvantages
  - Clients need to listen to multiples channels simulateneously
    - Need extra bandwidth at the last mile!
  - Increased buffer requirements at the client
    - Counter argument: buffers are cheap