7 Streaming Architectures

7.1 Streaming: Basic Terminology
7.2 High-Level Streaming Architecture
7.3 Real-Time Data Transport *
7.4 Scalability and Multicast *

Literature:

David Austerberry: The Technology of Video & Audio Streaming, Focal Press 2002

Gregory C. Demetriades: Streaming Media, Wiley 2003

Tobias Künkel: Streaming Media – Technologien, Standards, Anwendungen, Addison-Wesley 2001

* Hinweis: Teilweise Überlappung mit „Rechnernetze II“ (Hegering)

Outline

1. Introduction and Motivation

Part I: Content-Oriented Base Technologies

2. Digital Rights Management
3. Cryptographic Techniques
4. Electronic Payment Systems
5. Multimedia Content Description

Part II: Multimedia Distribution Services

6. Multimedia Content Production and Management
7. Streaming Architectures
8. Commercial Streaming Systems: An Overview
9. Communities, the Web and Multimedia
10. Web Radio and Web TV

Part III: Conversational Multimedia Services

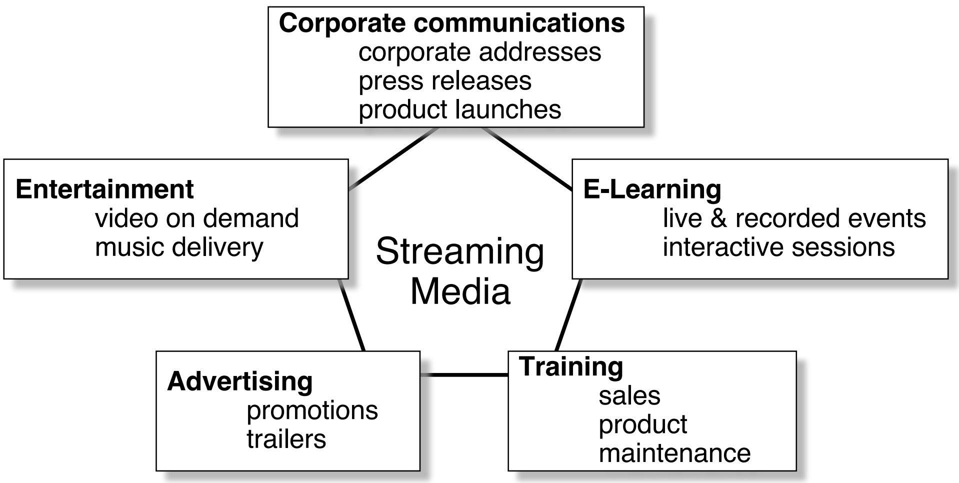
11. Signaling Protocols for Multimedia Communication
12. Multimedia Conferencing

Streaming, Streaming Media

- *Streaming media* is the term used to describe the real-time delivery of moving images, moving text and sound, over the internet.
- Delivery types for audio and video content:
  – *Download and Play:* Content must be downloaded completely to the client before it can be played
  – *Progressive Download:* Playback is started while download is still in progress. Download rate independent of program bit rate.
  – *True Streaming:* Delivered media is viewed/listened in "real-time".
    » Playback takes place with roughly the same rate as delivery of data
    » Delay between send and receive event of data packet kept small
- Subtypes of True Streaming:
  – *Static File Streaming:* Delivery of pre-recorded media files. Often also called *on-demand* delivery (e.g. *Video on Demand)*
  – *Live Streaming:* Source of delivered media is not pre-recorded but fed into the server in real-time. Examples: *Webcast* of live events, Web radio
  – Static file streaming enables a higher degree of interactivity

Based on material from www.streamingalliance.org

Applications of Streaming Media



Session

- A *session* is an association between communicating parties, which
  - Persists over a limited time span
  - Incorporates at least two parties
  - May comprise a large number of communication connections of different characteristics
- Examples of sessions:
  - Movie streamed to consumer, consisting of audio and video parts
  - Multimedia conference among five participants, consisting of audio and video source from each of of the participants (plus possibly some global information)
- Session awareness at which levels?
  - At application level: unavoidable
  - At network level: possible
    - » Requires specific protocols

Network Feature vs. End System Application

- Traditional telecommunication networks:
  - End systems are "dumb" (classical phone sets)
  - All advanced features (e.g. multimedia sessions) are realized in the network
    - » Upgrade of network equipment
    - » Upgrade of signalling protocols
- Internet:
  - End systems are universal computers
  - Main network function (IP routing) is extremely simple
  - Advanced features are mainly realized by software on end systems
  - Additional protocols
    - » "end-to-end" for communication among end system software
    - » triggering specific behaviour of the network (e.g. resource reservation in routers) —> difficult to deploy, therefore rare to find

7 Streaming Architectures

7.1 Streaming: Basic Terminology

7.2 High-Level Streaming Architecture

7.3 Real-Time Data Transport

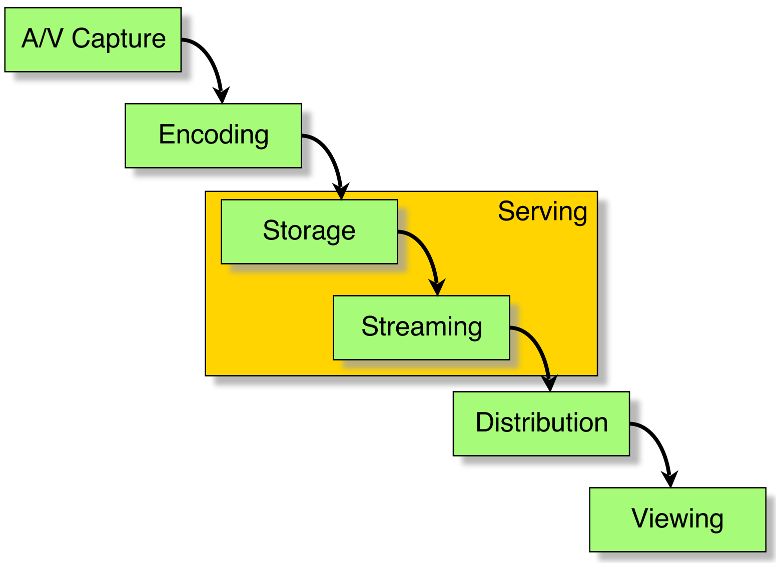
7.4 Scalability and Multicast

Literature:

David Austerberry: The Technology of Video & Audio Streaming, Focal Press 2002

Gregory C. Demetriades: Streaming Media, Wiley 2003

Streaming Delivery Chain



Encoding

- Format conversions
  - E.g. analog/digital conversion
  - E.g. downscaling of picture size
- Compression
  - Adequate for player capabilities and typical transmission bandwidth
- Indexing
  - Analysing internal structure
- Metadata creation
  - Possibly including digital rights specification

Serving

- Storage
  - Live: only buffering and archiving
  - Static files: archive management, retrieval
- Streaming
  - Request-response driven similar to Web server
- Interactivity
  - In static files:
    - » VCR-like control (PLAY, STOP, PAUSE, FFWD, REW)
    - » Random access based on various criteria
  - In structured (mostly static) material archives:
    - » Hyperlinks in A/V material (“hypervideo”)
    - » Web-like technology: Video-Web, links to arbitrary streaming sources

Distribution

- Key topic: Quality of Service (QoS)
  - Determining realizable bandwidth, delay, jitter
  - See “Rechnernetze II”
- Key concepts:
  - Overprovisioning
  - Detailed reservations (“Integrated Services”, reservation protocol RSVP)
    - » Difficult to scale to large numbers of users
  - Traffic classes (“Differentiated Services”)
    - » Difficult to control access to privileges
  - Resource management layer
  - Technology-specific solutions
    - » E.g. ATM (Asynchronous Transfer Mode)

7 Streaming Architectures

7.1 Streaming: Basic Terminology
7.2 High-Level Streaming Architecture
7.3 Real-Time Data Transport
7.4 Scalability and Multicast

Literature:

David Austerberry: The Technology of Video & Audio Streaming, Focal Press 2002

Stephan Rupp, Gerd Siegmund, Wolfgang Lautenschlager: SIP – Multimediale Dienste im Internet. dpunkt 2002

IP and TCP

- Internet Protocol
  - Network communication protocol (ISO layer 3)
  - Packets transferred from address to address (through routers)
  - Main problems:
    - » Variable network latency
    - » Packet order on arrival may be different than on sending
    - » Packets may be lost
- Transport Control Protocol (TCP)
  - Connection establishment (by “three-way handshake”)
    - » Connections are sequences of associated IP packets
  - Sequencing of bytes with forwarding acknowledgement number
  - Non-acknowledged bytes are re-transmitted after a defined time period
  - Flow control
- For audio/video streaming:
  - Retransmissions (and associated delays) are harmful
  - Lost packets can be tolerated to some extent
  - TCP not adequate for data transport

UDP

- User Datagram Protocol (UDP)
- Extremely simple transport protocol over IP
  - Connectionless (TCP: connection-oriented)
  - Unreliable (TCP: reliable)
  - No flow control (TCP: has flow control)
- Contents of a UDP datagram:
  - Ports used by application program
  - Checksum
- Basically adequate for media data transport
  - Very efficient, protocol overhead of TCP avoided
  - Flow control and handling of packet loss have to be handled by higher protocol layer

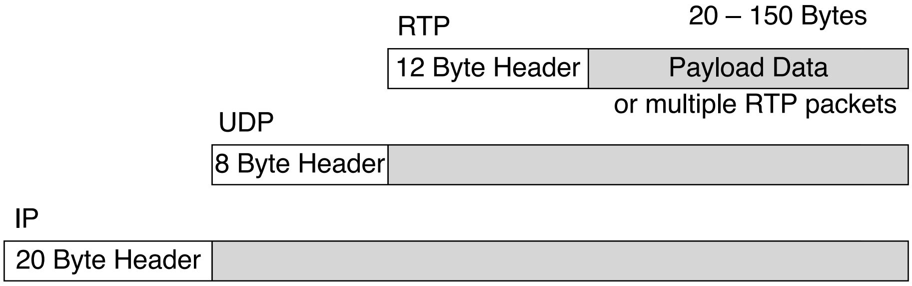
Streaming with HTTP?

- Theoretically possible
  - Client requests next slice of audio/video information from server
  - Server responds with audio/video data
  - Popular in simple streaming applications
    - » e.g. MP3-Streaming with ShoutCast, Live365
- Main problem
  - HTTP usually run over TCP
  - Large overhead for ensuring correct transmission
- Consequence:
  - Specialized streaming protocols

Real-Time Transport Protocol RTP

- Transport protocol specifically developed for streaming data
  - IETF (Internet Engineering Task Force) RFC (Request for Comments) 1889
- RTP packets contain
  - Sequence number
  - Time stamp
  - Identification of sender and destination
- RTP usually carried over UDP
- Very important:
  - **RTP does not at all change the way how IP packets are transferred in the network!**
  - To achieve “Quality of Service”, additional network technologies are required (see above)
- RTP used by:
  - Apple QuickTime architecture
  - RealSystems streaming architecture

RTP Packets and Other Protocols



- IP Header:
  – Source address, destination address, length, time to live, ...
- UDP Header:
  – Port numbers (source and target processes), length, checksum

RTP Header Format

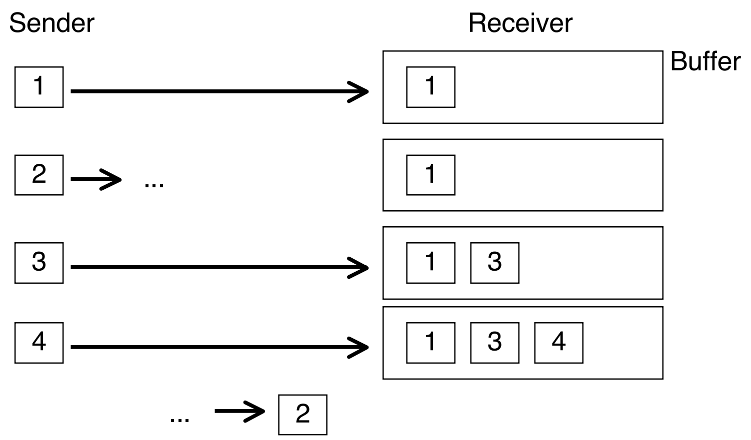
Payload Type (7 Bit)	Sequence Number (16 Bit)	…
Timestamp (32 Bit)		
Synchronization Source (SSRC) Identifier (32 Bit)		
Contributing Source (CSRC) Identifier (32 Bit) (repeated)		

- Payload Type: Source coding (codec used)
- SSRC: Identification of sending data source, defined basis for sequence numbers and time stamps
- CSRC (optional): Identifications for original data sources which have been *mixed* together to form the synchronization source

Buffer Size Allocation

- Streaming quality relies on
  - Features of the network, and
  - *Adaptive* codecs on client/server side
  - E.g. intelligent use of buffering
- *Constant* delay does not require buffering
- Buffers are necessary to deal with *jitter*
  - Varying network delays
- Estimation of buffer sizes
  - Based on preliminary test transmissions
  - Adaptively during content transmission
- Problem:
  - Keep buffers small to achieve proper “real-time”
  - Avoid “buffer underrun” in case of exceptionally long delays

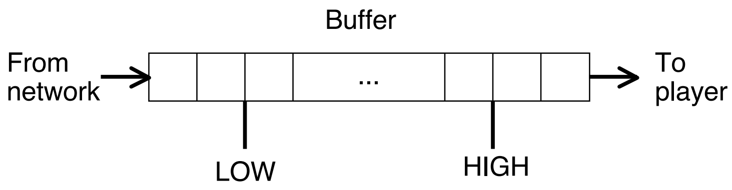
Jitter and Loss Compensation



- Options for application on receiver side:
  - Wait *(not* adequate), repeat last packet (1), interpolate (between 1 and 3)
  - Missing audio information is difficult, missing video can be compensated

Adaptive Transmission Rate Control

- Application-level mechanism
- Define “low” and “high” thresholds on buffer
- Communication between client and server
  - Lower effective transmission rate when high threshold is reached
  - Increase effective transmission rate when low threshold is reached
- Changing effective transmission rate
  - In case of guaranteed network QoS: identical to sending rate
  - In case of non-guaranteed network QoS: may change content quality (frame rate, resolution etc.)



Real-Time Control Protocol RTCP

- RTCP controls the transmission (not the setup of connection)
- RTCP periodically sends monitoring information to all participants in a streaming session
- Main functions of RTCP:
  - Feedback on QoS of transmission
    - » Information for adaptive codecs, e.g. whether problem is local or global
  - Identification of sender by “canonical name”
    - » Helpful when synchronization source changes
    - » Supports lip synchronization between audio and video
  - Number of participants in a session
    - » Adaptation of sending rate of RTCP control information to number of participants, to avoid network overload
  - Transmission of additional information, e.g. names of session participants

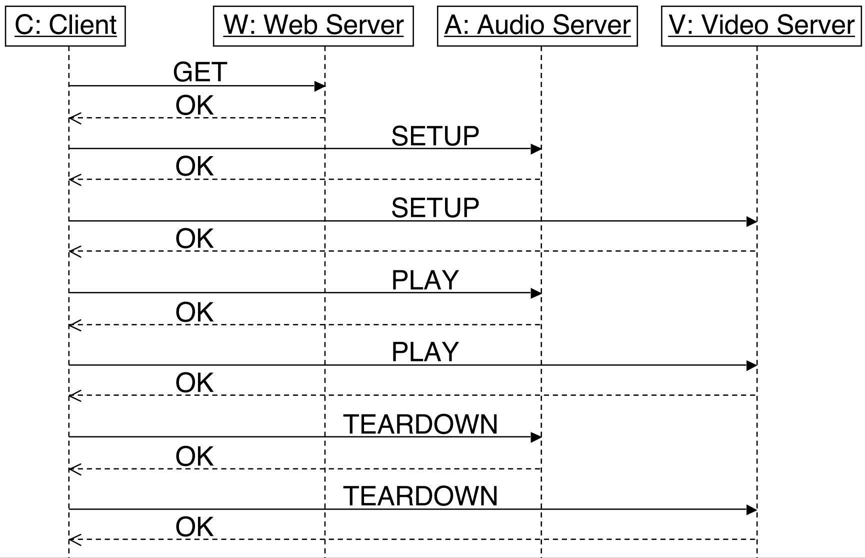
Real Time Streaming Protocol RTSP

- Client-server multimedia presentation protocol, designed specifically for streamed media
  - IETF (Internet Engineering Task Force) RFC (Request for Comments) 2326 (“MMUSIC” work group)
    - » February 1998, draft revision February 2004
  - “The Internet VCR remote control protocol” (www.rtsp.org)
  - Independent of the use of RTP for transport
  - Syntactically similar to HTTP 1.1 (carried over TCP or UDP, TLS in prepn.)
- Main operations supported by RTSP:
  - Retrieval of media from media server
  - Invitation of a media server to a conference
- Key terminology
  - Aggregate control (e.g. for audio & video)
  - Server control (clients should be able to stop streaming from a server)
  - Transport & capability negotiation (e.g. disallowing a “seek” function)

Main Methods of RTSP

- SETUP:
  - Causes the server to allocate resources for a stream and create a RTSP session.
- PLAY:
  - Starts data transmission on a stream allocated via SETUP
  - Fast forward (scale ratio parameter)
- PAUSE:
  - Temporarily halts a stream without freeing server resources.
- REDIRECT:
  - Indicates that the session should be moved to new server / location
- PING:
  - Prevents the identified session from being timed out.
- TEARDOWN:
  - Frees resources associated with the stream. The RTSP session ceases to exist on the server.

Example Session with RTSP



Microsoft Media Server MMS

- Microsoft Technology for transmission of data packets and control messages from server to client
  - Based on TCP
- Protocol Rollover: Server tries the following protocols in sequence
  - MMSU: Microsoft Media Server Protocol/UDP
  - MMST: Microsoft Media Server Protocol/TCP
  - HTTP (may be successful in Firewall configurations)
- MSBD (Media Stream Broadcast Distribution Protocol):
  - Earlier solution for client-server connections
  - Currently limited in client numbers

7 Streaming Architectures

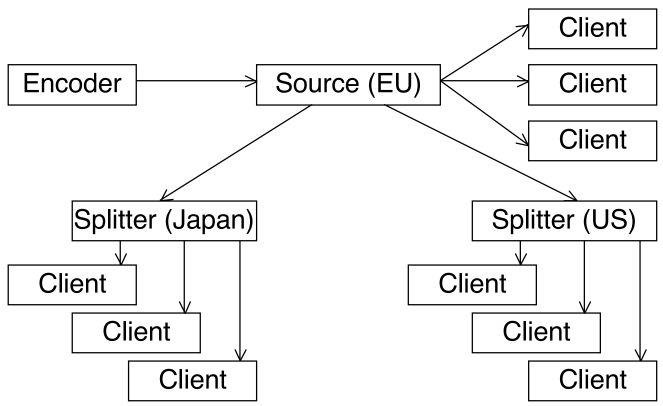
7.1 Streaming: Basic Terminology
7.2 High-Level Streaming Architecture
7.3 Real-Time Data Transport
7.4 Scalability and Multicast

Literature:

David Austerberry: The Technology of Video & Audio Streaming, Focal Press 2002
Gregory C. Demetriades: Streaming Media, Wiley 2003
Xueyan Tang et al.: Web Content Delivery, Springer 2005

Splitting

- Video servers are limited in capacity
- Assuming clients at spatially distant locations
  - Intermediate, forwarding server is useful: “splitter”



Content Delivery Networks (CDN)

- Serve content closer to to the user
  - "edge serving"
- Main components of CDN:
  - Smart routing
  - Edge caching of content
  - Proxy serving
  - Splitting of live webcasts

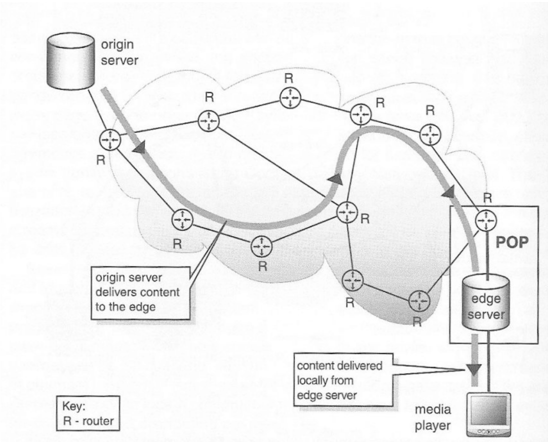


Figure 15.3 Serving from the edge with a CDN.
Figure from Austerberry 2002

Content Delivery Networks

- "Overlay server infrastructure"
  - Network of centrally operated Web and streaming servers
  - Geographically distributed, present in main ISP networks
  - Flexibly used for content from various sources
- Content delivery as a service
  - Content delivery service provider owns server infrastructure
  - Content owner pays for having content delivered to customers
- Examples:
  - Akamai (delivers 10-20% of all Web traffic)
    - » Runs 20,000 servers in 1000 networks spanning 70+ countries
  - VitalStream
    - » Specialized in media streaming

Example of Administration Interface

- VitalStream MediaConsole

MediaConsole Media Package Manager Screenshot

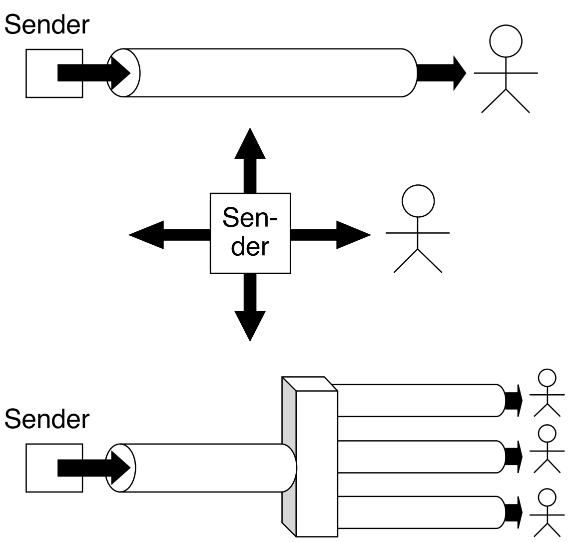
Key Problems in CDNs

- Replica placement:
  - Where to place copies of web sites or other content
  - Problem is in general NP-hard (Karlsson, Karamolis, 2004)
  - Replica placement algorithms (RPA) achieve a suboptimal solution within reasonable time frame
  - Global information is difficult or costly to get - RPA uses local information mostly
- Request routing:
  - Mechanism and policy of redirecting client requests to a suitable server containing the requested content
  - Redirection algorithm: Decides what node to direct a client request to
  - Redirection mechanism: Way of redirecting the request (client, network)

Streaming Media in CDN

- General idea: Local proxy caching. But: ...
  - Huge size (1 KB vs. 100 MB)
    - » To cache only portions of the original?
  - Intensive bandwidth usage
    - » Minimizing bandwidth consumption as primary consideration
  - High interactivity
    - » E.g. premature termination is frequent (Chen et al. 04: approx. 90 %)
  - However: Media content is rather static (compared to Web pages)
- Caching algorithms
  - Different for homogeneous and heterogeneous clients (in bandwidth/format)
  - Sliding interval caching: sequential access, mainly effective for similar requests in short time period
  - Prefix caching: Saves time to load remaining parts
  - Segment caching: Generalization of prefix caching to support fast forward
  - Rate-split caching: Lower layer from original server, higher layer from proxy
  - Co-operative proxy caching (e.g. MiddleMan)

Unicast, Broadcast, Multicast



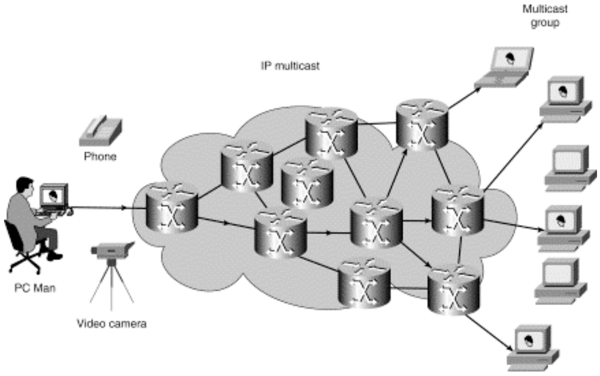
Unicast:
One specific receiver

Broadcast:
Receivable by everyone

Multicast:
Precisely defined group of receivers

IP Multicast

- Multicast relatively easy to integrate in routers
- IP address class D (224.0.0.0 through to 239.255.255.255) reserved for multicast (multicast groups)
- Registration/deregistration with IGMP (Internet Group Management Protocol)



- Reliable multicast: e.g. “Mbone” overlay network
- Multicast still rarely used in today’s Internet

IP Version 6

- Next generation of the IP protocol
- 128 Bit address space
  - Intended to relieve shortage of IP v4 addresses
- Built-in support for multicast
  - Specific multicast addresses
- Uptake of IP version 6 is (strangely) slow