

# Systeme II

## 2. Multimedia

Thomas Janson<sup>°</sup>, Kristof Van Laerhoven\*, Christian Ortolf<sup>°</sup>

Folien: Christian Schindelbauer<sup>°</sup>

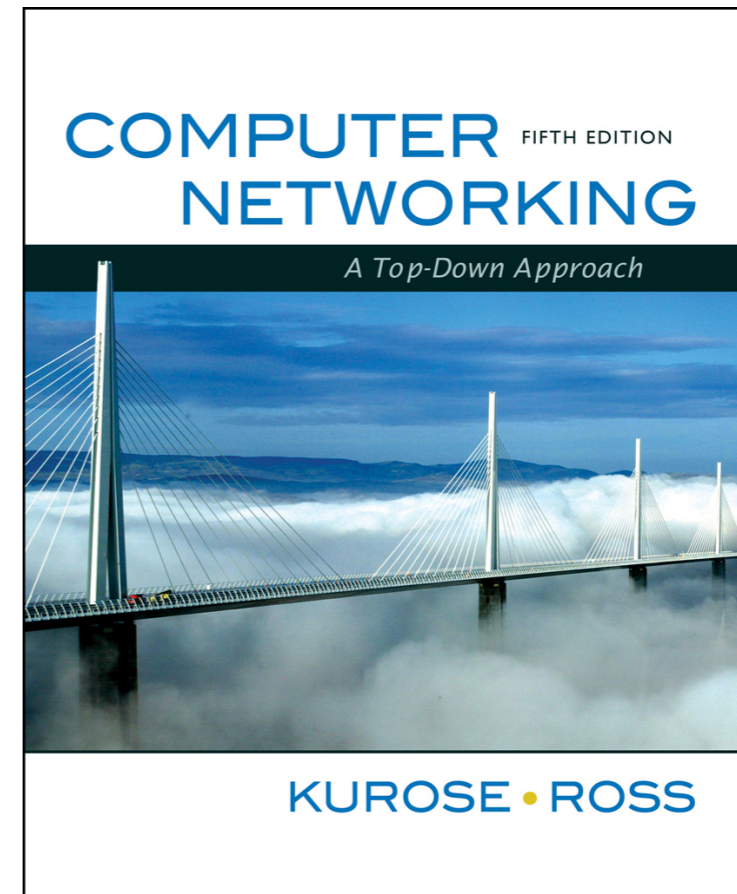
Technische Fakultät

<sup>°</sup>: Rechnernetze und Telematik, \*: Eingebettete Systeme

Albert-Ludwigs-Universität Freiburg

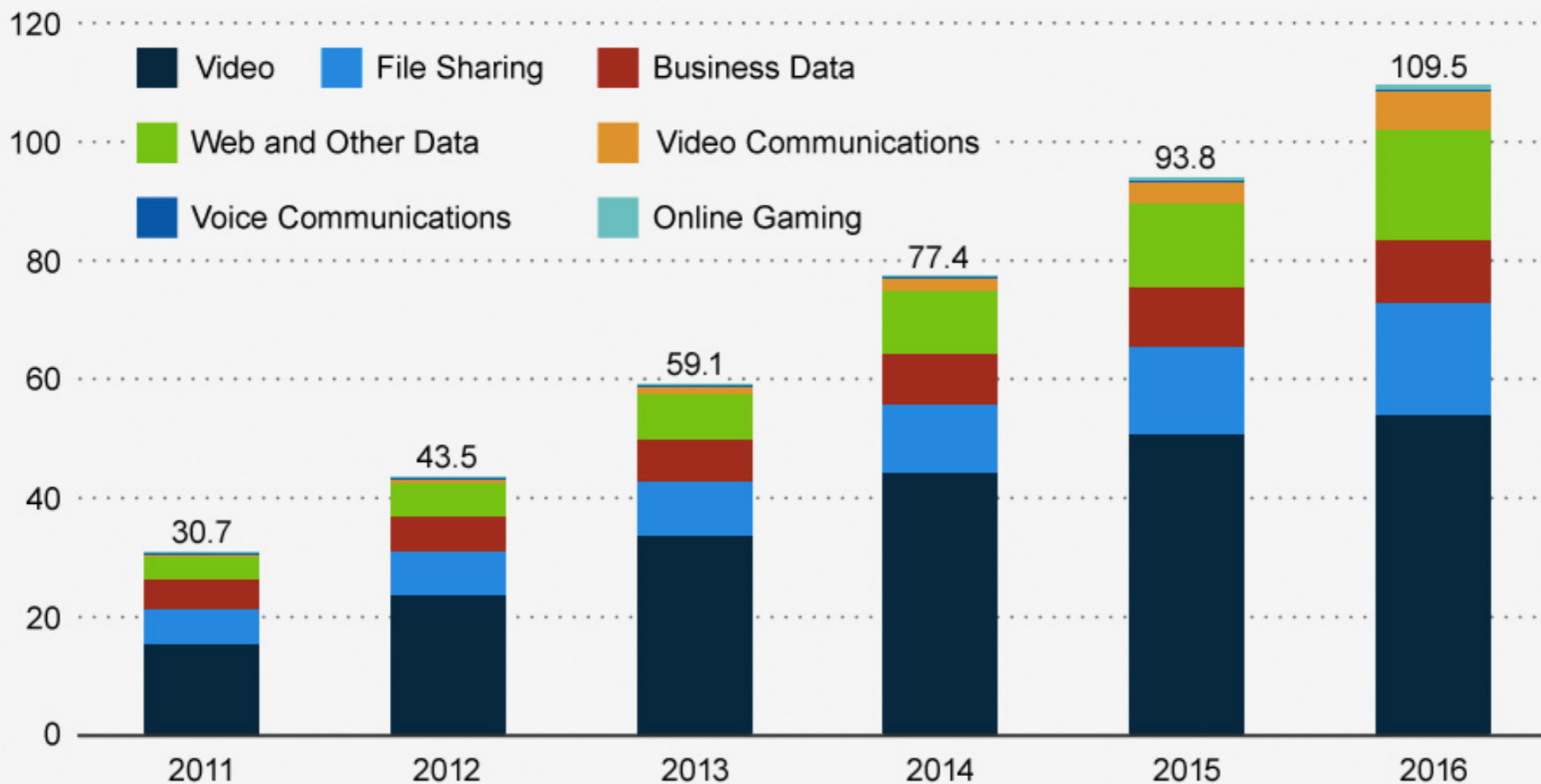
Version 24.04.2015

- Folien und Inhalte aus
  - Computer Networking:  
A Top Down Approach  
5th edition.  
Jim Kurose, Keith Ross  
Addison-Wesley, April 2009.
  - Copyright liegt bei den  
Autoren Kurose und Ross



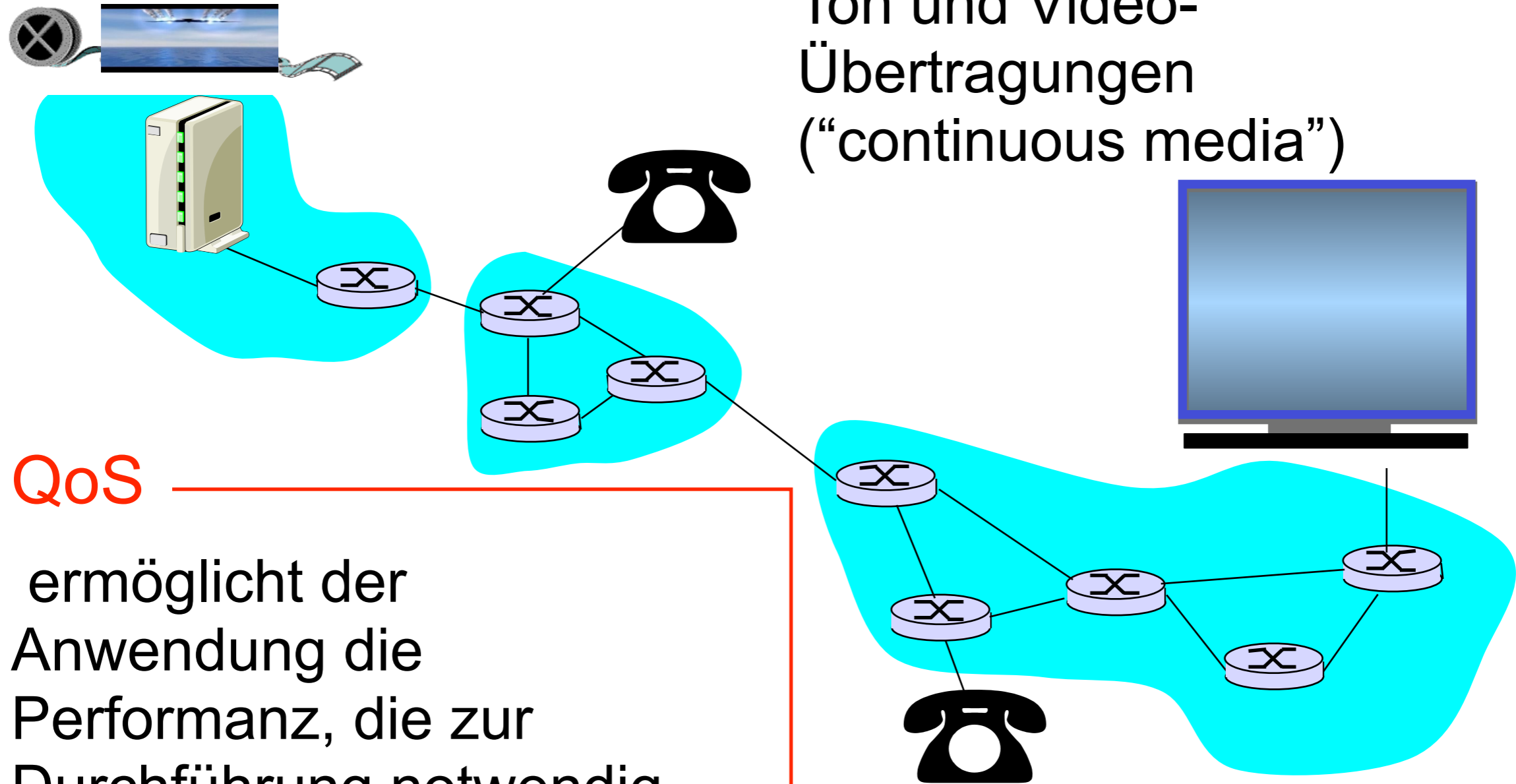
## Video Accounts for Half of Ever-Growing Internet Traffic

Estimated global IP traffic per month (in exabyte)



## Multimedia Anwendungen

Ton und Video-  
Übertragungen  
("continuous media")

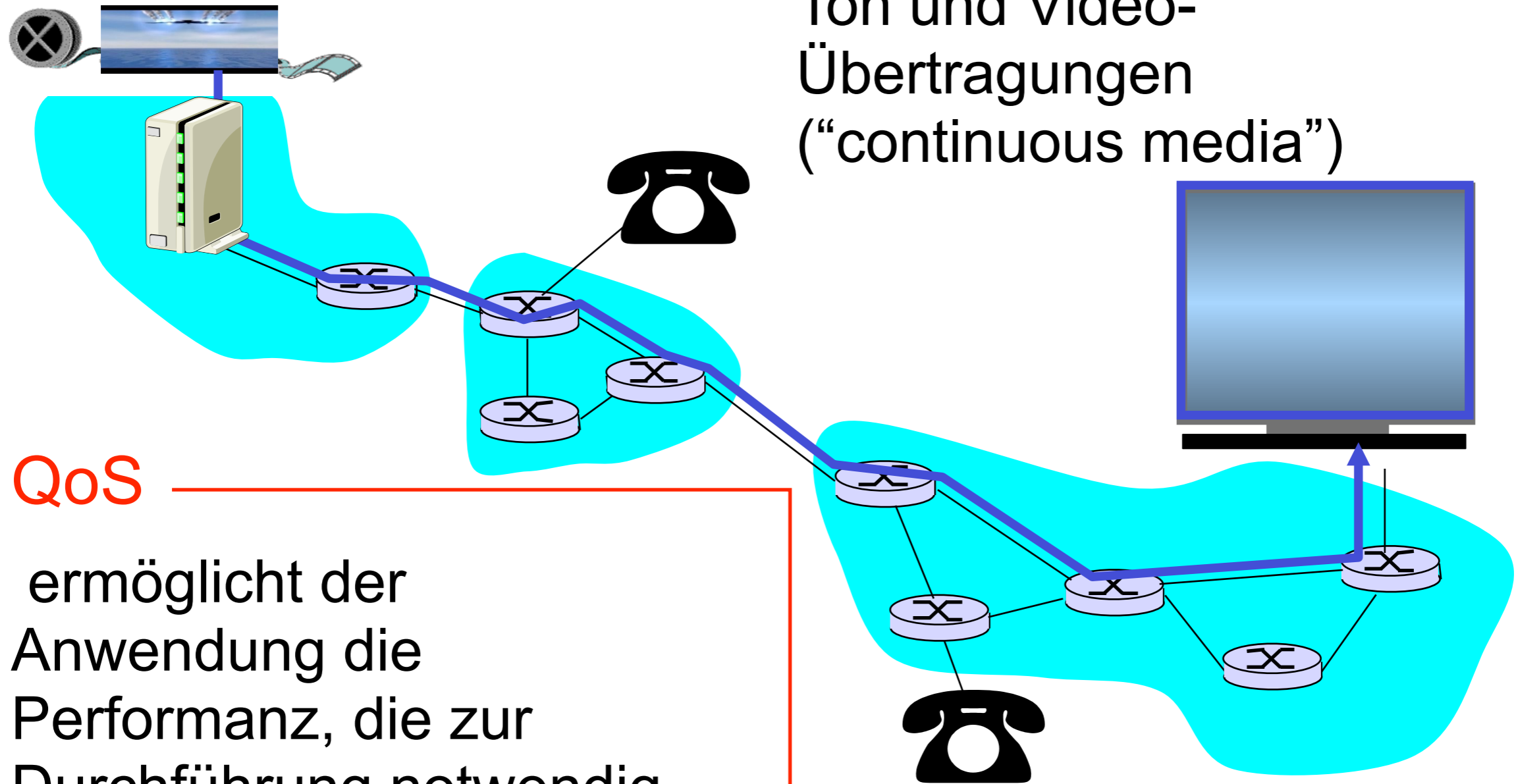


## QoS

ermöglicht der  
Anwendung die  
Performanz, die zur  
Durchführung notwendig

## Multimedia Anwendungen

Ton und Video-  
Übertragungen  
("continuous media")

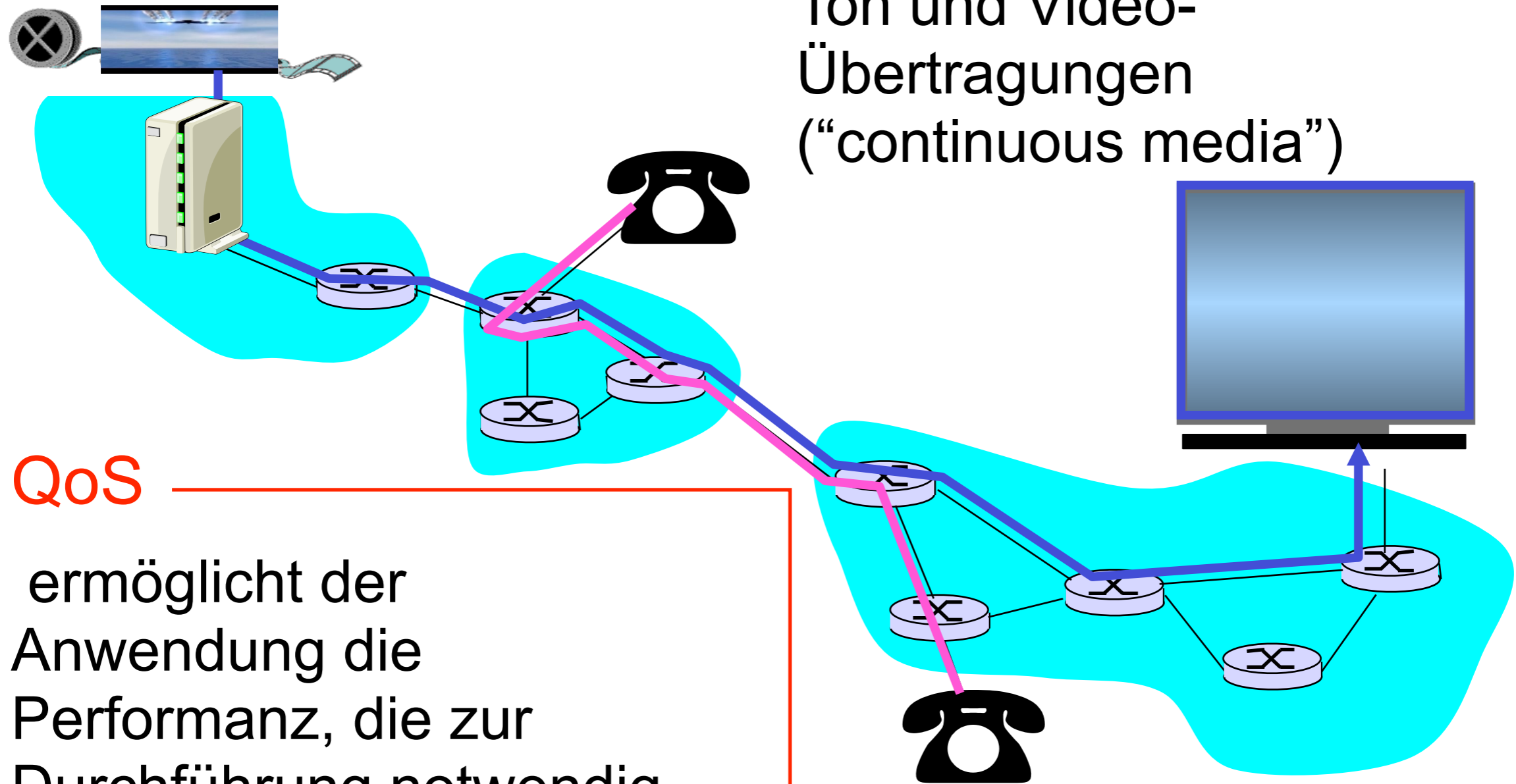


## QoS

ermöglicht der  
Anwendung die  
Performanz, die zur  
Durchführung notwendig

## Multimedia Anwendungen

Ton und Video-  
Übertragungen  
("continuous media")



## QoS

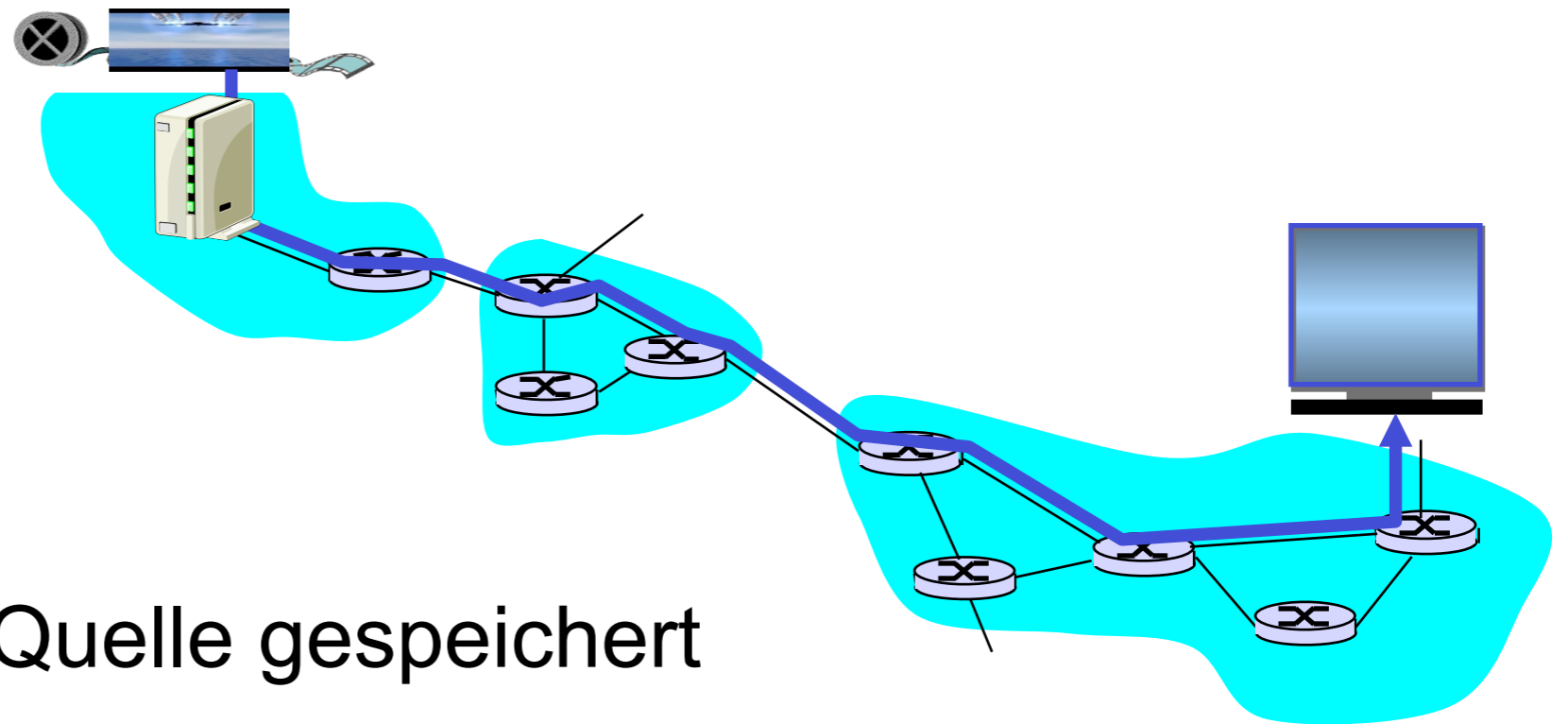
ermöglicht der  
Anwendung die  
Performanz, die zur  
Durchführung notwendig

- Allgemeines
  - Klassifikation von Multimedia-Anwendungen
  - Erkennen der Bedürfnisse von Multimedia-Anwendungen
  - Best-Effort so gut wie möglich nutzen
- Protokolle und Architekturen
  - Protokolle für Best-Effort
  - Mechanismen für QoS
  - Architekturen für QoS

- Klassen
  - 1) Gespeicherte Ströme (stored streaming)
  - 2) Live Streams
  - 3) Interaktiv und Realtime
- Typische Eigenschaften
  - Verzögerungs-Empfindlichkeit (delay sensitive)
    - End-to-End Delay
    - Delay Jitter
      - Jitter ist die Veränderung des Paket-Delays innerhalb des Paketstroms
  - Toleranz von Paketverlusten:
    - seltene Verluste verursachen kleine Aussetzung
    - Gegenkonzept von Datenübertragung, die nicht Verluste toleriert, aber Delays

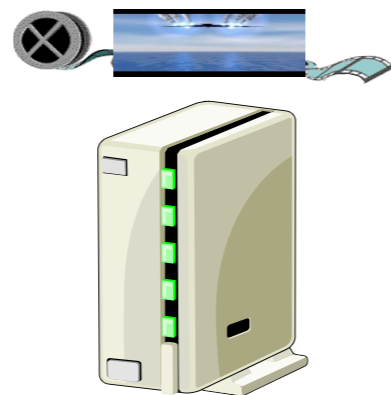
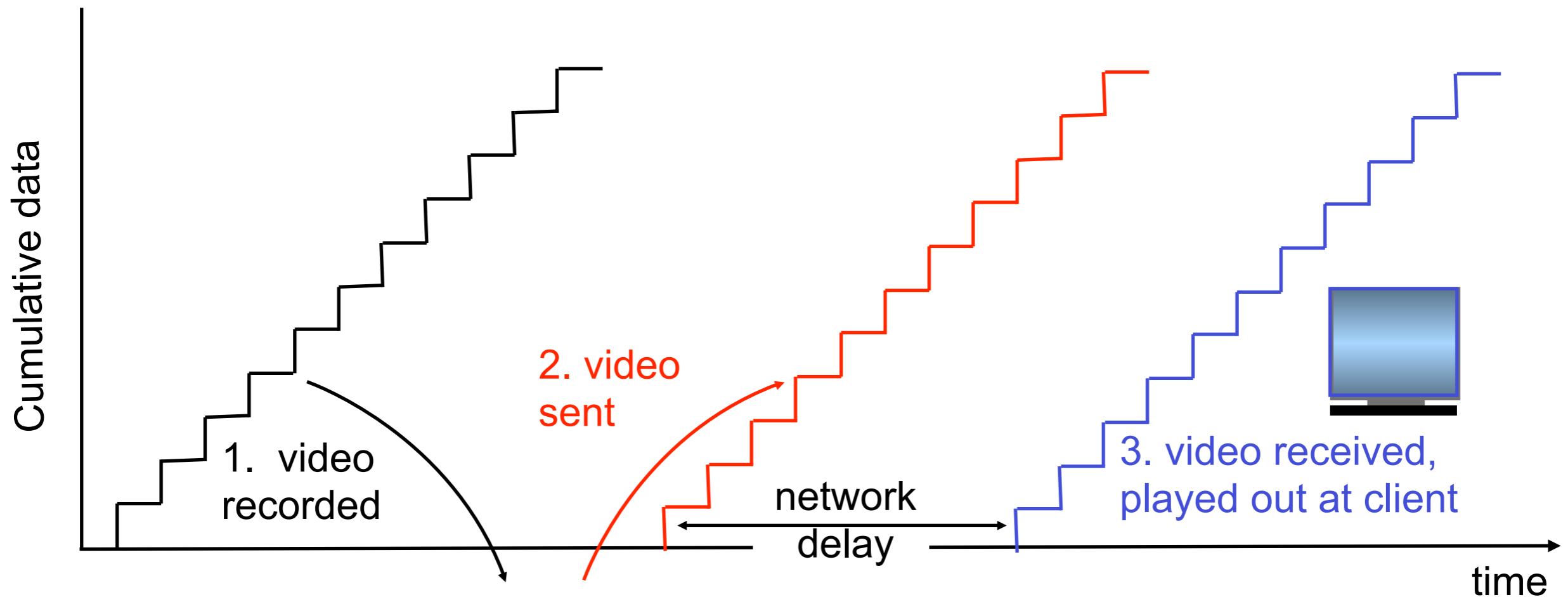


# 1. Gespeicherte Ströme (stored streaming)



- **Stored streaming:**
  - Medium bei der Quelle gespeichert
  - Übertragung zum Client
- **Streaming**
  - Client spielt ab, bevor alle Daten angekommen sind

# 1. Gespeicherte Ströme (stored streaming)



## ■ Streaming

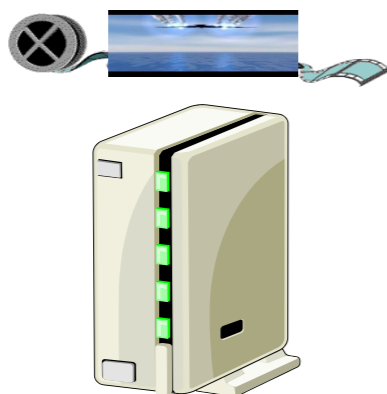
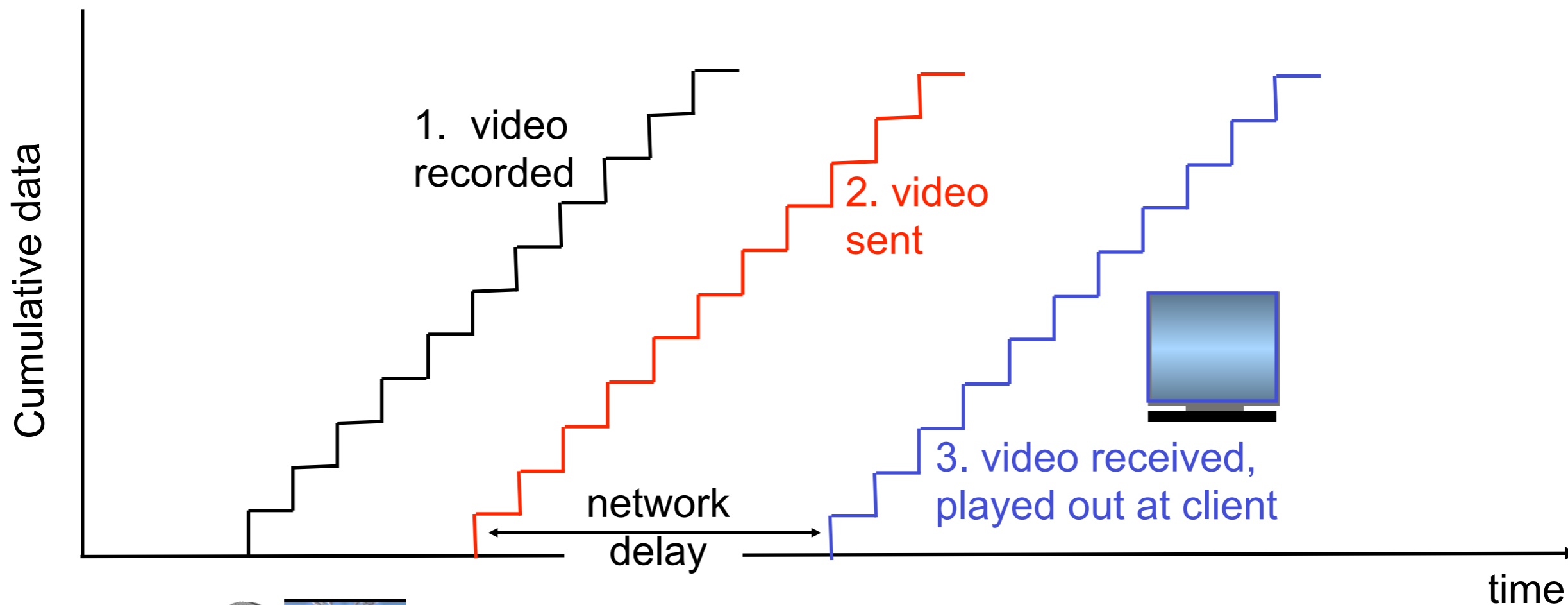
- Client spielt ab, bevor alle Daten angekommen sind

## 2. Live Multimedia Streams

---

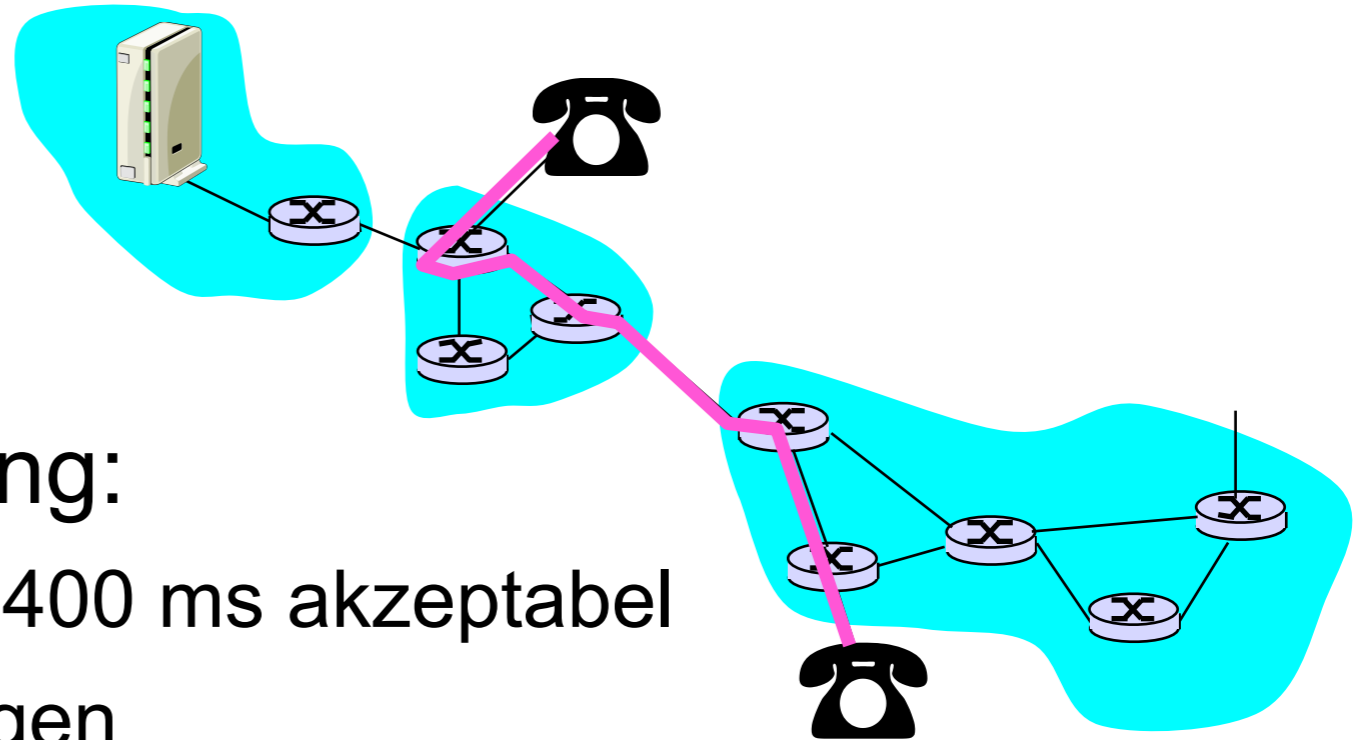
- Beispiele:
  - Internet Radio
  - Sport Ereignis (Fußball..)
- Streaming
  - Wiedergabe-Puffer
  - Playback kann einige Zehntel Sekunden nach der Übertragung stattfinden
  - Dennoch Zeit-Constraints
- Interaktion
  - Vorspulen, Pause, und Zurückspulen möglich

# 2. Live Multimedia Streams



# 3. Real Time Interactive Multimedia

- Anwendungen:
  - IP-Telefonie
  - Video-Konferenz
- Verzögerungsanforderung:
  - Audio:  $< 150$  ms ist gut,  $< 400$  ms akzeptabel
  - inklusive aller Verzögerungen
  - Anwendungsschicht und Netzwerk-Delay
- Größere Delays werden als Einschränkung empfunden
  - Initialisierung der Sitzung



- TCP/UDP/IP: “best-effort”
  - keine Garantien für Delay und Verlustraten
- Wie kann man QoS und Performanz erreichen?
  - Durch Methoden in der Anwendungsschicht
  - schwächen negative Einflüsse auf Delay und Verlust ab

- **Integration einer Service-Philosophie**
  - Fundamentale Veränderungen im Internet zur Reservierung von Bandweiten
  - benötigt neue, komplexe Software in Rechnern und Routern
- **Laissez-faire**
  - keine größeren Veränderungen
  - Mehr Bandweite soweit nötig
  - Verbreitung der Inhalte durch Multicast in der Anwendungsschicht
- **Differenzierte Service-Strategie**
  - kleine Veränderungen im Internet
  - Unterscheidung in erste und zweite Klasse-Pakete durch Priorisierung

- Sample Rate
  - Telefon: 8000 samples/sec
  - CD: 44100 samples/sec
- Sample-Größe diskretisiert
  - Werte durch Bits dargestellt
  - z.B.  $2^8=256$  mögliche Werte
  - 8 bits for 256 values
- Beispiel: 8000 samples/sec, 256 Werte ergibt 64000 bps
  - Receiver wandelt Bits zurück zum Analogsignal um
  - mit gewissen Qualitätseinbußen
- CD: 1411 kbps
- MP3: 96, 128, 160 kbps
- Internet Telephonie: 5,3 kbps oder mehr



- Video: Bildsequenzen mit konstanter Rate
  - z.B. 24 Bilder/sec
- Digitales Bild: Pixelfeld
  - Jedes Pixel wird durch Bits dargestellt
- Redundanzen
  - im Raum (innerhalb des Bilds, z.B. eintönige Flächen)
  - in der Zeit (von einem Bild zum nächsten, z.B. Standbilder)
- Beispiele:
  - MPEG1 (CD-ROM) 1.5 Mbps
  - MPEG2 (DVD) 3-6 Mbps
  - MPEG4 (Internet, < 1 Mbps)
- Forschung:
  - geschichtetes (skalierendes) Video
  - Schichten passen sich an die Bandweite an

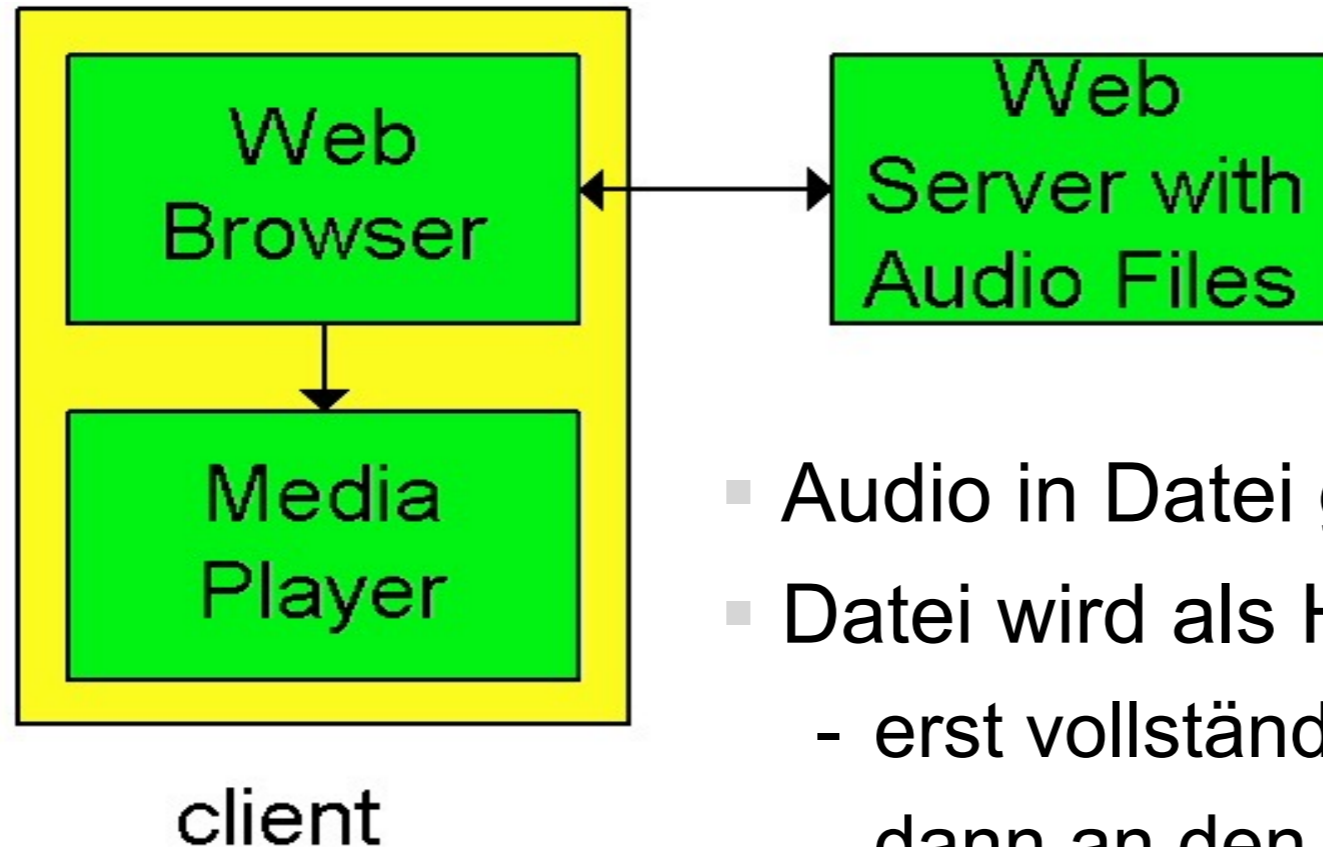
- Anwendungsschicht für Best-Effort-Service

- Client-Puffer
- UDP oder TCP
- Verschiedene Kodierungsmethoden für Multimedia

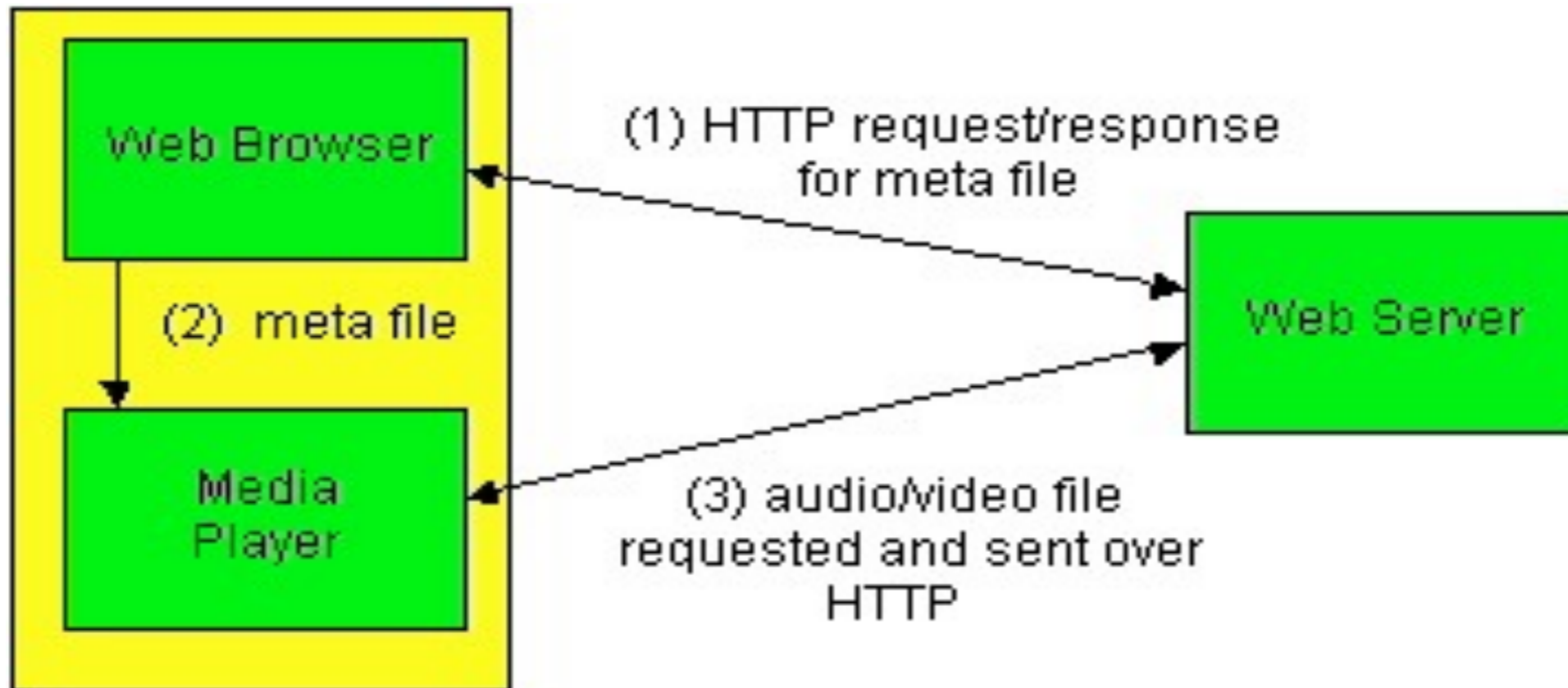
## Media Player

- jitter-Entfernung
- Dekomprimierung
- Fehlerbehandlung
- GUI

# Internet Multimedia: Der einfachste Ansatz

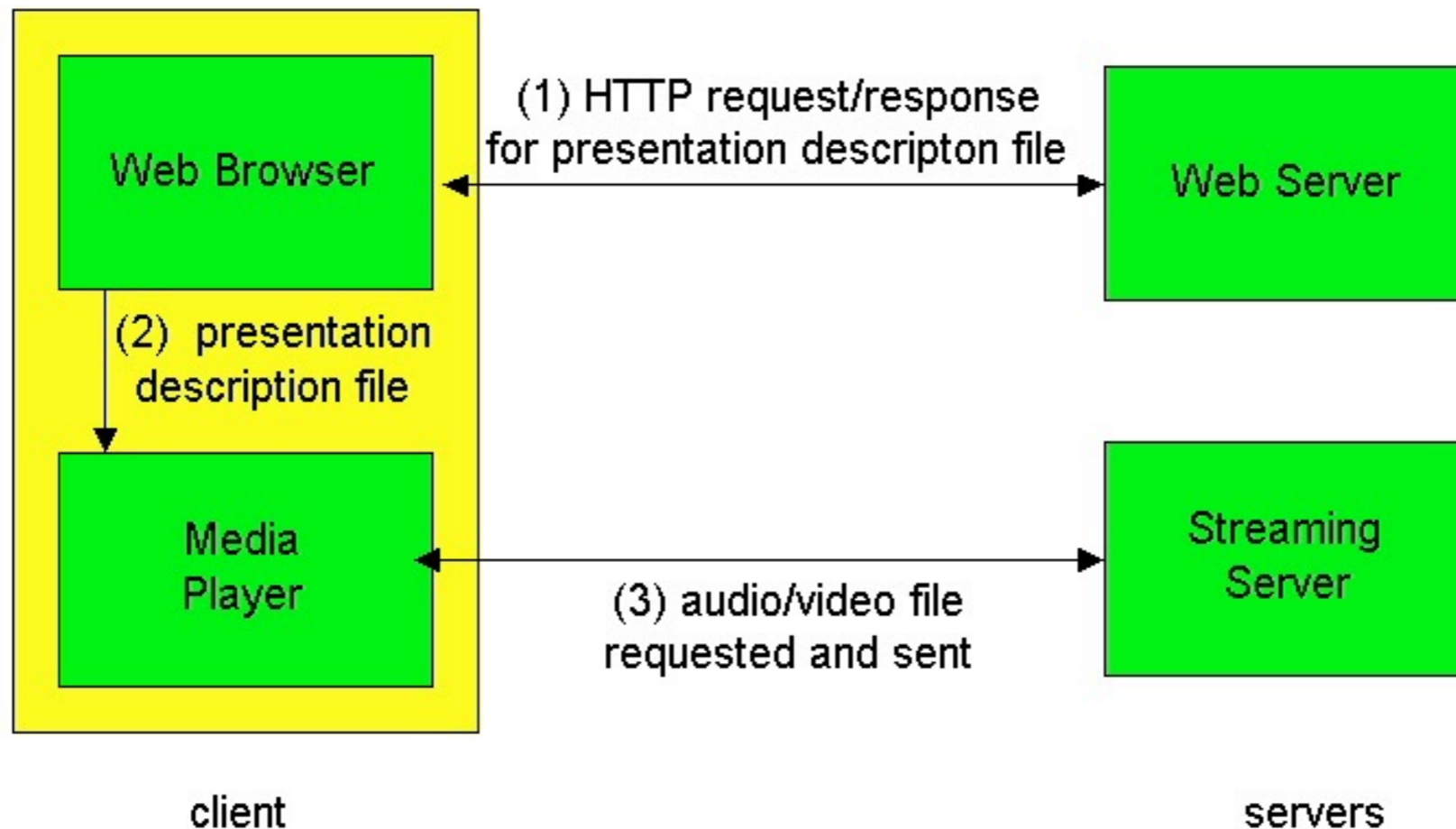


- Audio in Datei gespeichert
- Datei wird als HTTP-Objekt übertragen
  - erst vollständig empfangen
  - dann an den Player weiter gegeben
- Kein Audio oder Video-Stream
- Kein Pipelining
- Extrem lange Delay bis zum Spielbeginn

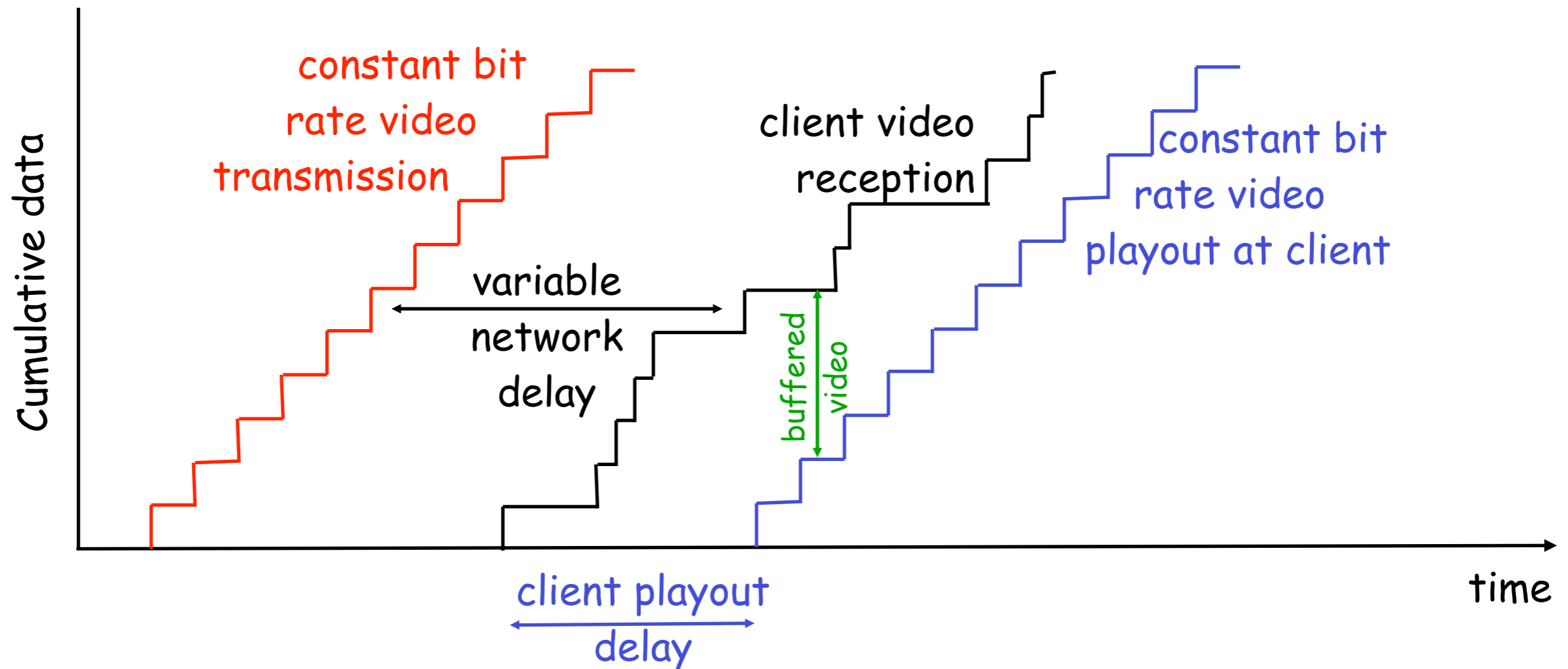


- Browser empfängt Metafile durch GET
- Browser startet Player
  - gibt Metafile weiter
- Player kontaktiert Server
- Server „stream“t Audio und Video zum Player

# Internet Multimedia: Streaming von einem Streaming-Server



- Erlaubt Nicht-HTTP-Protokoll zwischen Server und Media-Player
- UDP oder TCP Medienübertragung (3)



- **Puffer auf der Client-Seite**
  - Abspielverzögerung kompensiert Netzwerk-Delay und Delay-Jitter