

TSIN02: Internetworking

Lecture 9: Streaming video

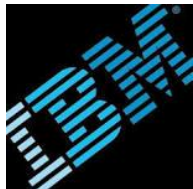
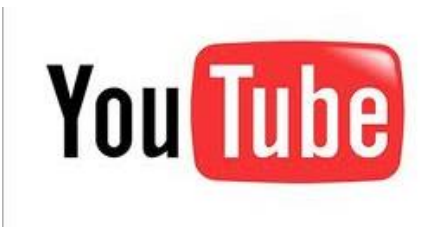
Guest Instructor: Niklas Carlsson

Email: niklas.carlsson@liu.se

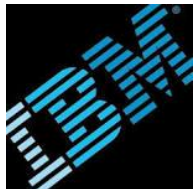
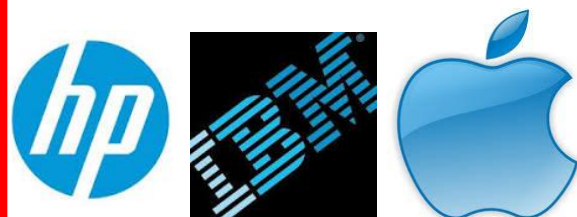
Web: <http://www.ida.liu.se/~nikca/>

These notes have been derived and adapted based on various online resources, including slides from Anirban Mahanti, Carey Williamson, and "*Computer Networking: A Top Down Approach*", by Jim Kurose and Keith Ross, Addison-Wesley.

Today's service/company landscape include ...



Today's service/company landscape include ...



Today's service/company landscape include ...

Equipment manufacturers
(also sell services and help
Operate networks)



Today's service/company landscape include ...

Network operators

Equipment manufacturers
(also sell services and help
Operate networks)



Today's service/company landscape include ...



Enterprise solutions
and network service
(e.g., data center
solutions and cloud
providers)

Today's service/company landscape include ...

Content delivery networks



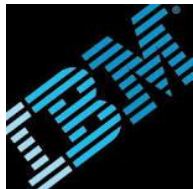
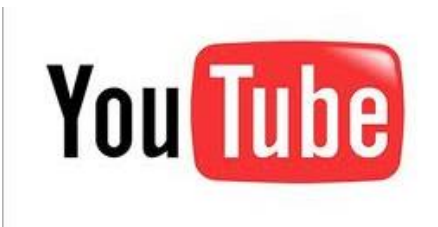
Enterprise solutions and network service (e.g., data center solutions and cloud providers)

Today's service/company landscape include ...



End user services (e.g., web-based social networks, search, communication, and streaming)

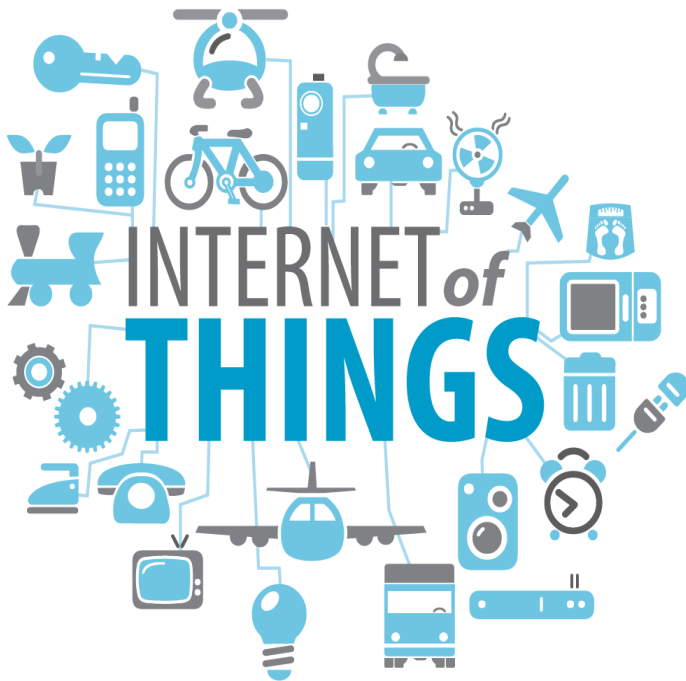
Today's service/company landscape include ...
However, "boundaries" are getting fuzzy ...



Some common applications today ...

- ❑ World Wide Web (WWW)
- ❑ Remote login (telnet, rlogin, ssh)
- ❑ File transfer
- ❑ Peer-to-peer file sharing
- ❑ Cloud computing/services
- ❑ Instant messaging (chat, text messaging, etc.)
- ❑ Live and video-on-demand streaming
- ❑ Internet phone (Voice-Over-IP)
- ❑ Distributed games

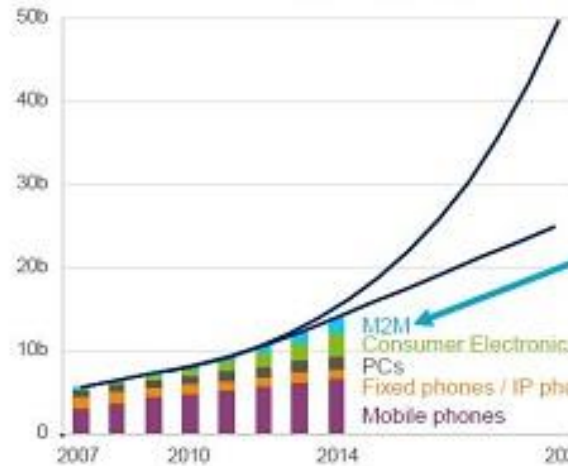
... and tomorrow



NEW DEVICES AND NEW INDUSTRIES BRING NEW BUSINESS OPPORTUNITIES



Connected Devices Worldwide



Addressing Industries

Traffic systems, Automotive
Transport and logistics
Utilities – smart grid
Security – connected buildings
Home appliances
Medical automation, Remote healthcare
ATM, Point of sale, Vending
Critical infrastructures
Monitoring and control

More devices per person

eBook readers, Music players, DVD players, Gaming devices, Cameras, Home appliances, In-vehicle entertainment etc.

New telecom cycle: 10x devices, 10x industries

The 2020 vision

- ❑ Everything that can be connected will be connected
 - 50B devices (perhaps more like 500B ...)
- ❑ IoT and smart cities
 - Machine-to-machine
- ❑ High-definition 3D streaming to heterogeneous clients

Requirements and quality of service

- Quality of Service (QoS)
 - Real-time requirements (e.g., latency, jitter)
 - Loss/stall requirements (e.g., drop rates, late packets)
 - Bandwidth requirements (e.g., throughput)
 - Service availability
- Quality of Experience (QoE)
 - Measure of the users quality of experience
 - Multimedia: Most negatively effected by stalls

Requirements and quality of service

- ❑ Quality of Service (QoS)
 - Real-time requirements (e.g., latency, jitter)
 - Loss/stall requirements (e.g., drop rates, late packets)
 - Bandwidth requirements (e.g., throughput)
 - Service availability
- ❑ Quality of Experience (QoE)
 - Measure of the users quality of experience
 - Multimedia: Most negatively effected by stalls



Scalable Content Delivery

- Use of Internet for content delivery is massive ... and becoming more so (e.g., majority of all IP traffic is video streaming content)
- Variety of approaches: HTTP-based Adaptive Streaming (HAS), broadcast/multicast, batching, replication/caching (e.g. CDNs), P2P, peer-assisted, ...
- In these slides, we only provide a few high-level examples

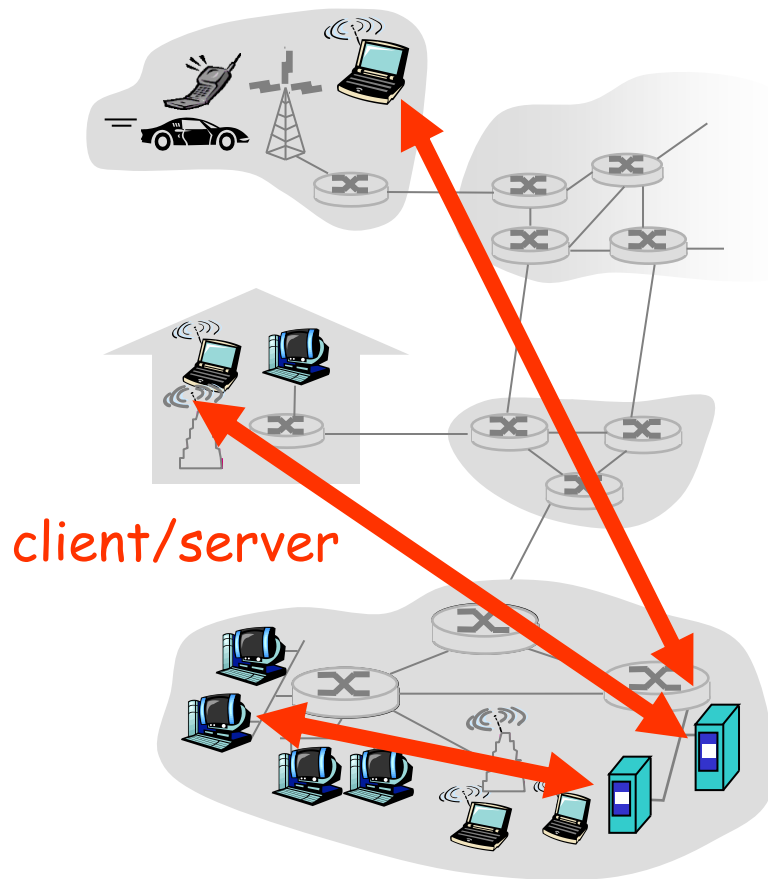
Service models

- ❑ Client-server (one-to-one)
- ❑ Peer-to-peer (machines can act as both client and server)

- ❑ Multicast/broadcast (one-to-many and many-to-many)
 - Application layer, IP-based, and down at the MAC-layer
- ❑ Replication: ISP-based caching, CDNs, cloud, and other third-party solutions

Client-server architecture

Client/server model has well-defined roles.



server:

- always-on host
- permanent IP address
- server farms for scaling

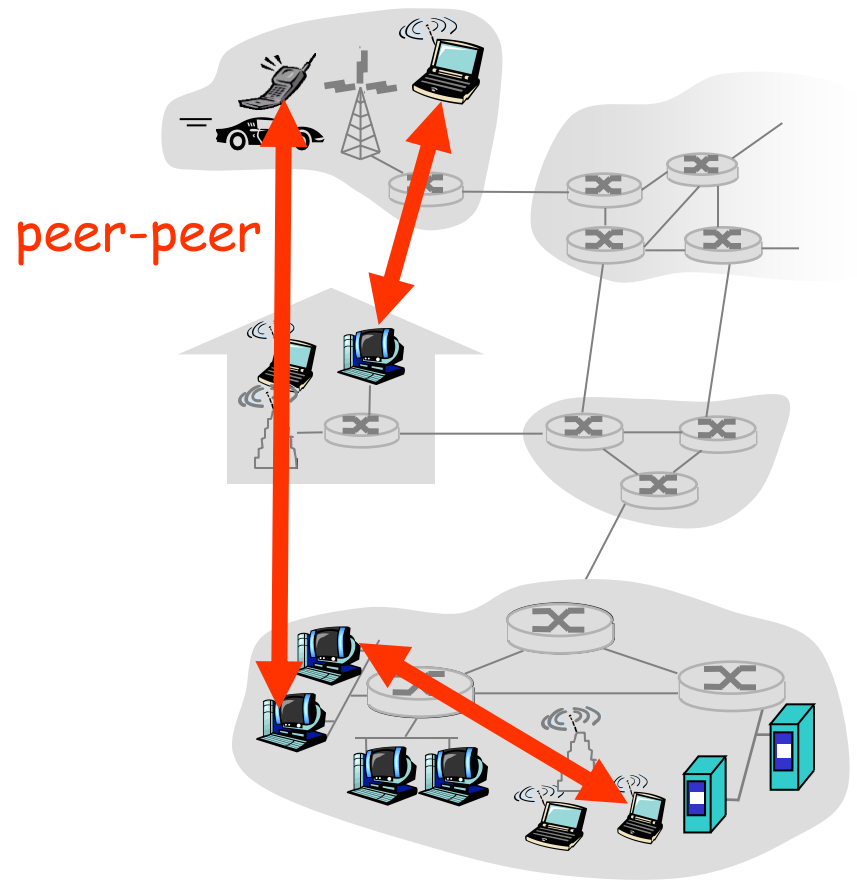
clients:

- communicate with server
- may be intermittently connected
- may have dynamic IP addresses
- do not communicate directly with each other

Pure P2P architecture

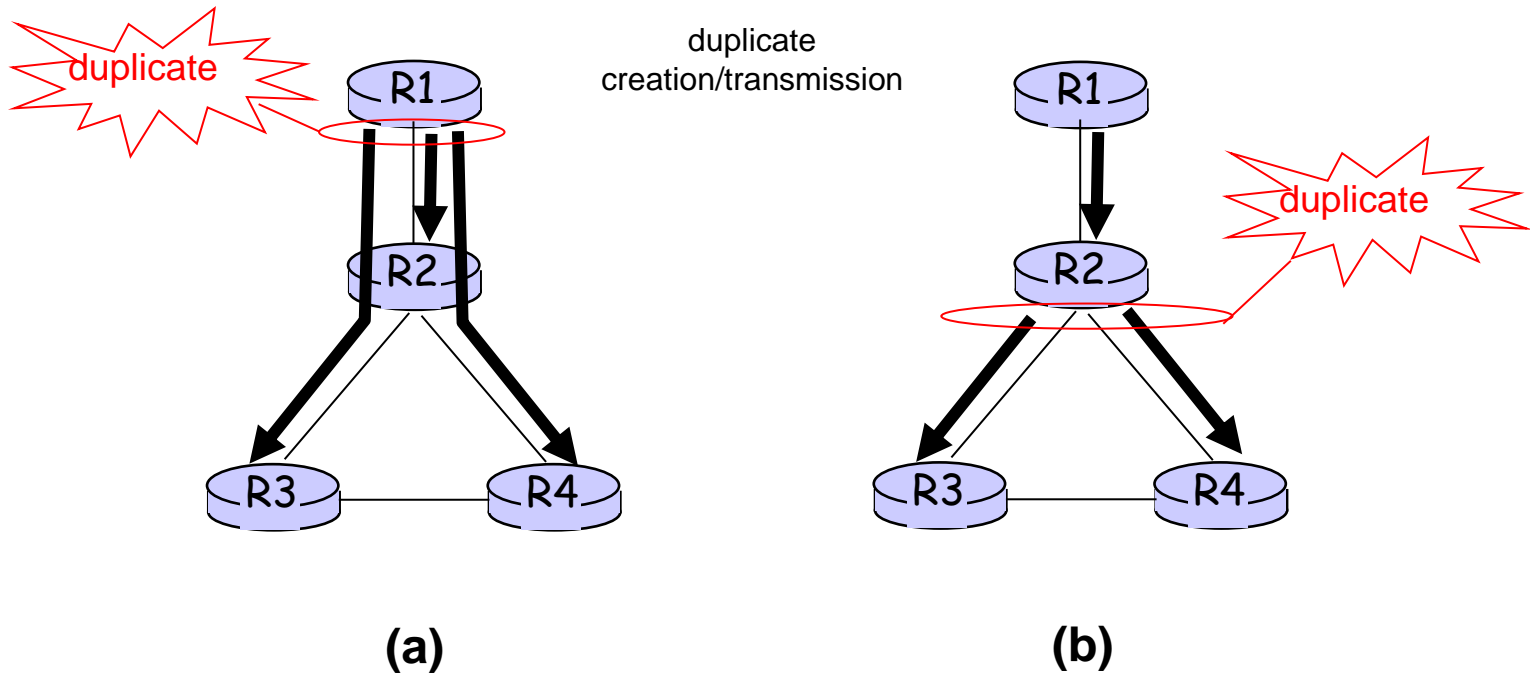
No fixed clients or servers: Each host can act as both client and server at any time

- ❑ *no* always-on server
- ❑ arbitrary end systems directly communicate
- ❑ peers are intermittently connected and change IP addresses



One-to-many delivery

Multicast/Broadcast



Source-duplication versus in-network duplication.
(a) source duplication, (b) in-network duplication

Also, application-layer multicast ...

Evolved Multimedia Broadcast/Multicast Service (eMBMS) in LTE-advanced

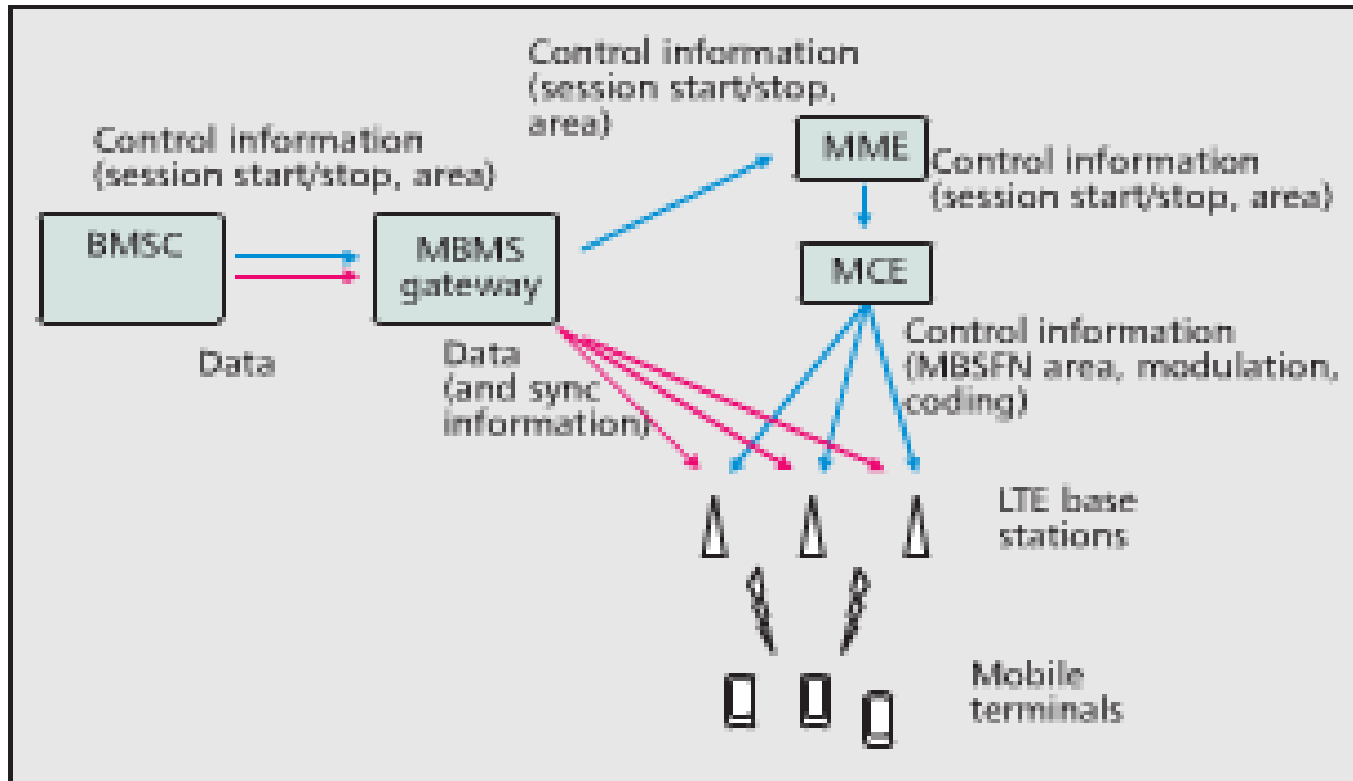


Figure 4. RAN architecture for SFN across LTE base stations.

□ Separation of control plane and data plane

Image from: Lecompte and Gabin, Evolved Multimedia Broadcast/Multicast Service (eMBMS) in LTE-Advanced: Overview and Rel-11 Enhancements, IEEE Communications Magazine, Nov. 2012.

Evolved Multimedia Broadcast/Multicast Service (eMBMS) in LTE-advanced

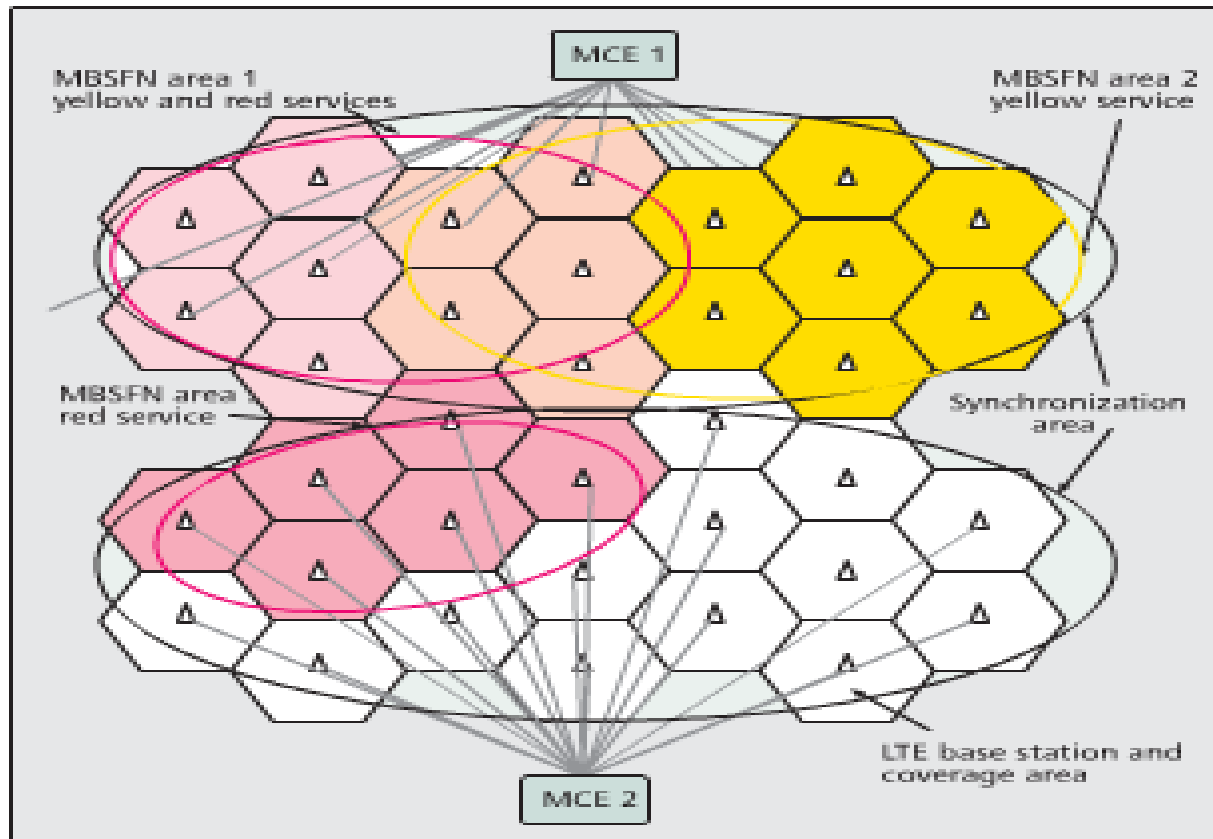


Figure 5. Example with two MBMS services with different services areas.

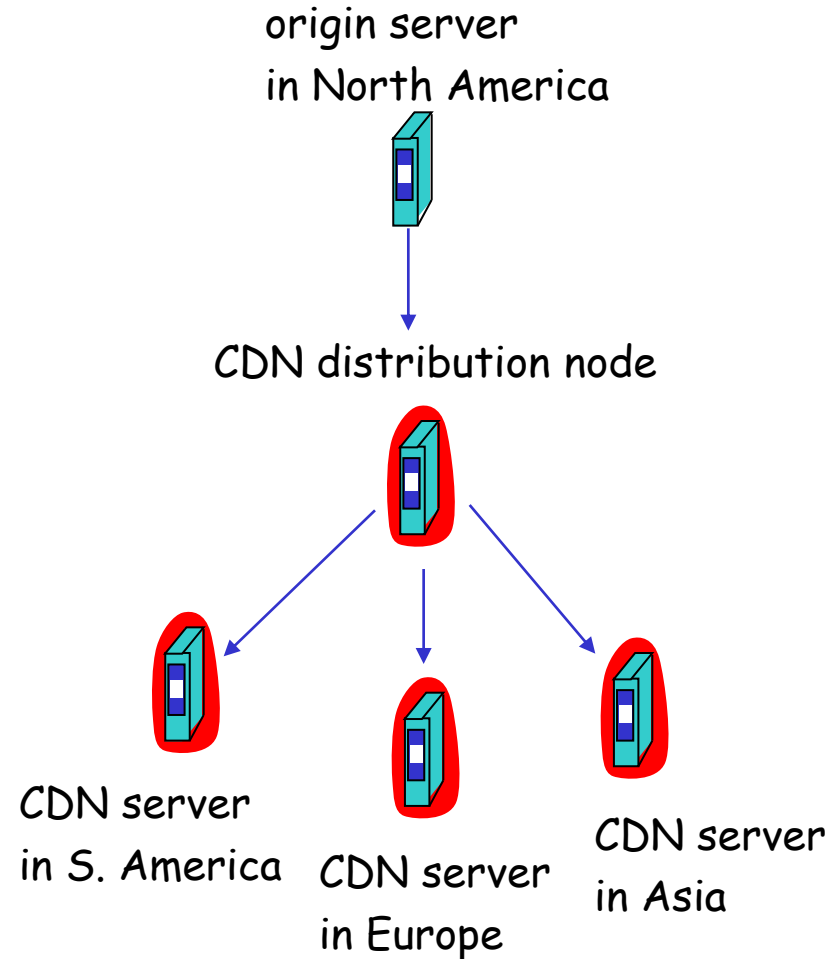
□ MBMSFN and use of services areas

Image from: Lecompte and Gabin, Evolved Multimedia Broadcast/Multicast Service (eMBMS) in LTE-Advanced: Overview and Rel-11 Enhancements, IEEE Communications Magazine, Nov. 2012.

Content distribution networks (CDNs)

Content replication

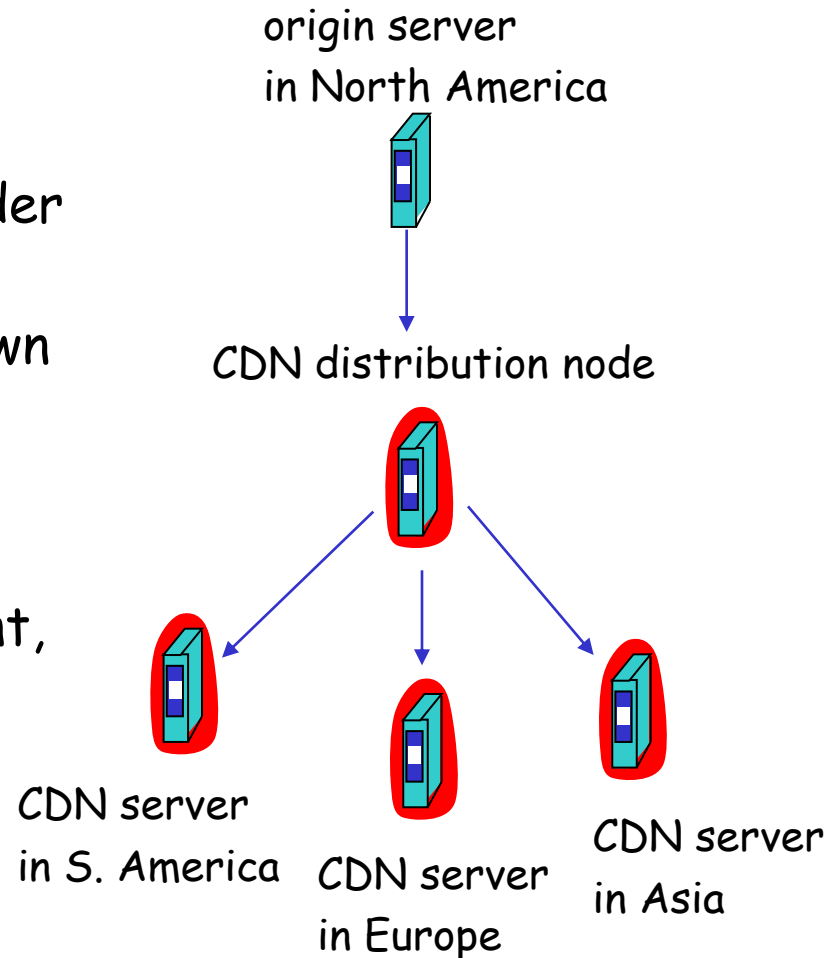
- ❑ replicate content at hundreds of servers throughout Internet (often in edge/access network)
- ❑ content "close" to user reduce impairments (loss, delay) of sending content over long paths



Content distribution networks (CDNs)

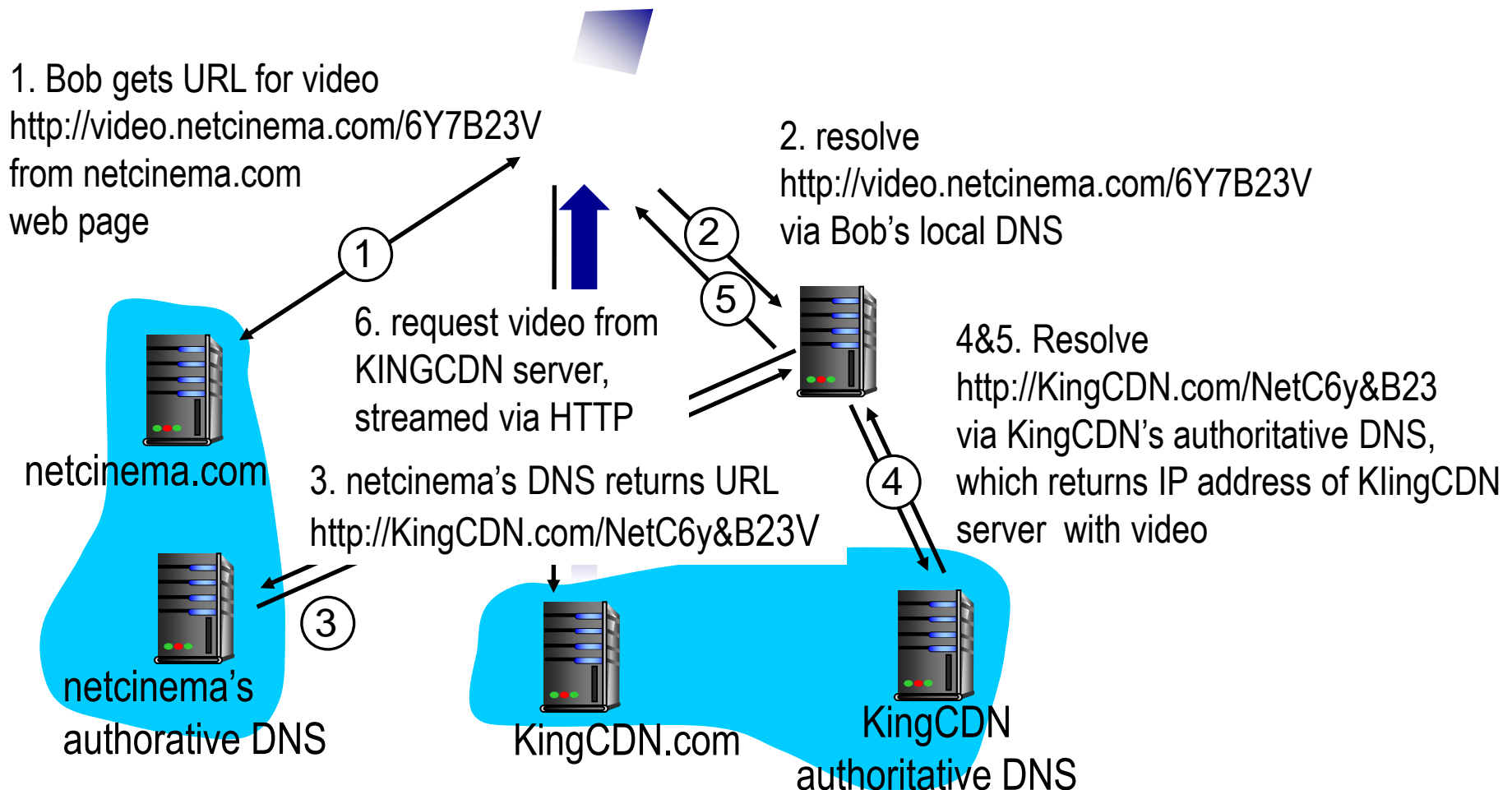
Content replication

- ❑ CDN (e.g., Akamai, Limewire) customer is the content provider (e.g., CNN)
- ❑ Other companies build their own CDN (e.g., Google)
- ❑ CDN replicates customers' content in CDN servers.
- ❑ When provider updates content, CDN updates servers



CDN: "simple" content access scenario

- Bob (client) requests video <http://video.netcinema.com/6Y7B23V>
- video stored in CDN at <http://KingCDN.com/NetC6y&B23V>



Multimedia Networking Applications

Classes of MM applications:

Multimedia Networking Applications

Classes of MM applications:

- 1) Streaming stored audio and video
- 2) Streaming live audio and video
- 3) Real-time interactive audio and video

Consider first ...

Streaming Stored Multimedia

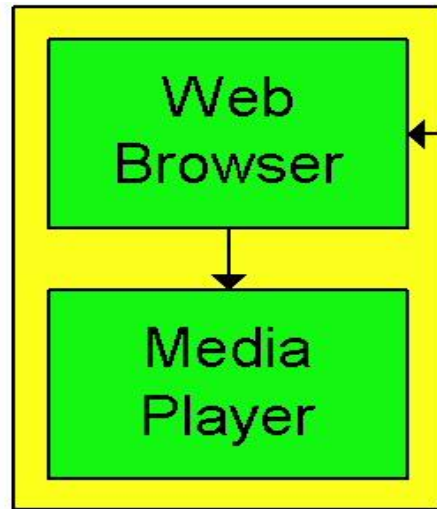
application-level
streaming techniques for
making the best out of
best effort service:

- client-side buffering
- use of UDP versus TCP
- multiple encodings of multimedia

Media Player

- jitter removal
- decompression
- error concealment
- graphical user interface
w/ controls for
interactivity

Internet multimedia: simplest approach



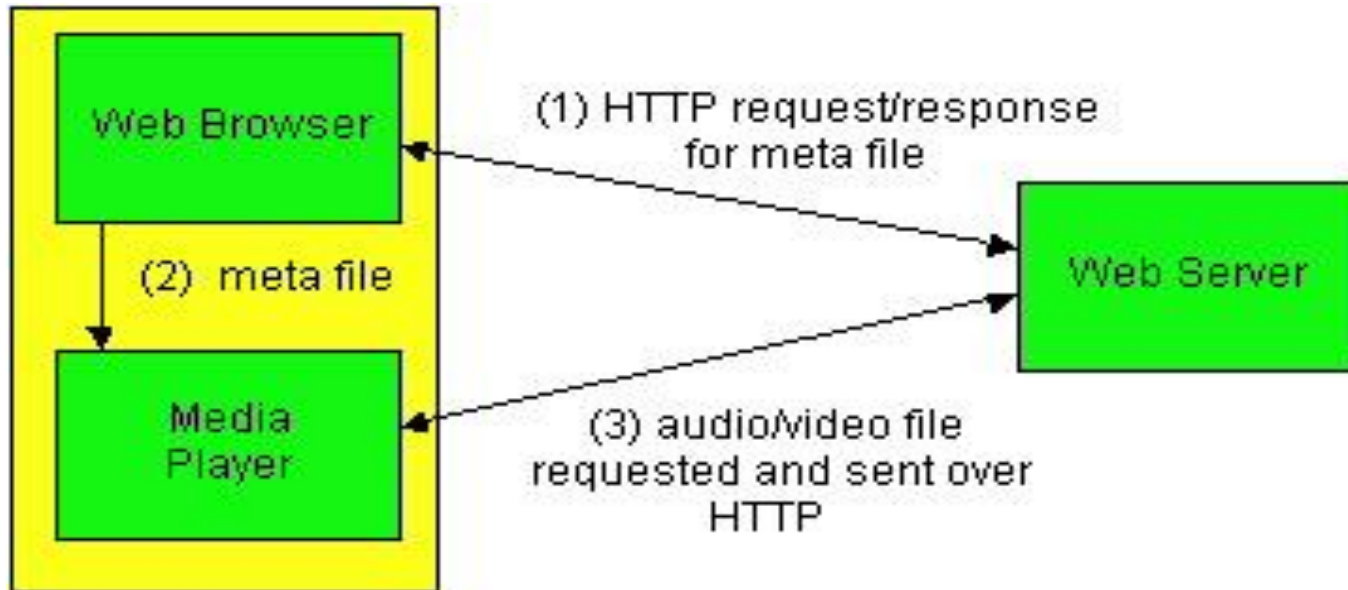
client

- ❑ audio or video stored in file
- ❑ files transferred as HTTP object
 - received in entirety at client
 - then passed to player

audio, video is downloaded, not streamed:

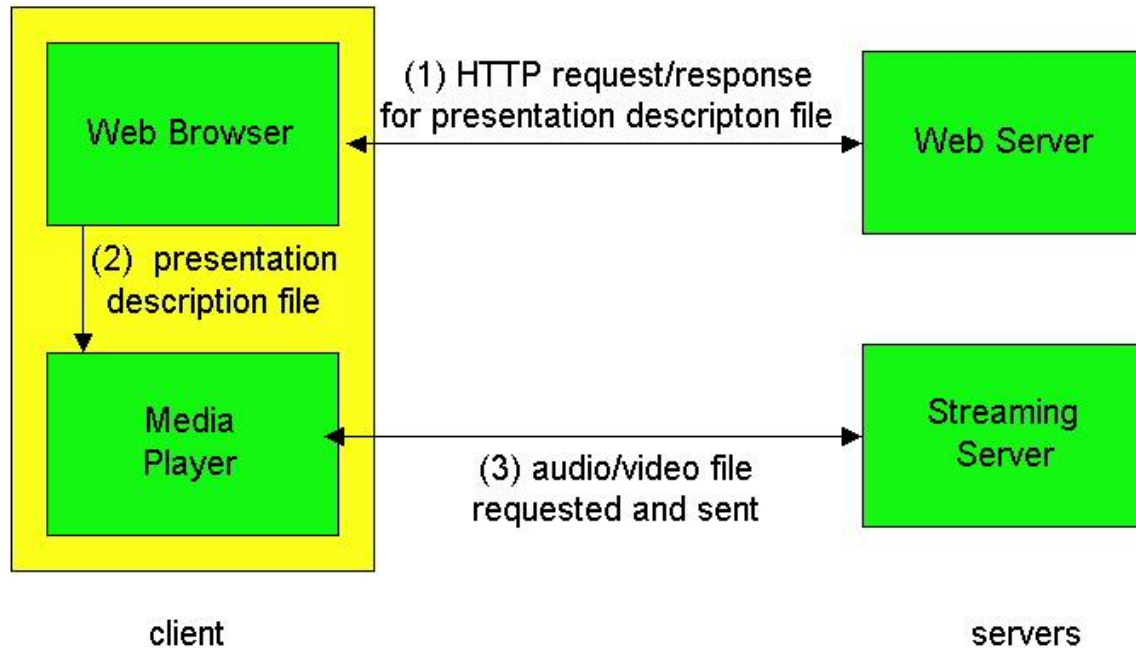
- ❑ long delays until playout, since no pipelining!

Progressive Download



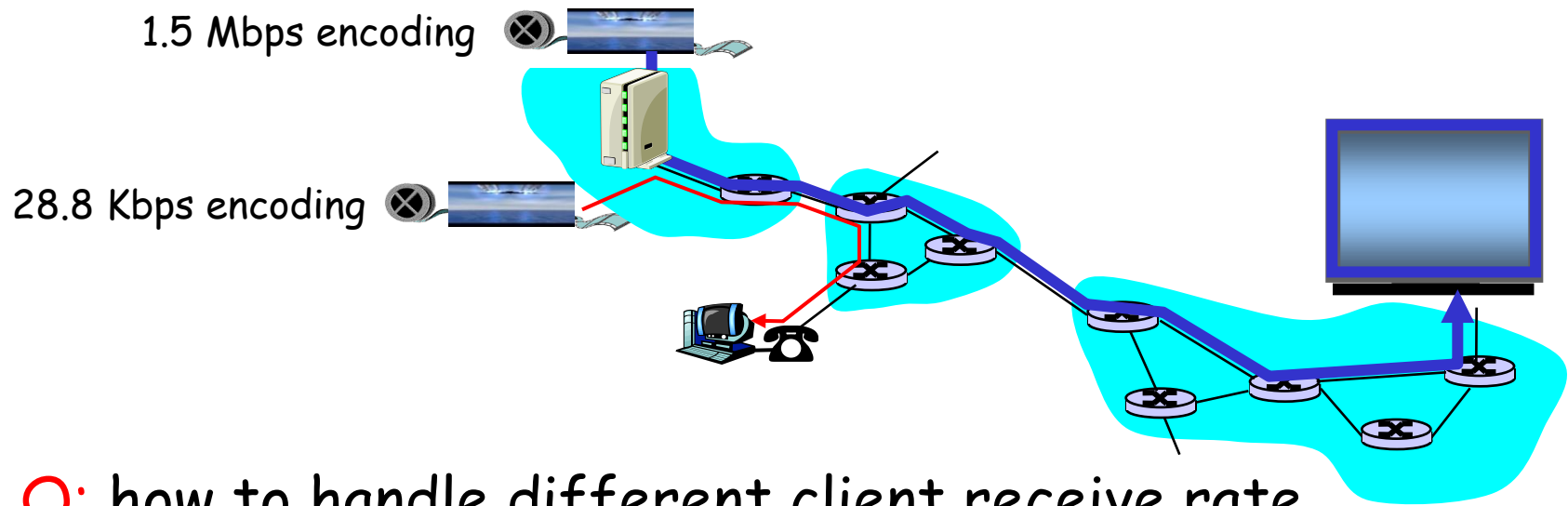
- ❑ browser retrieves **metafile** using HTTP GET
- ❑ browser launches player, passing metafile to it
- ❑ media player contacts server directly
- ❑ server **downloads** audio/video to player

Streaming from a Streaming Server



- ❑ This architecture allows for non-HTTP protocol between server and media player
- ❑ Can also use UDP instead of TCP.

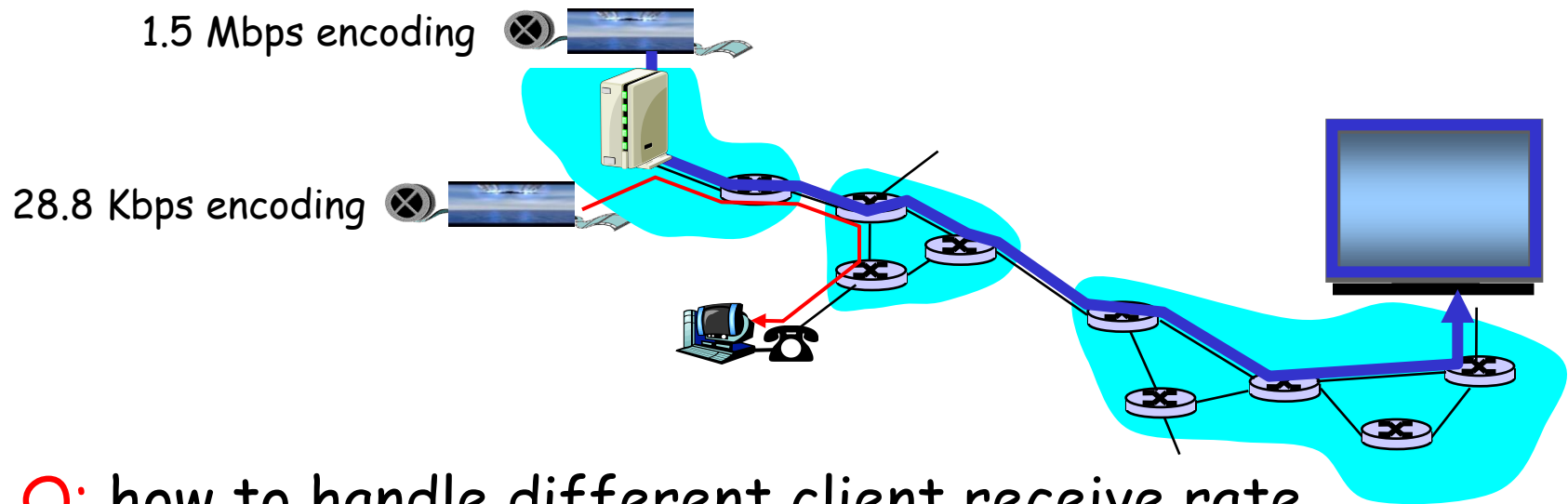
Streaming Multimedia: client rate(s)



Q: how to handle different client receive rate capabilities?

- 28.8 Kbps dialup
- 100 Mbps Ethernet

Streaming Multimedia: client rate(s)



Q: how to handle different client receive rate capabilities?

- 28.8 Kbps dialup
- 100 Mbps Ethernet

A1: server stores, transmits multiple copies of video, encoded at different rates

A2: layered and/or dynamically rate-based encoding

Streaming Multimedia: UDP or TCP?

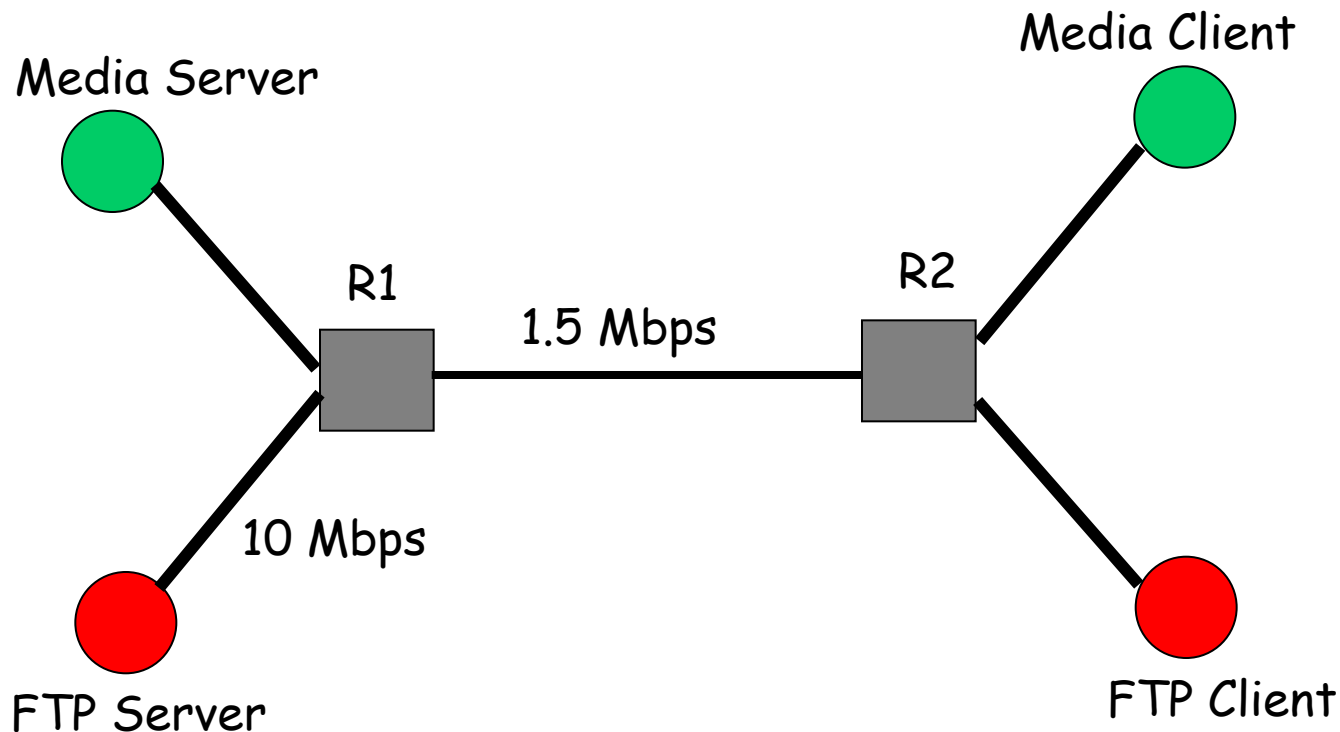
UDP

- ❑ server sends at rate appropriate for client (oblivious to network congestion !)
 - often send rate = encoding rate = constant rate
 - then, fill rate = constant rate - packet loss
- ❑ short playout delay (2-5 seconds) to compensate for network delay jitter
- ❑ error recover: time permitting

TCP

- ❑ send at maximum possible rate under TCP
- ❑ fill rate fluctuates due to TCP congestion control
- ❑ larger playout delay: smooth TCP delivery rate
- ❑ HTTP/TCP passes more easily through firewalls

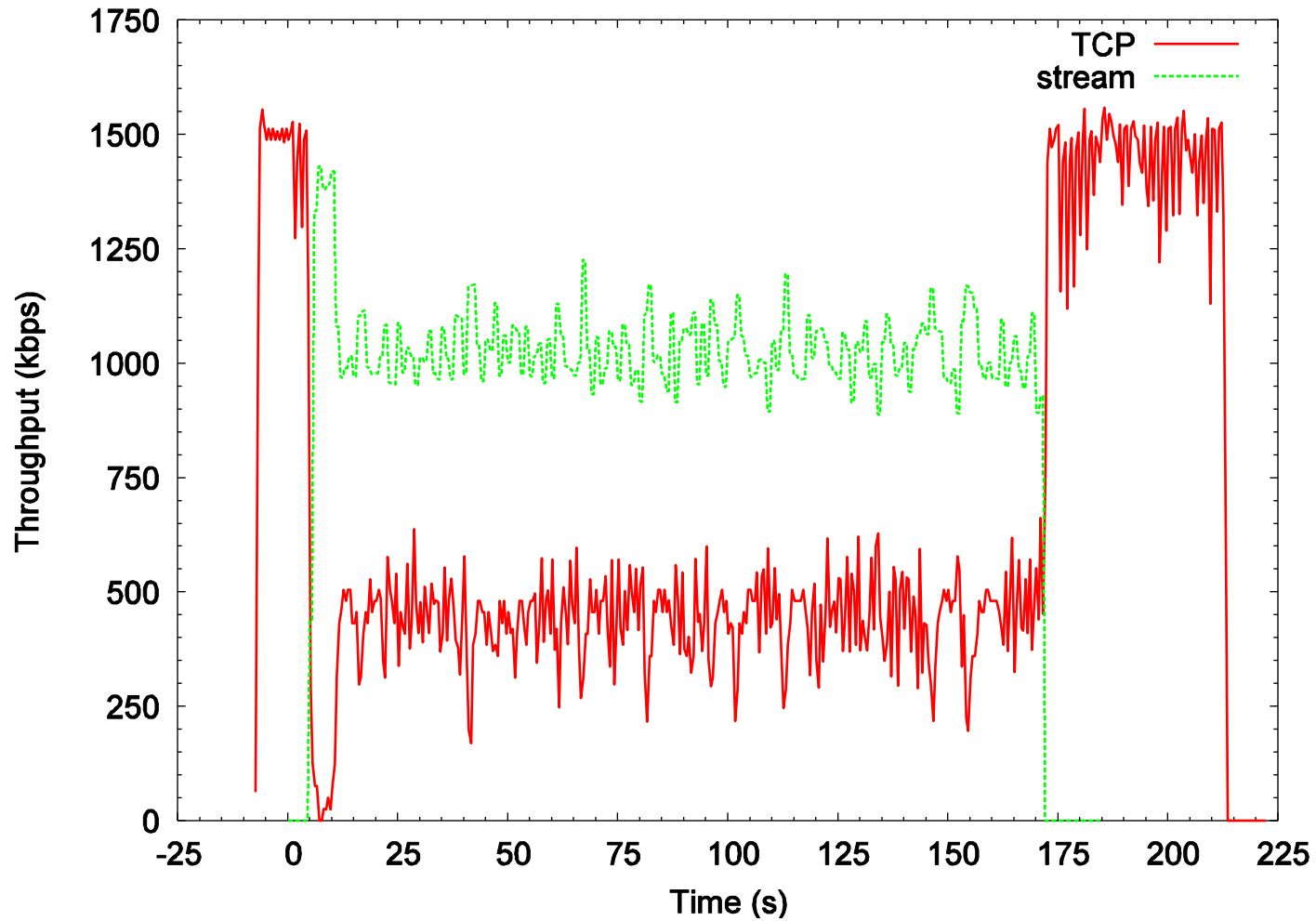
Fairness of UDP Streams (1/2)



- R1-R2 is the bottleneck link
- Streaming uses UDP at the transport layer; requested media encoded at 1 Mbps
- What fraction of the bottleneck is available to FTP?

Credit: MSc thesis work by [Sean Boyden](#) (2006)

Fairness of RealVideo Streams (2/2)



A protocol family for streaming

- RTSP
- RTP
- RTCP

RTSP: out-of-band control

RTSP messages sent out-of-band:

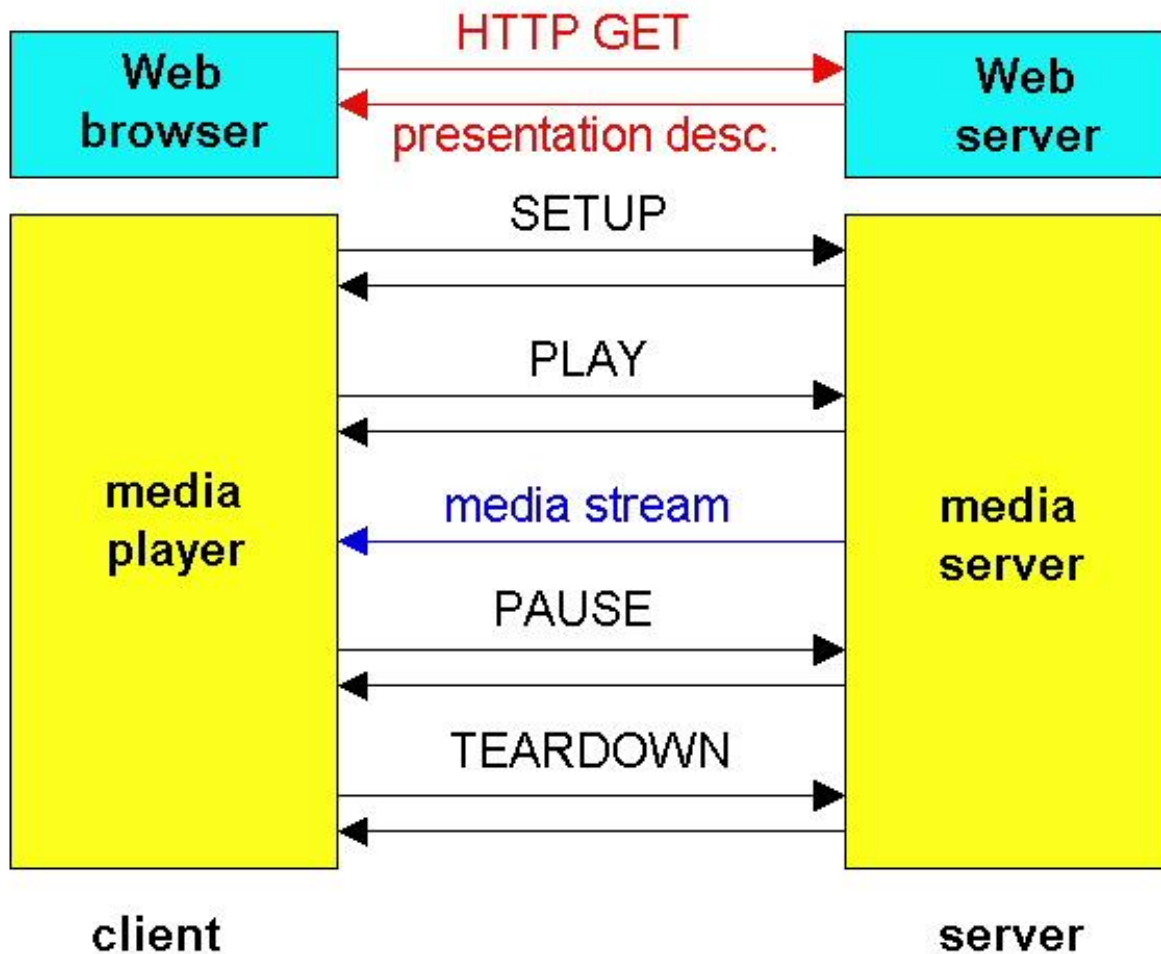
- ❑ RTSP control messages use different port numbers than media stream: out-of-band.
 - port 554
- ❑ media stream is considered "in-band".

RTSP Example

Scenario:

- ❑ metafile communicated to web browser
- ❑ browser launches player
- ❑ player sets up an RTSP control connection, data connection to streaming server

RTSP Operation



Real-Time Protocol (RTP)

- ❑ RTP specifies packet structure for packets carrying audio, video data
- ❑ RFC 3550
- ❑ RTP runs in end systems
- ❑ RTP packets encapsulated in UDP segments

RTP Header



RTP Header

Payload Type (7 bits): Indicates type of encoding currently being used. If sender changes encoding in middle of conference, sender informs receiver via payload type field.

- Payload type 0: PCM mu-law, 64 kbps
- Payload type 3, GSM, 13 kbps
- Payload type 7, LPC, 2.4 kbps
- Payload type 26, Motion JPEG
- Payload type 31. H.261
- Payload type 33, MPEG2 video

Sequence Number (16 bits): Increments by one for each RTP packet sent, and may be used to detect packet loss and to restore packet sequence.

Real-time Control Protocol (RTCP)

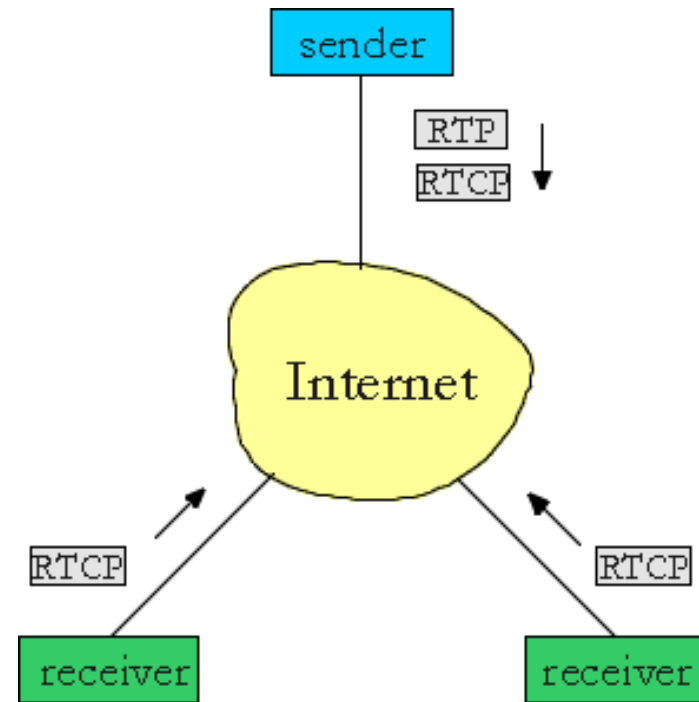
Receiver report packets:

- ❑ fraction of packets lost, last sequence number, average interarrival jitter

Sender report packets:

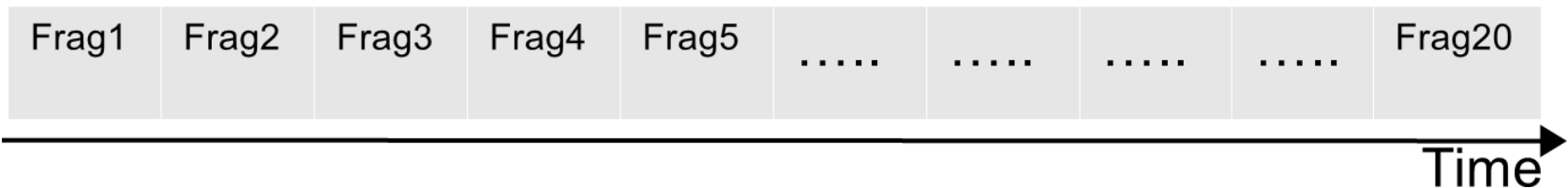
- ❑ SSRC of RTP stream, current time, number of packets sent, number of bytes sent

- ❑ feedback can be used to control performance



- ❑ RTCP attempts to limit its traffic to 5% of session bandwidth

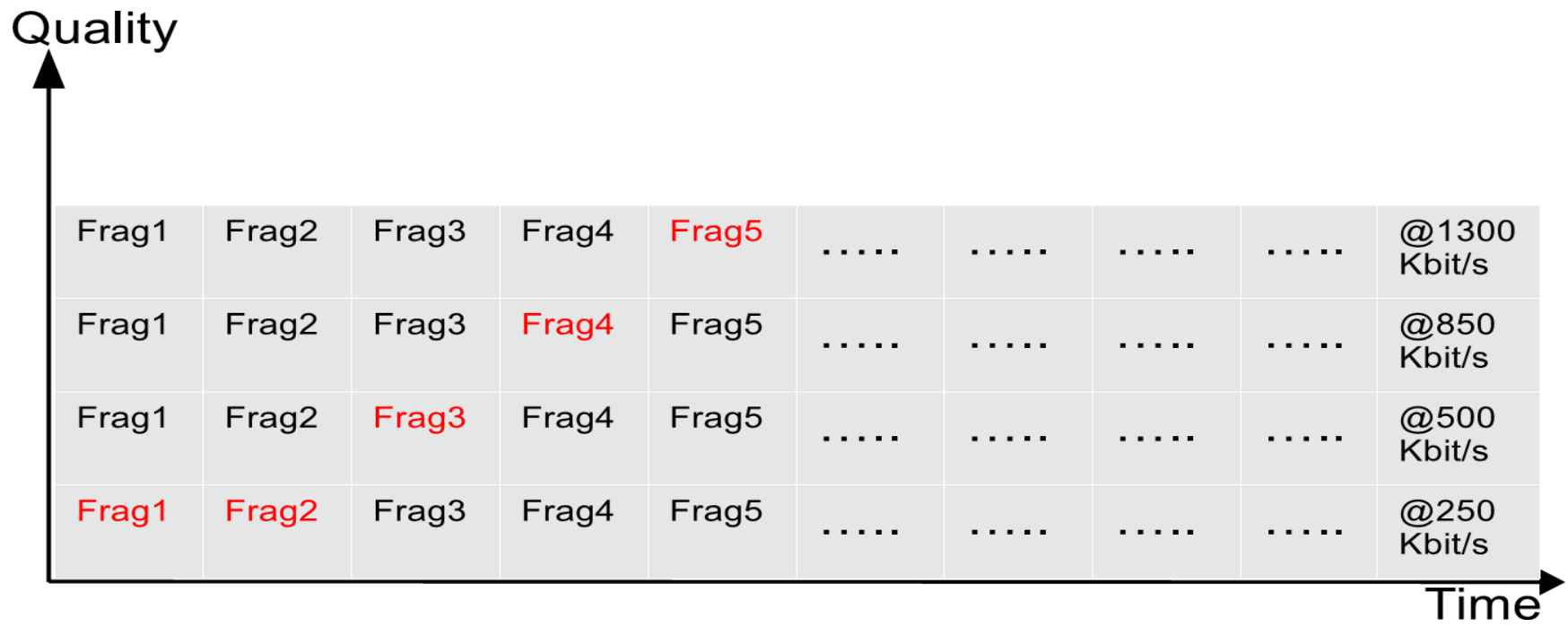
HTTP-based streaming



□ HTTP-based streaming

- Allows easy caching, NAT/firewall traversal, etc.
- Use of TCP provides natural bandwidth adaptation
- Split into fragments, download sequentially
- Some support for interactive VoD

HTTP-based adaptive streaming (HAS)



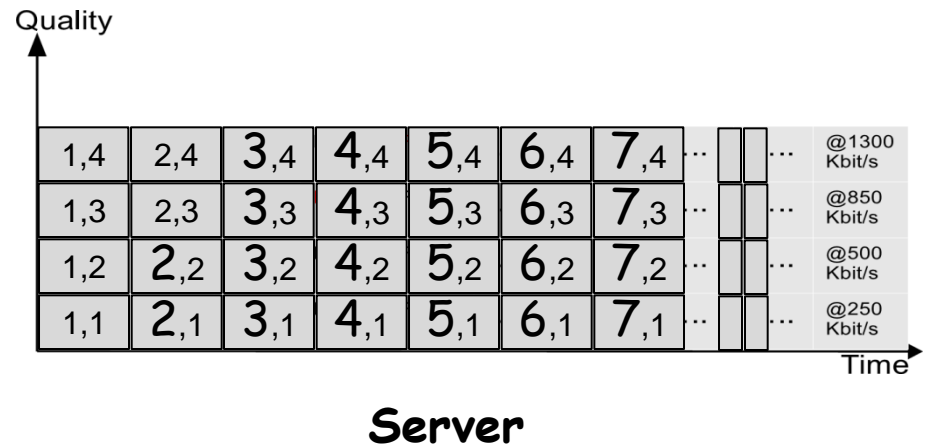
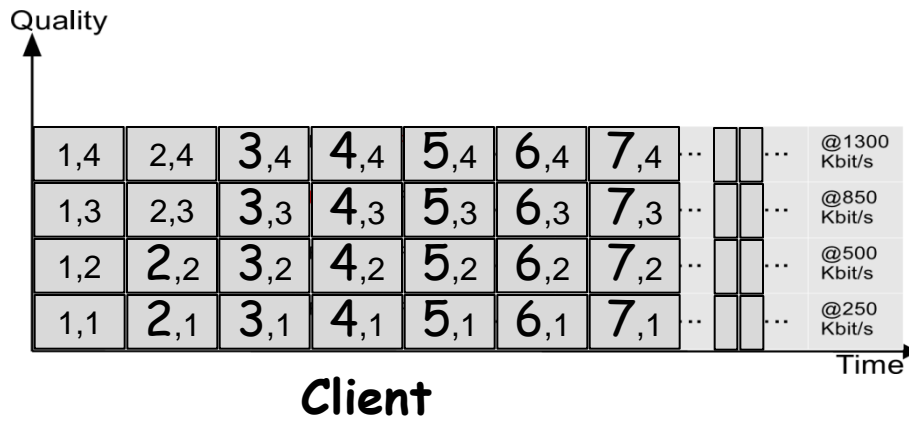
□ HTTP-based adaptive streaming

- Multiple encodings of each fragment (defined in manifest file)
- Clients adapt quality encoding based on (buffer and network) conditions

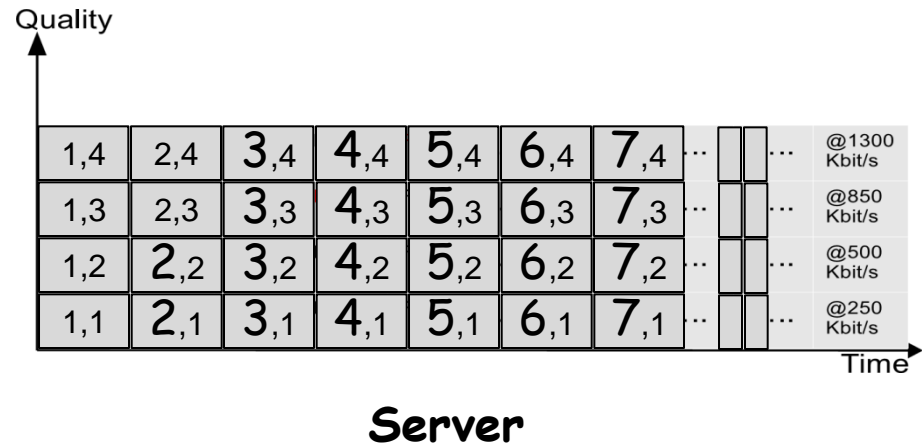
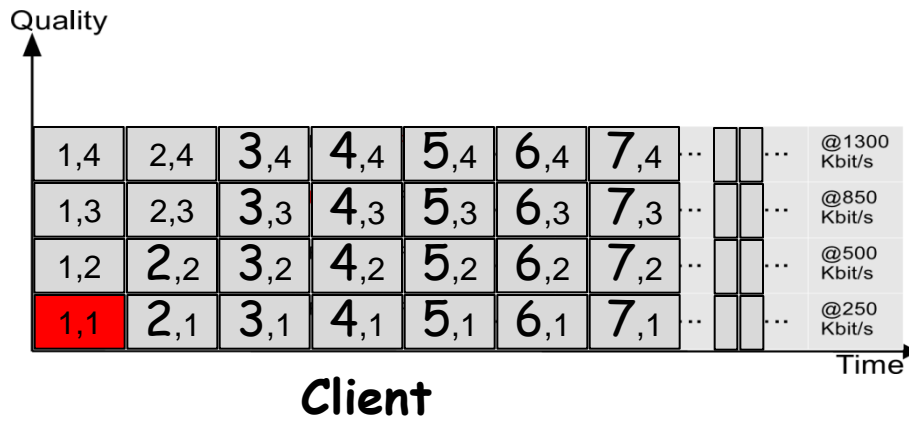
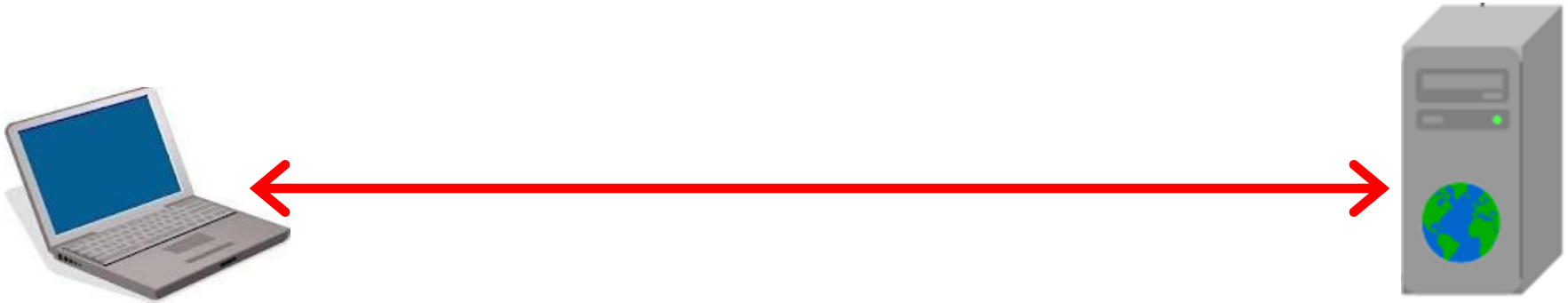
Chunk-based streaming

- ❑ Chunks begin with keyframe so independent of other chunks
- ❑ Playing chunks in sequence gives seamless video
- ❑ Hybrid of streaming and progressive download:
 - Stream-like: sequence of small chunks requested as needed
 - Progressive download-like: HTTP transfer mechanism, stateless servers

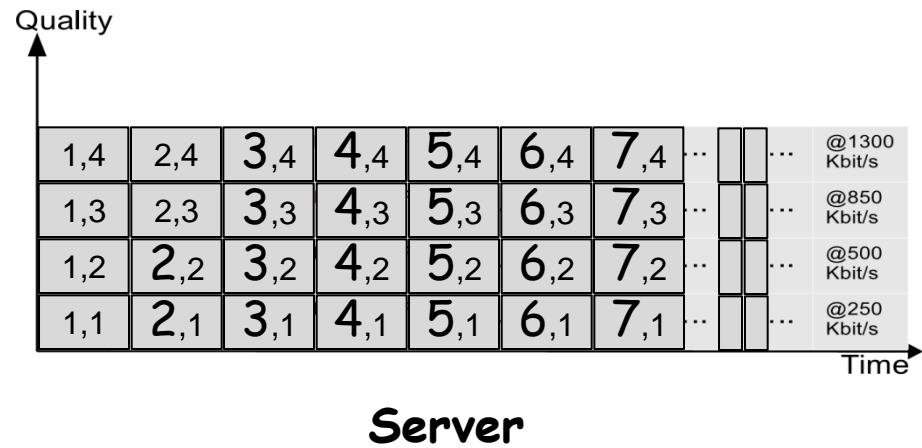
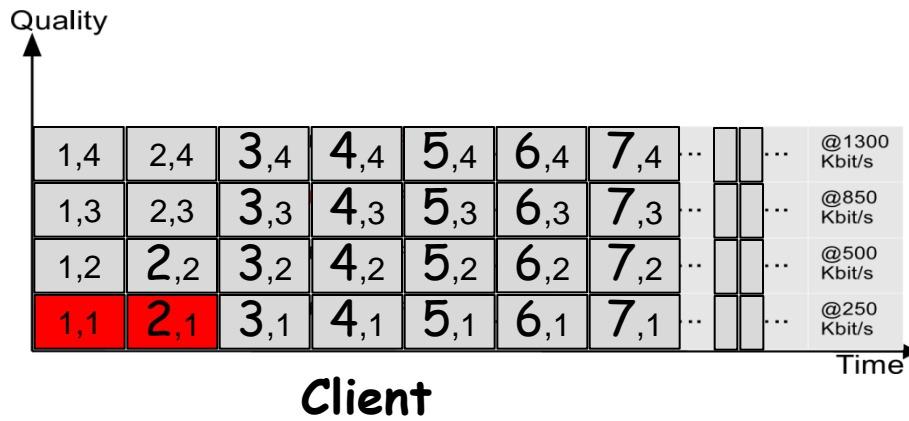
Example



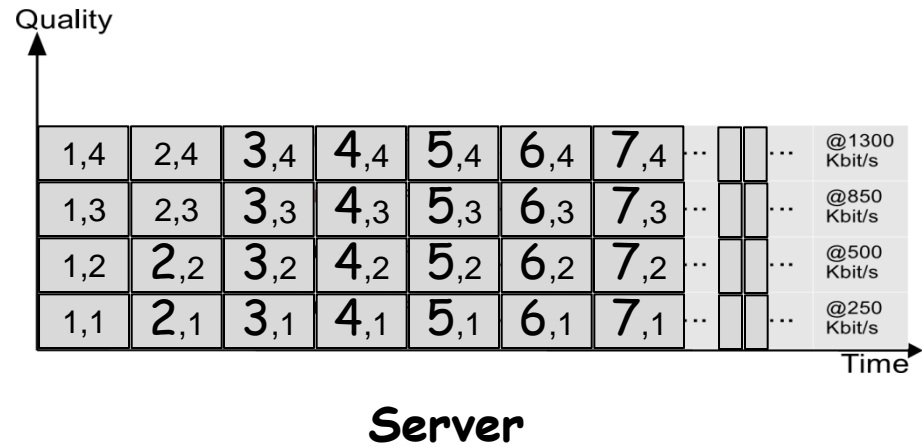
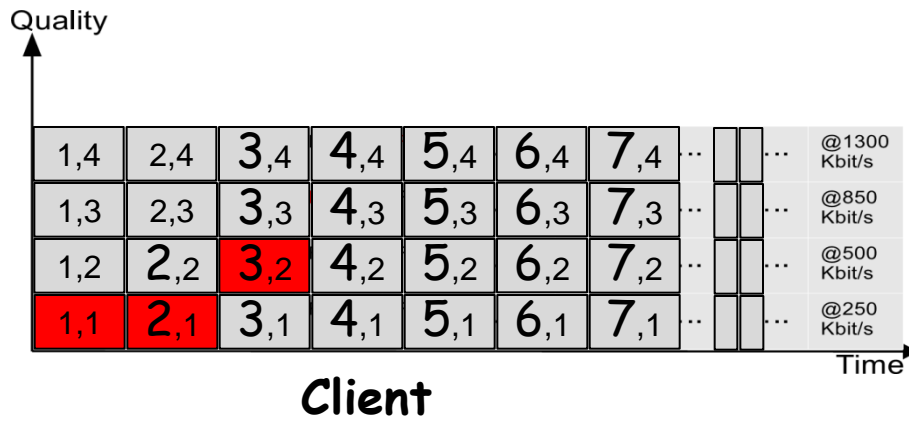
Example



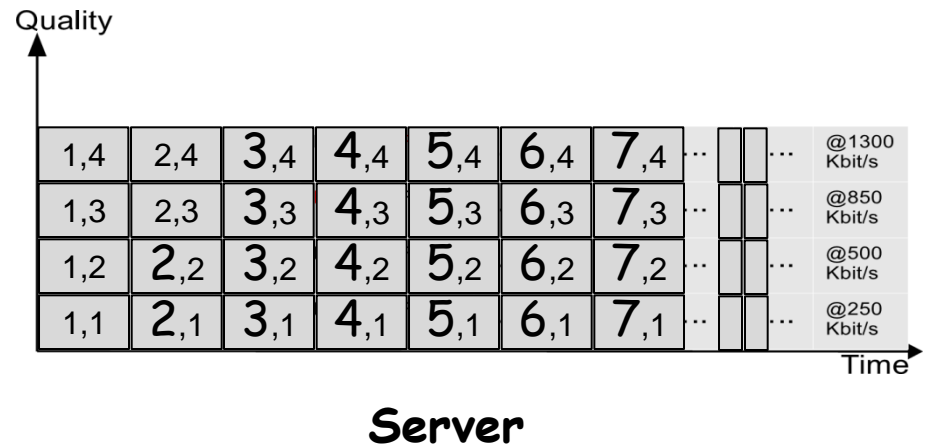
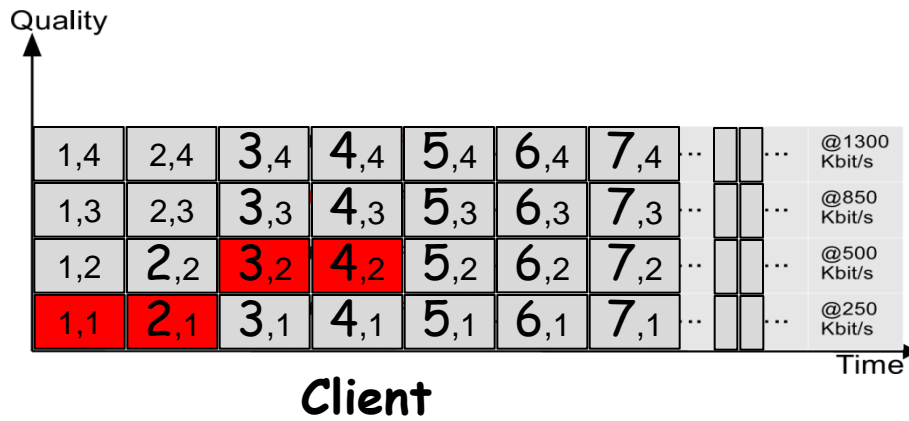
Example



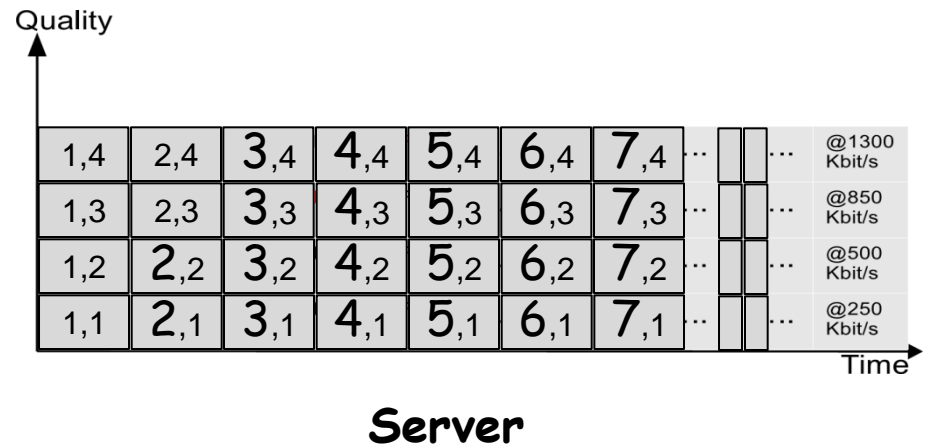
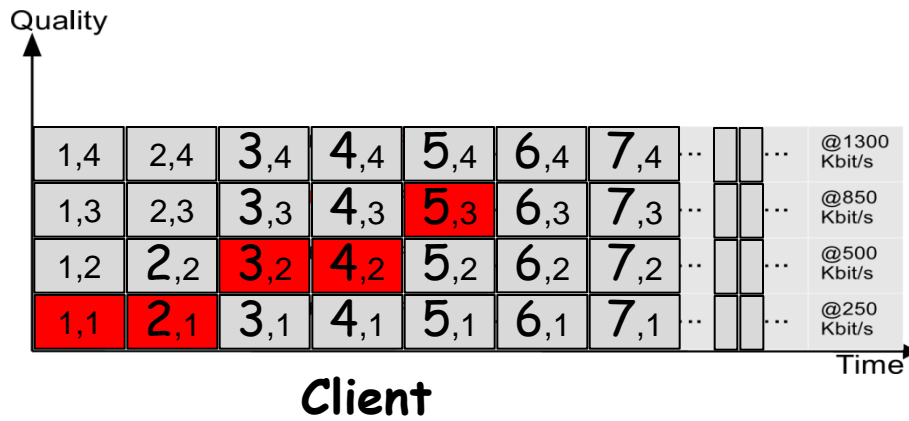
Example



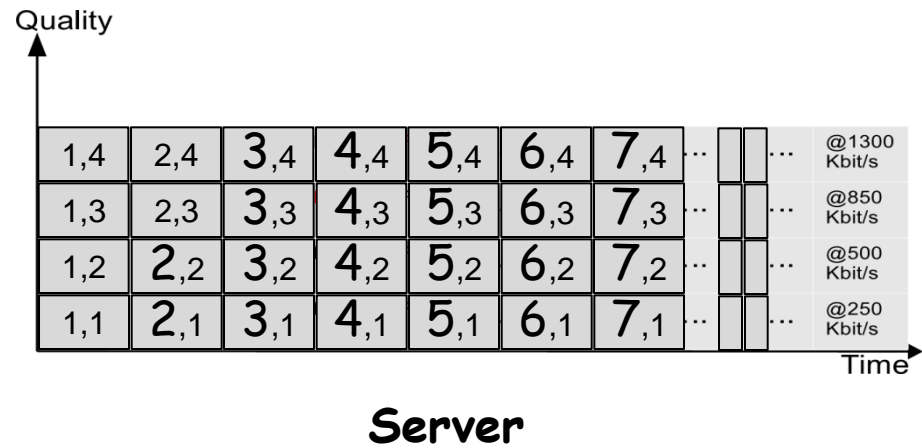
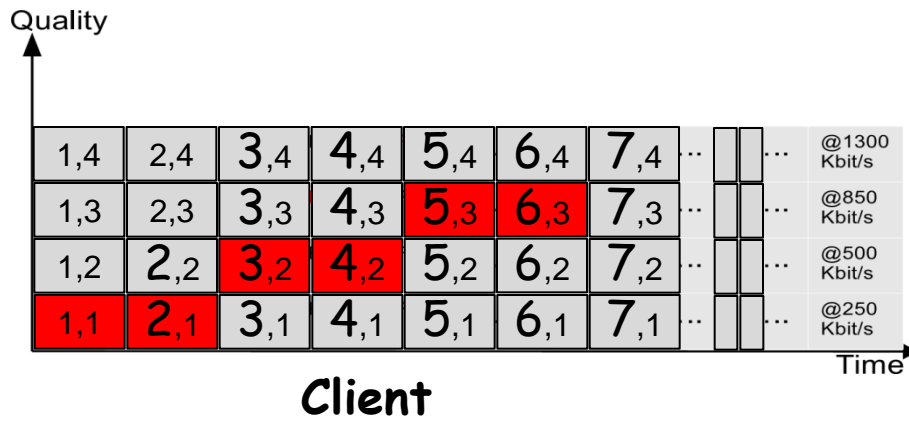
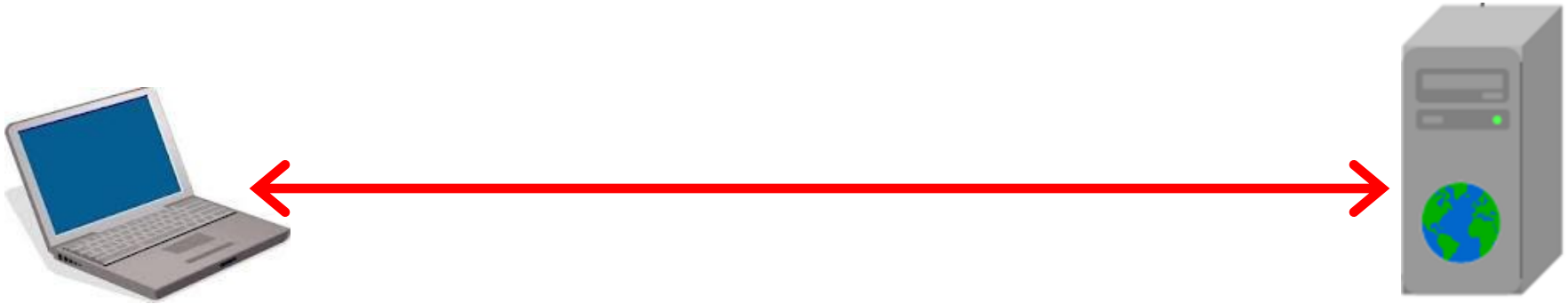
Example



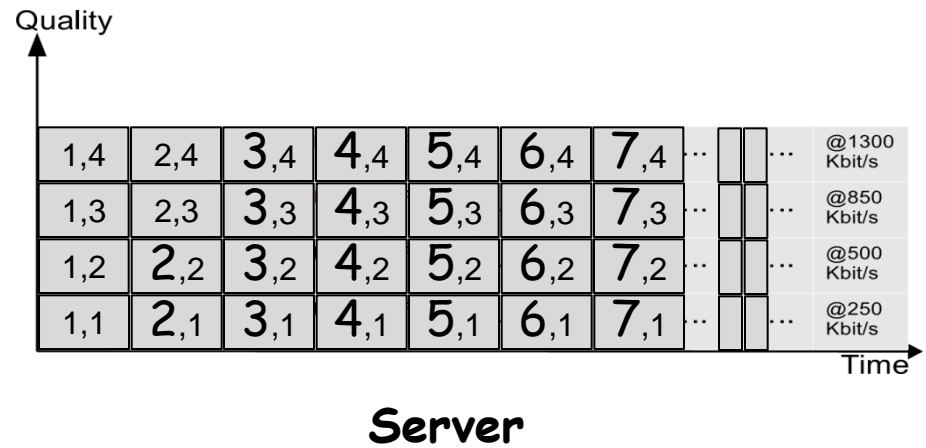
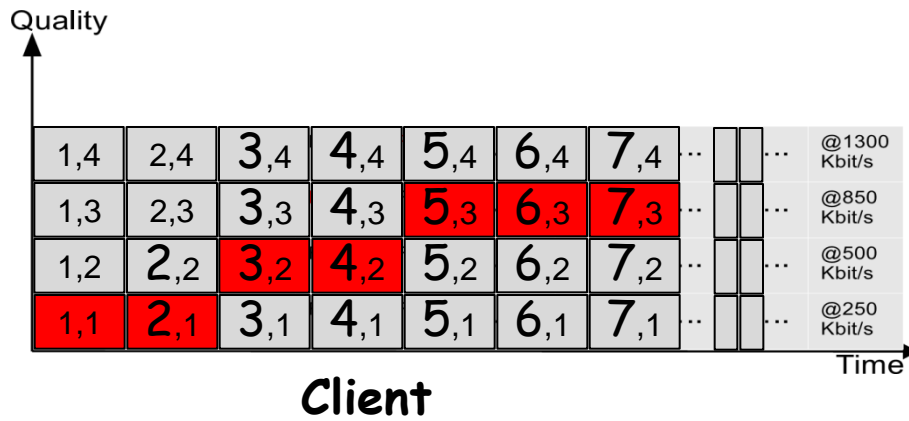
Example



Example



Example



HTTP-based Adaptive Streaming (HAS)





- ❑ Other terms for similar concepts: Adaptive Streaming, Smooth Streaming, HTTP Chunking
- ❑ Actually a series of small progressive downloads of chunks (or range requests)
- ❑ No universally used protocol ...

HTTP-based Adaptive Streaming (HAS)

- ❑ Other terms for similar concepts: Adaptive Streaming, Smooth Streaming, HTTP Chunking
- ❑ Actually a series of small progressive downloads of chunks (or range requests)
- ❑ No universally used protocol ...
 - Apple HLS: HTTP Live Streaming
 - Microsoft IIS Smooth Streaming: part of Silverlight
 - Adobe: Flash Dynamic Streaming
 - DASH: Dynamic Adaptive Streaming over HTTP

Example players



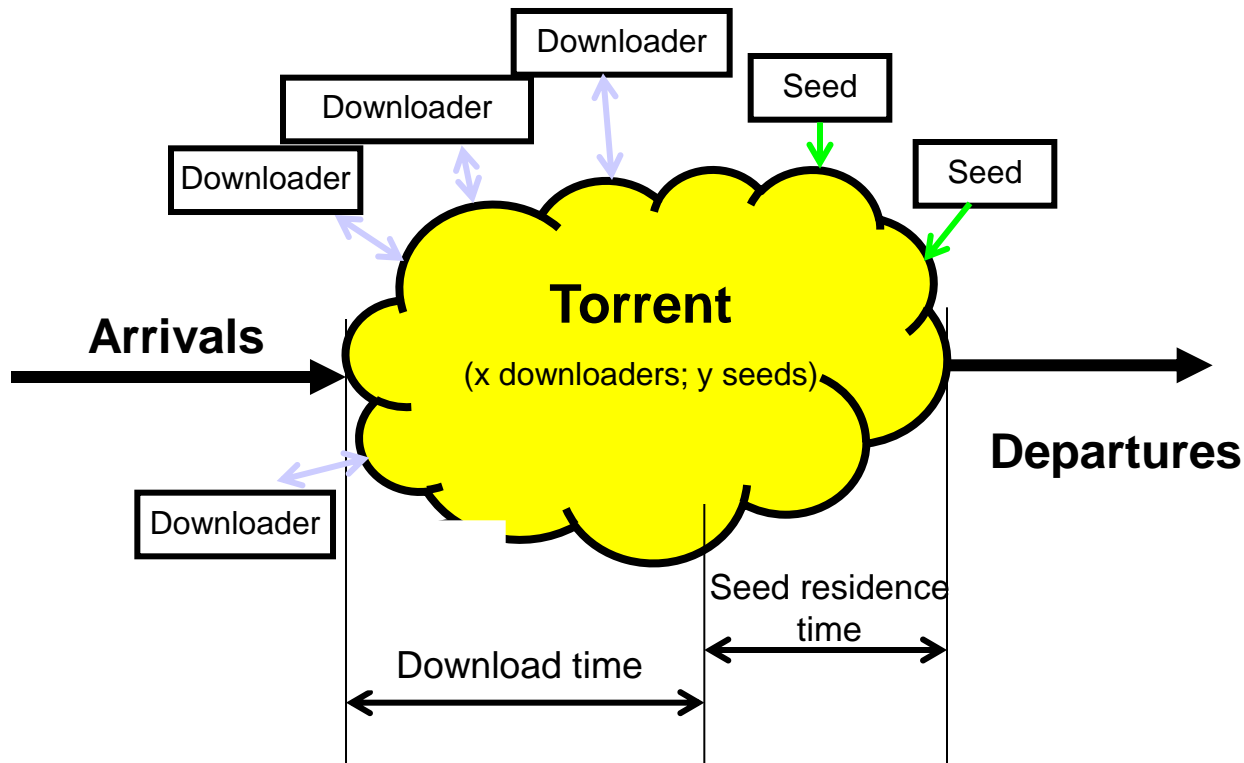
	Player	Container	Type	Open Source
 Microsoft Silverlight™	Microsoft Smooth Streaming	Silverlight	Chunk	✗
	Netflix player	Silverlight	Range	✗
	Apple HLS	QuickTime	Chunk	✗
	Adobe HDS	Flash	Chunk	✓

... and now YouTube (Google) is also using HAS!



BitTorrent-like systems

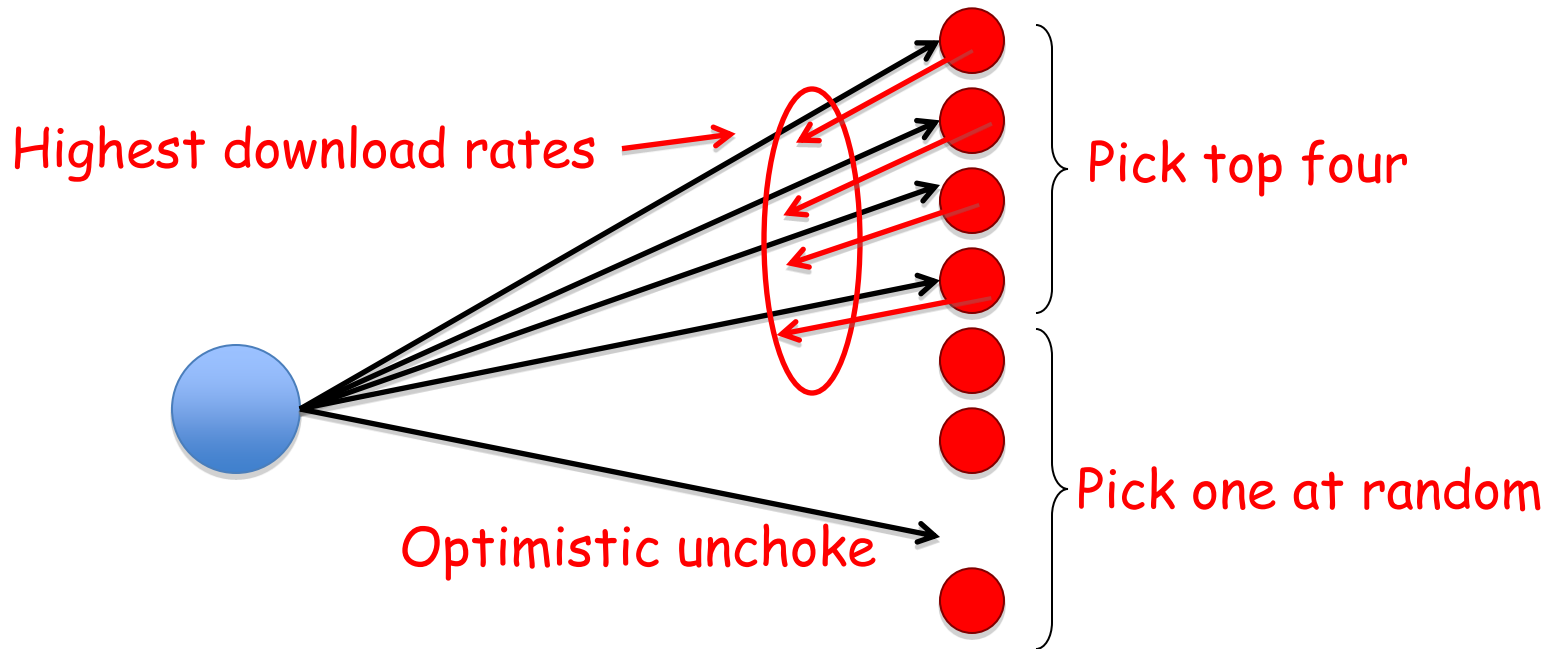
- ❑ File split into many smaller pieces
- ❑ Pieces are downloaded from both seeds and downloaders
- ❑ Distribution paths are dynamically determined
 - Based on data availability



Download using BitTorrent

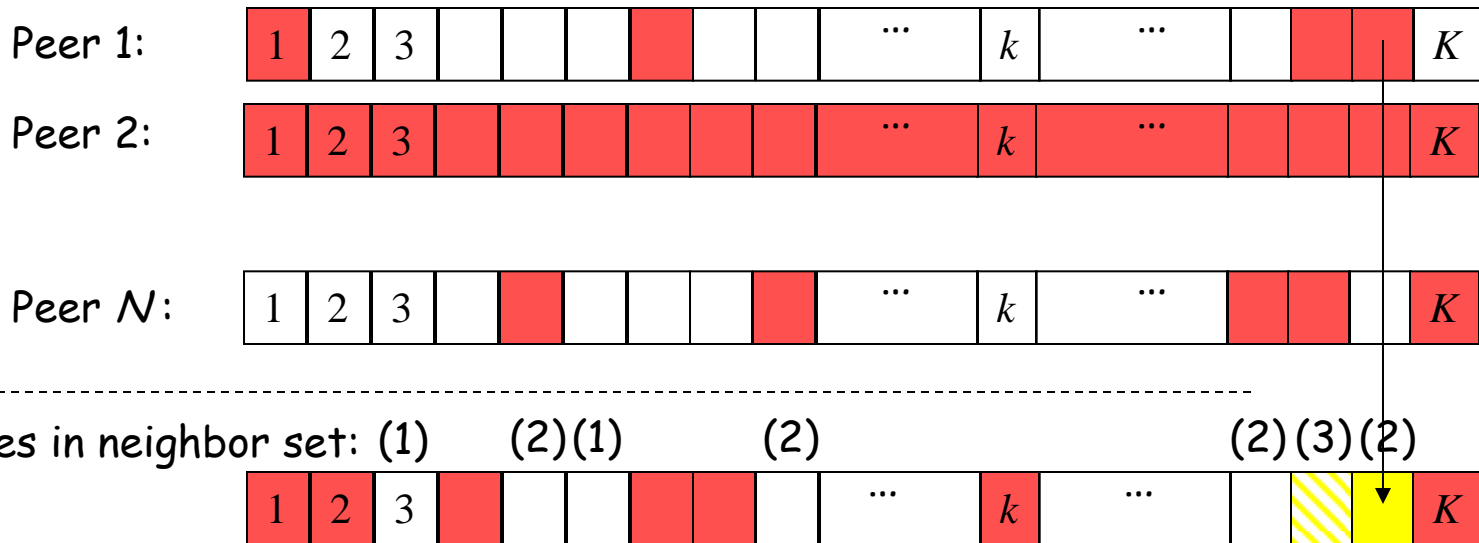
Background: Incentive mechanism

- ❑ Establish connections to large set of peers
 - At each time, only upload to a small (changing) set of peers
- ❑ Rate-based tit-for-tat policy
 - Downloaders give upload preference to the downloaders that provide the highest download rates



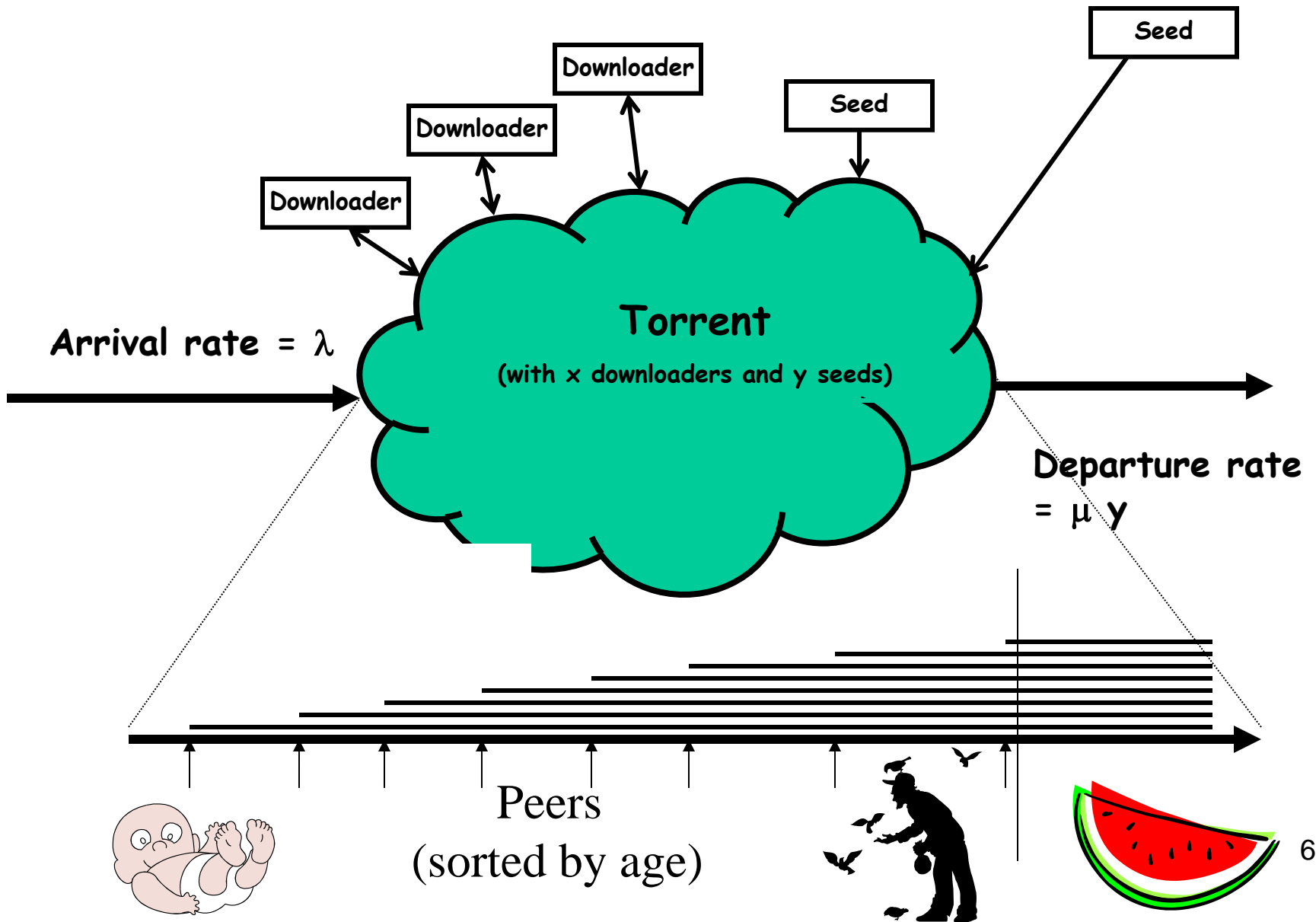
Download using BitTorrent

Background: Piece selection

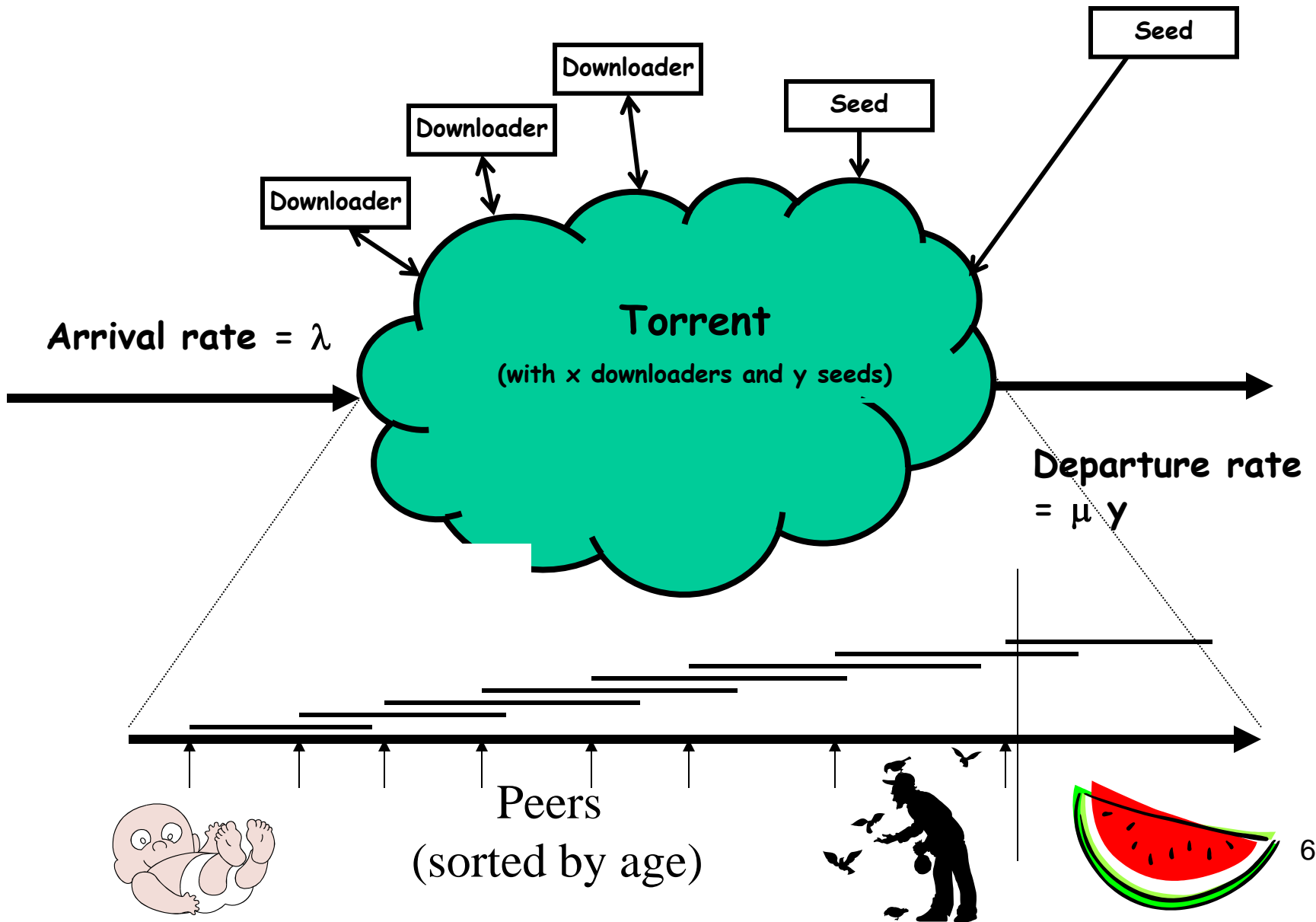


- ❑ Rarest first piece selection policy
 - Achieves high piece diversity
- ❑ Request pieces that
 - the uploader has;
 - the downloader is interested (wants); and
 - is the rarest among this set of pieces

BitTorrent Model (random)



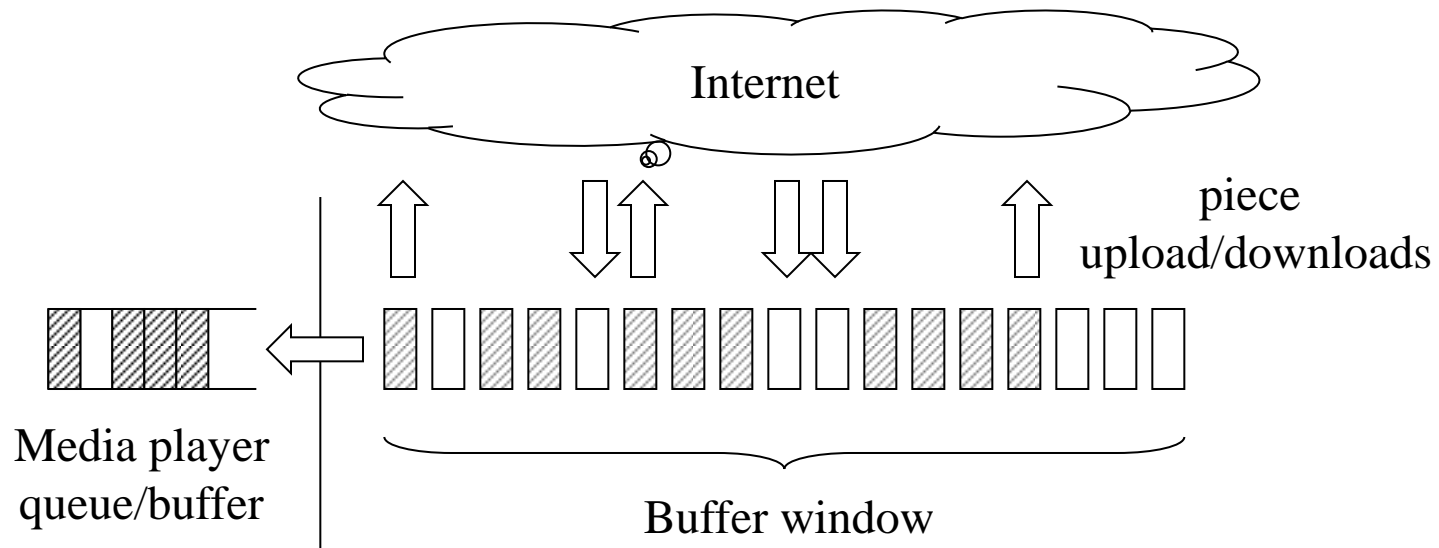
BitTorrent Model (chaining)



Peer-assisted VoD streaming

- ❑ Can BitTorrent-like protocols provide scalable on-demand streaming?
- ❑ How sensitive is the performance to the application configuration parameters?
 - Piece selection policy (rarest vs. in-order tradeoff)
 - Peer selection policy
 - Upload/download bandwidth
- ❑ What is the user-perceived performance?
 - Start-up delay
 - Probability of disrupted playback

Live Streaming using BT-like systems



□ Live streaming (e.g., CoolStreaming)

- All peers at roughly the same play/download position
 - High bandwidth peers can easily contribute more ...
- (relatively) Small buffer window
 - Within which pieces are exchanged

More slides

Example: HAS and proxy



Clients' want

- ❑ High playback quality
- ❑ Small stall times
- ❑ Few buffer interruptions
- ❑ Few quality switches

Example: HAS and proxy



Clients' want

- ❑ High playback quality
- ❑ Small stall times
- ❑ Few buffer interruptions
- ❑ Few quality switches

HAS is increasingly responsible
for larger traffic volumes
... proxies to reduce traffic??

Example: HAS and proxy



Clients' want

- High playback quality
- Small stall times
- Few buffer interruptions
- Few quality switches

Network providers' want

- High QoE of customers/clients

Example: HAS and proxy



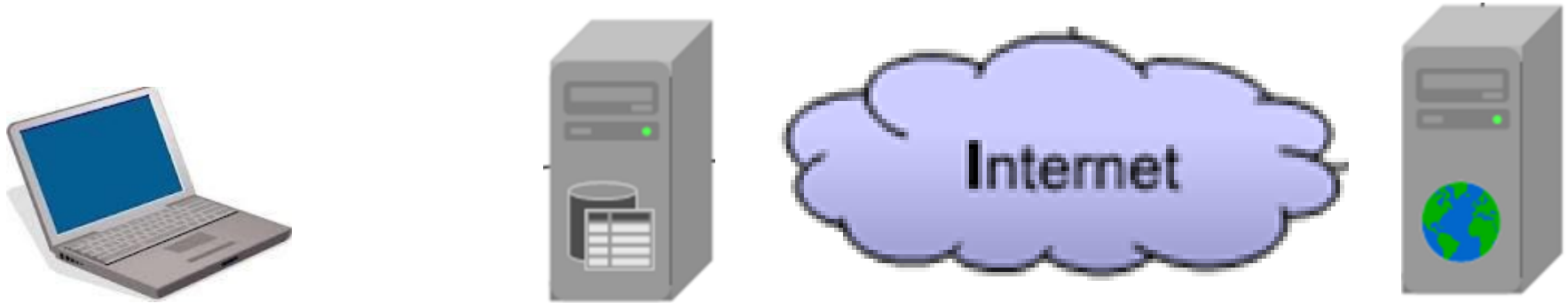
Clients' want

- ❑ High playback quality
- ❑ Small stall times
- ❑ Few buffer interruptions
- ❑ Few quality switches

Network providers' want

- High QoE of customers/clients
- Low bandwidth usage
- High hit rate

Example: HAS and proxy



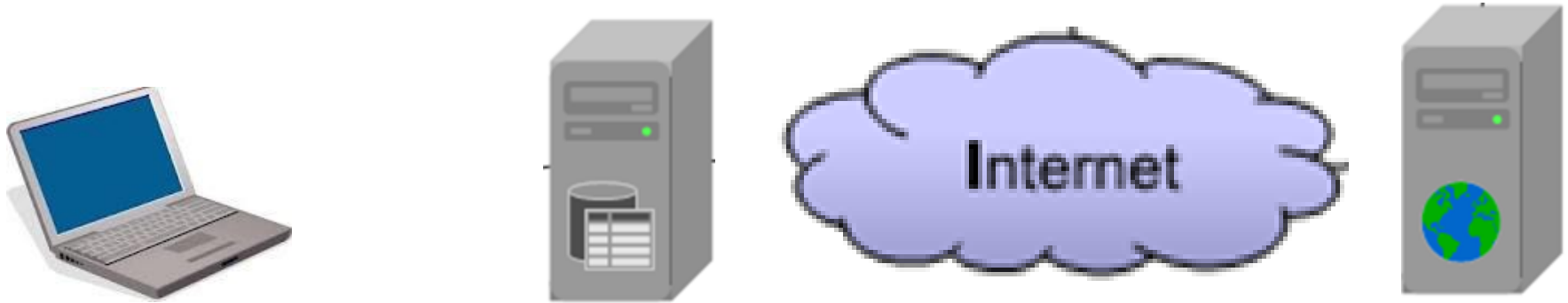
Clients' want

- ❑ High playback quality
- ❑ Small stall times
- ❑ Few buffer interruptions
- ❑ Few quality switches

Network providers' want

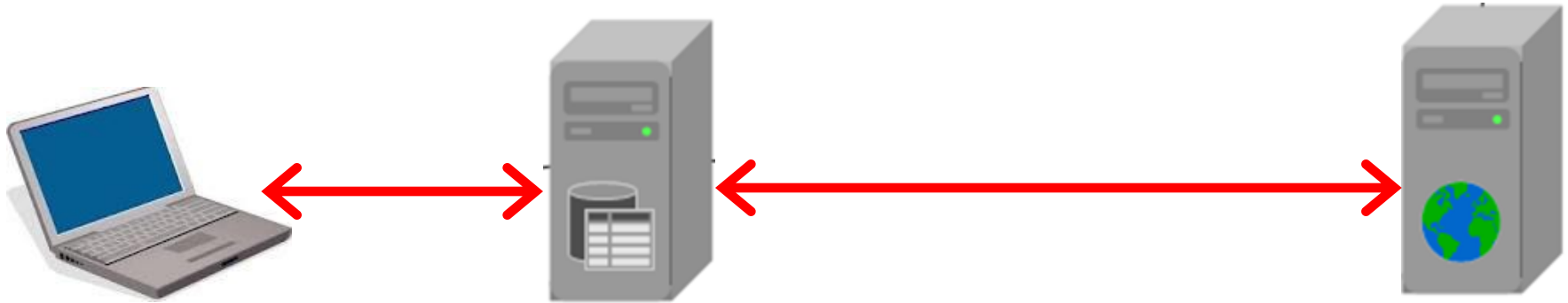
- High QoE of customers/clients
- Low bandwidth usage
- High hit rate

Example: HAS and proxy

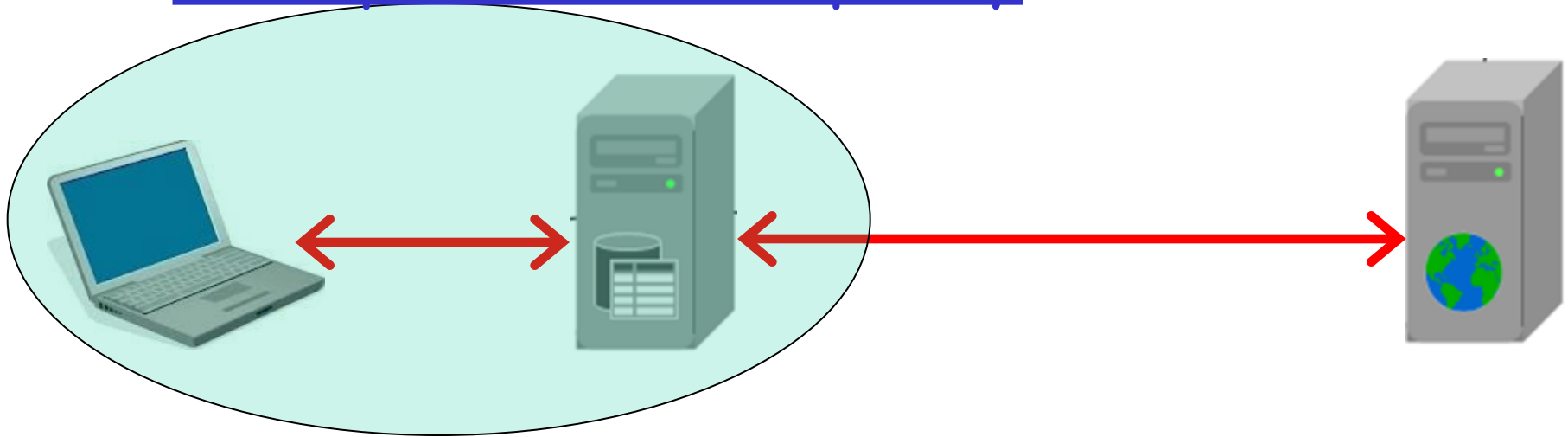


Proxy example ...

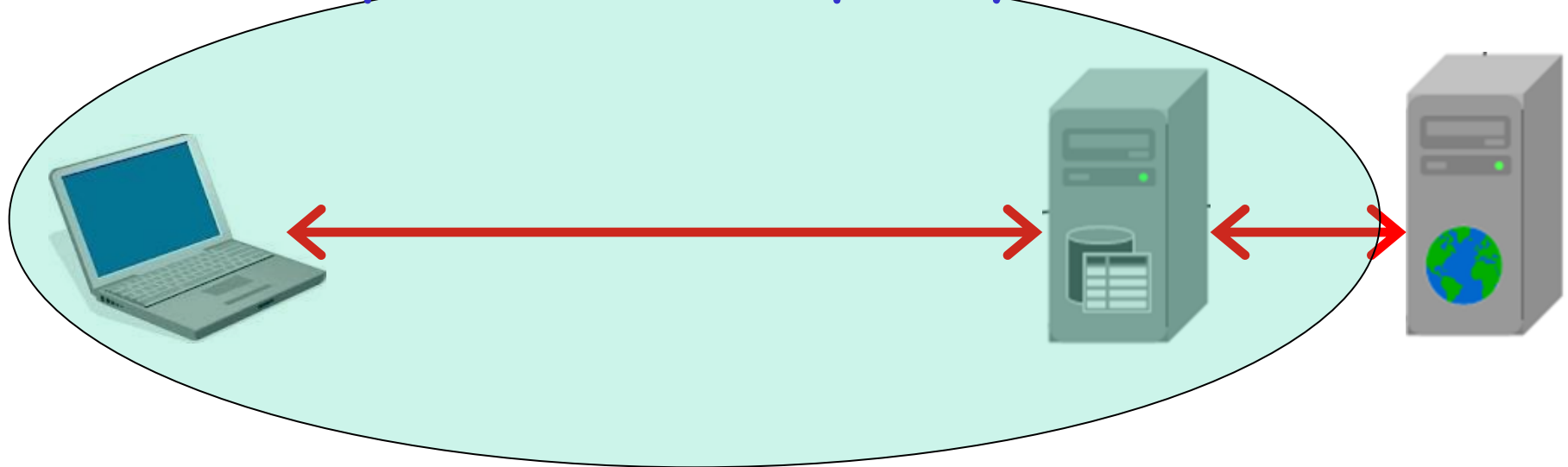
Example: HAS and proxy



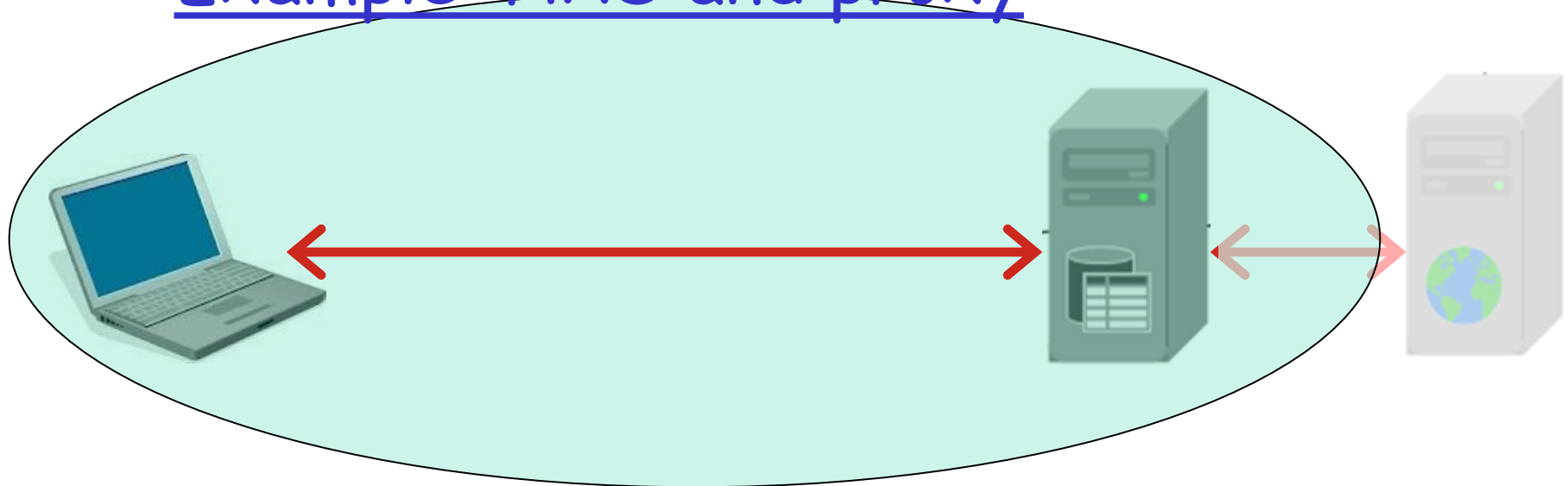
Example: HAS and proxy



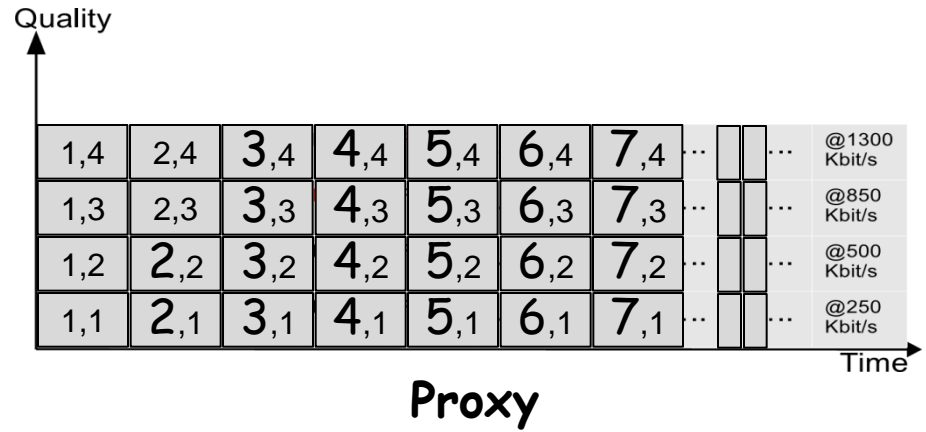
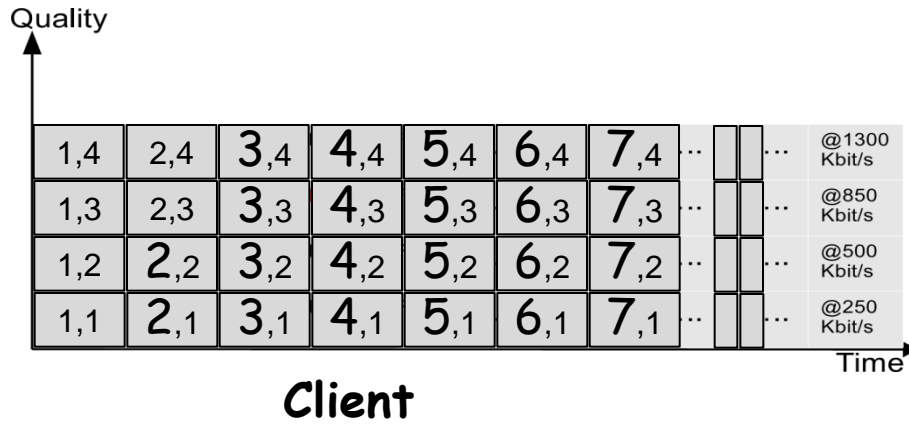
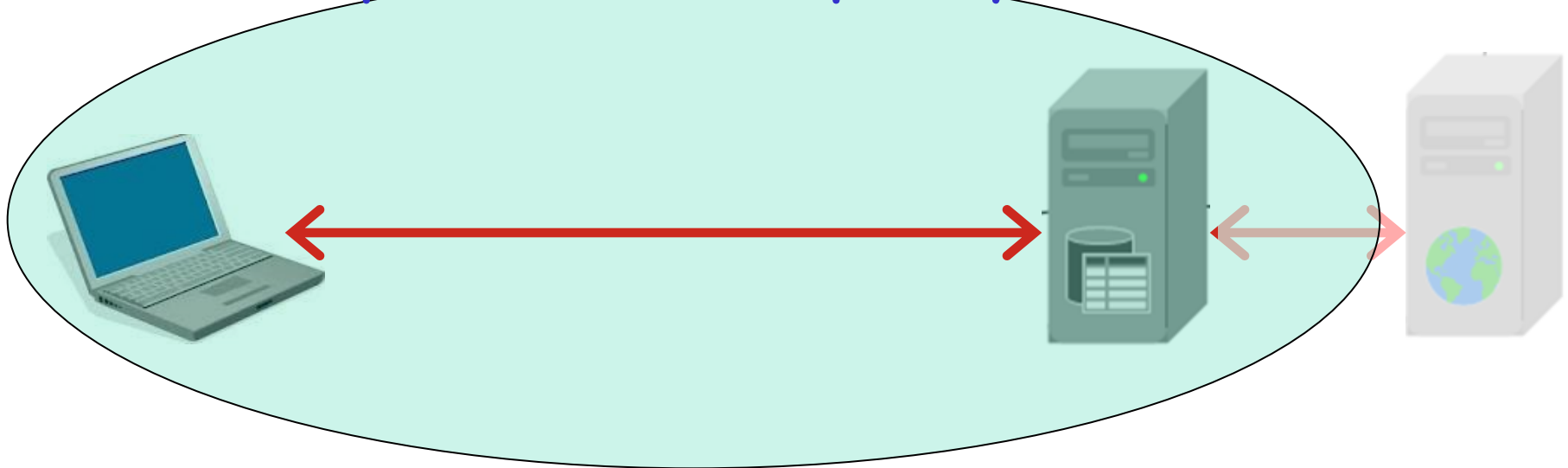
Example: HAS and proxy



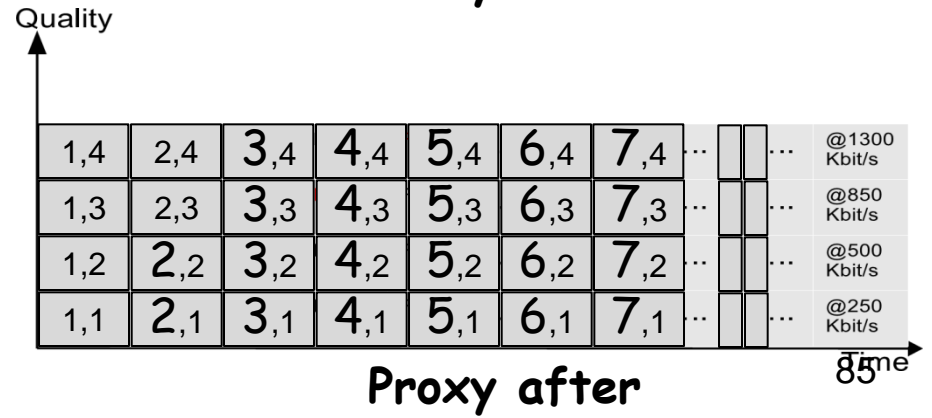
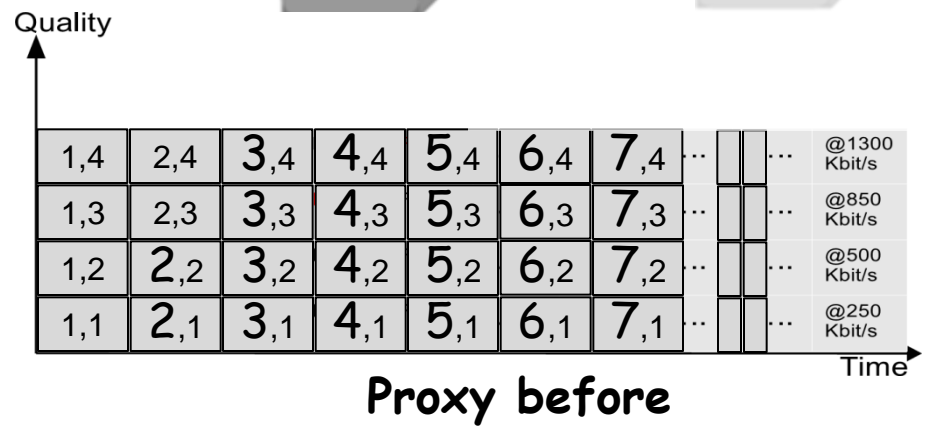
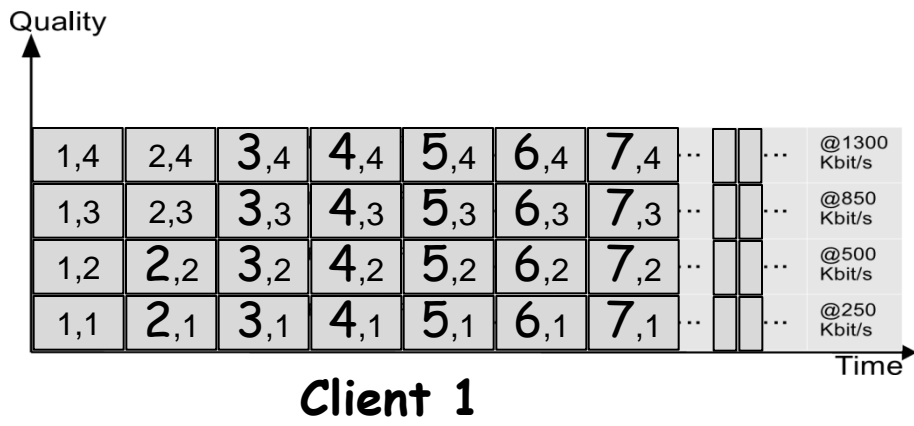
Example: HAS and proxy



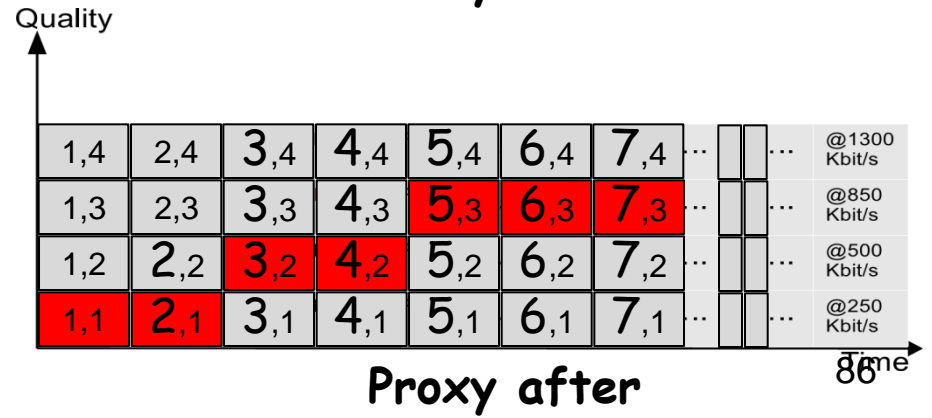
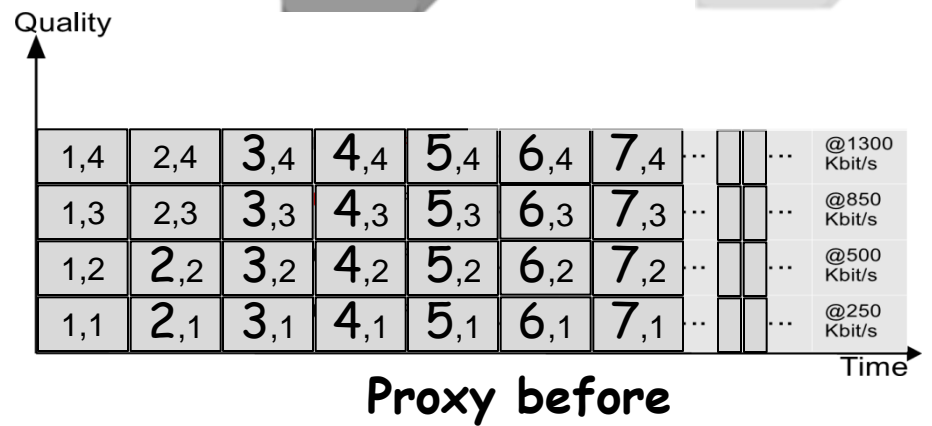
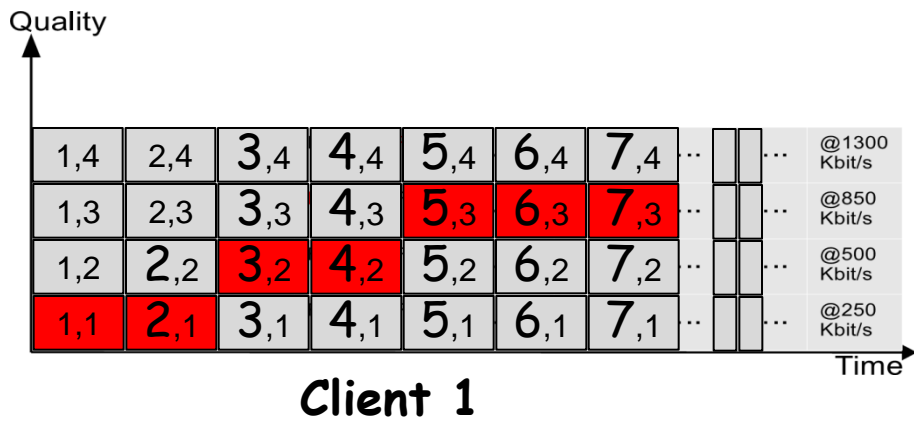
Example: HAS and proxy



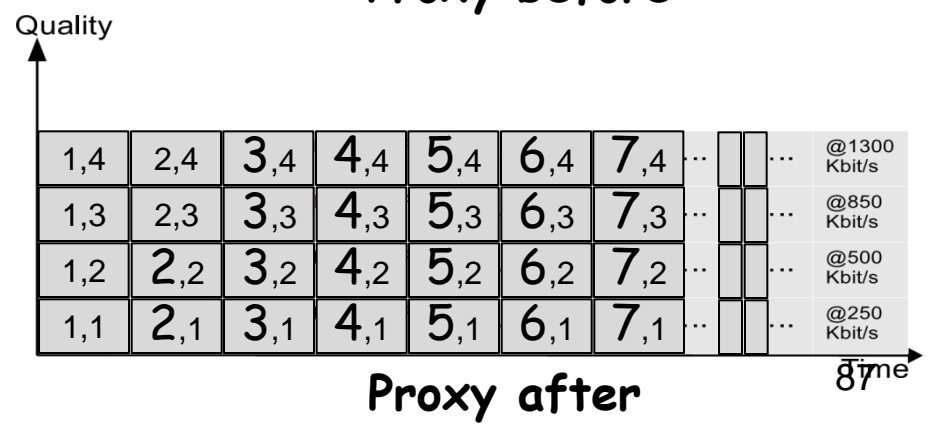
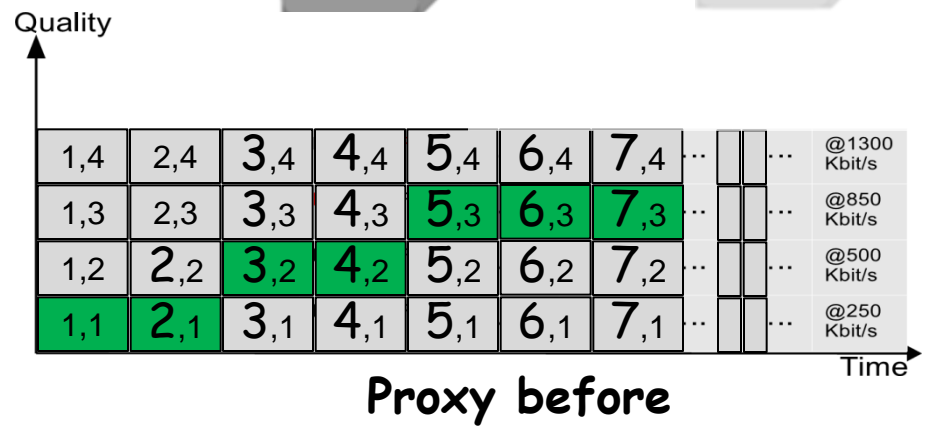
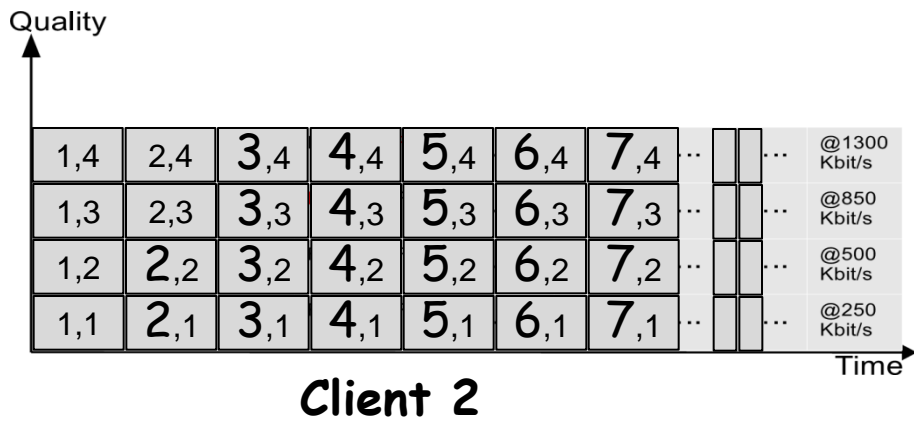
Example: HAS and proxy



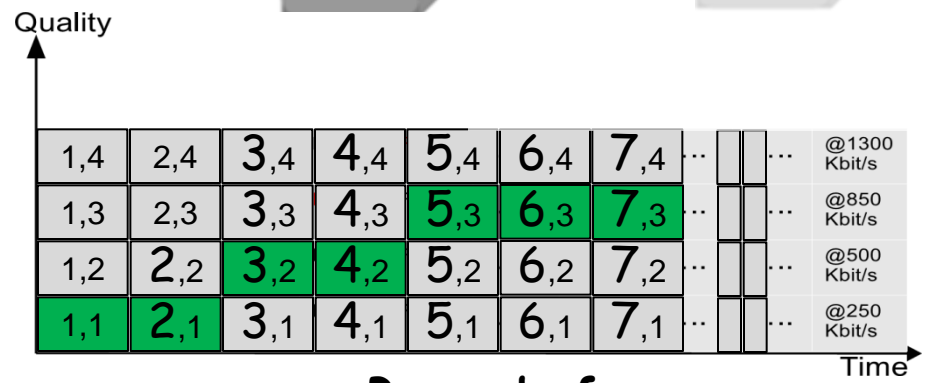
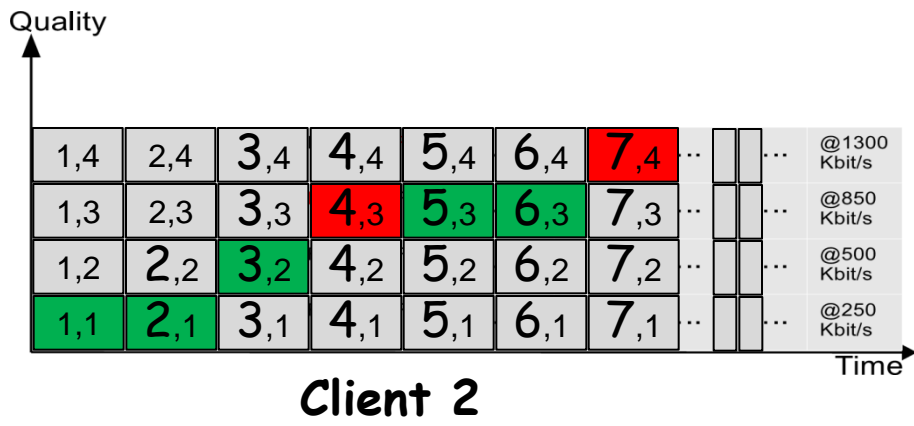
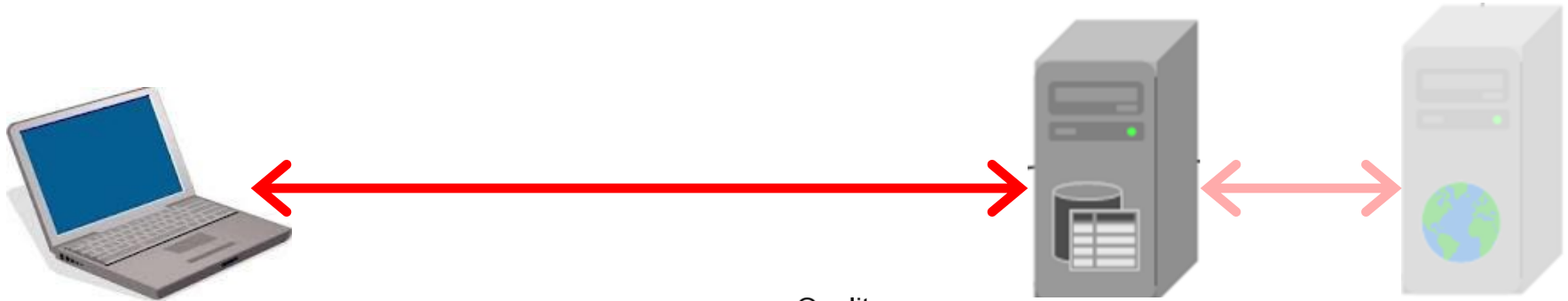
Example: HAS and proxy



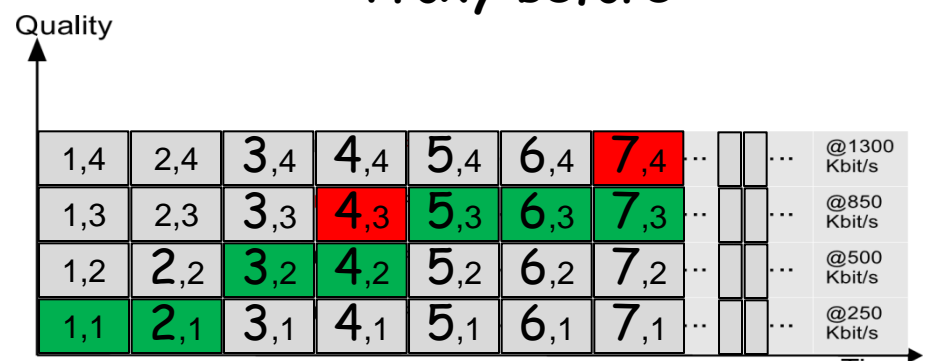
Example: HAS and proxy



Example: HAS and proxy

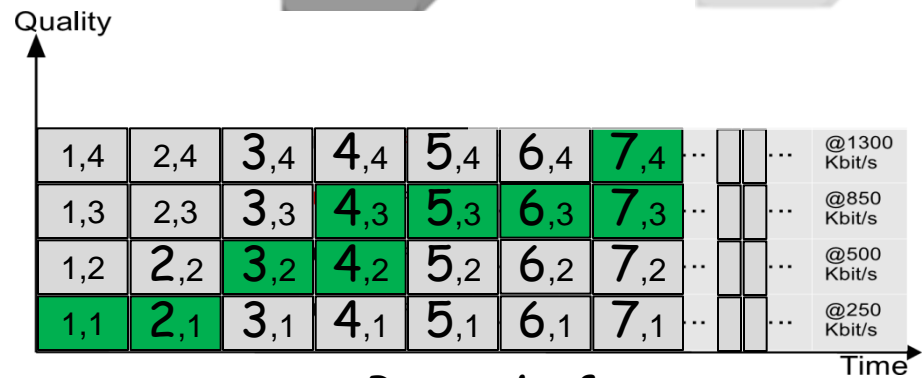
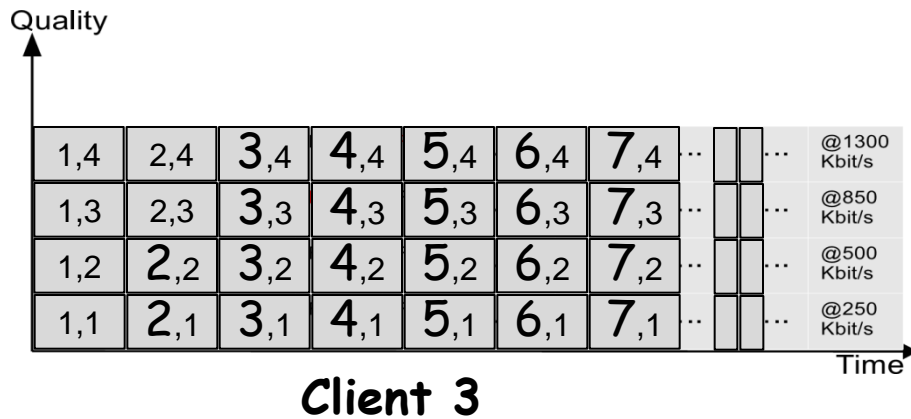


Proxy before

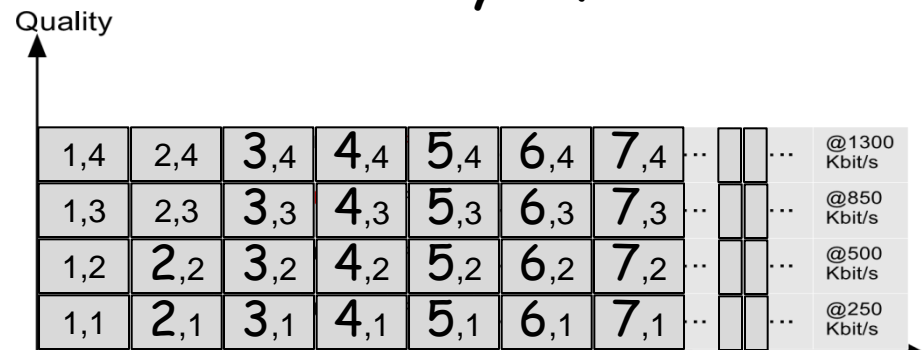


Proxy after

Example: HAS and proxy

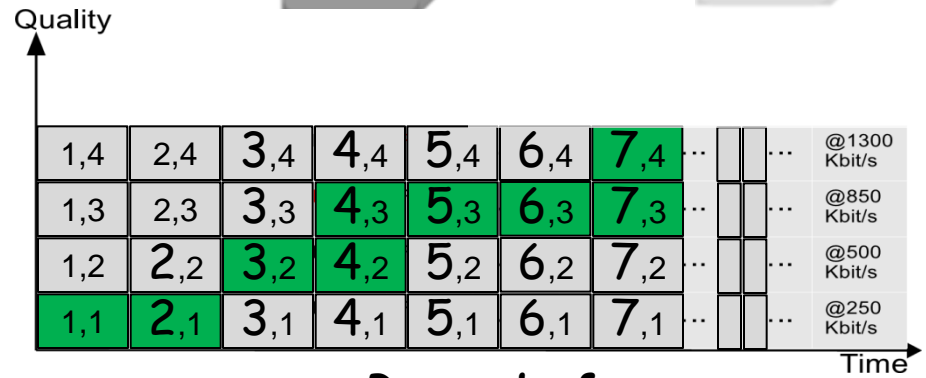
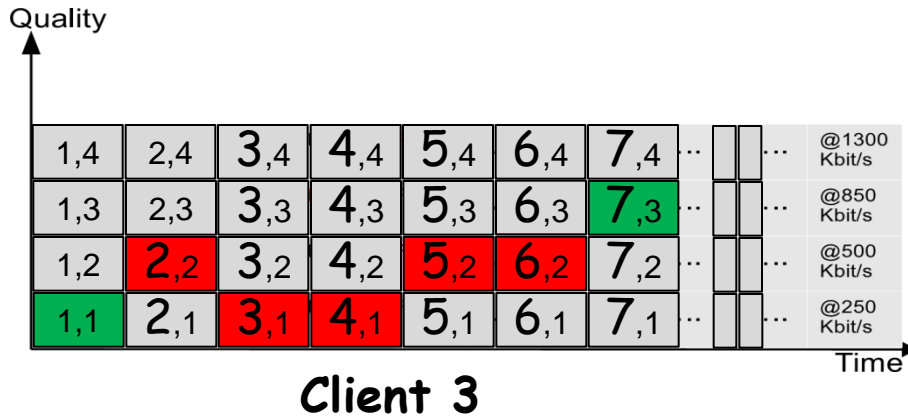


Proxy before

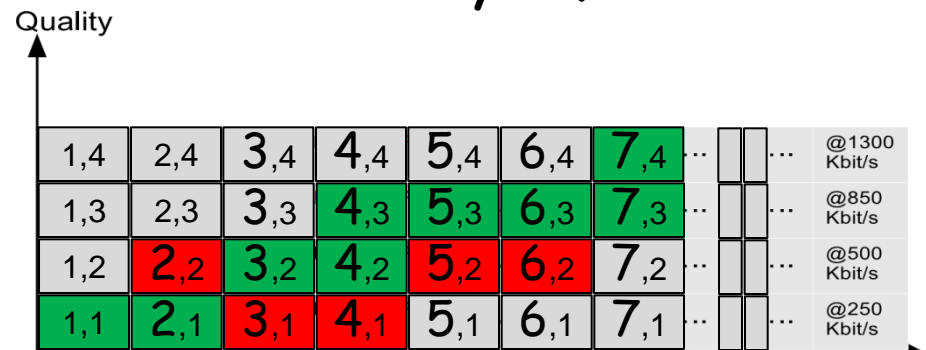


Proxy after

Example: HAS and proxy



Proxy before



Proxy after