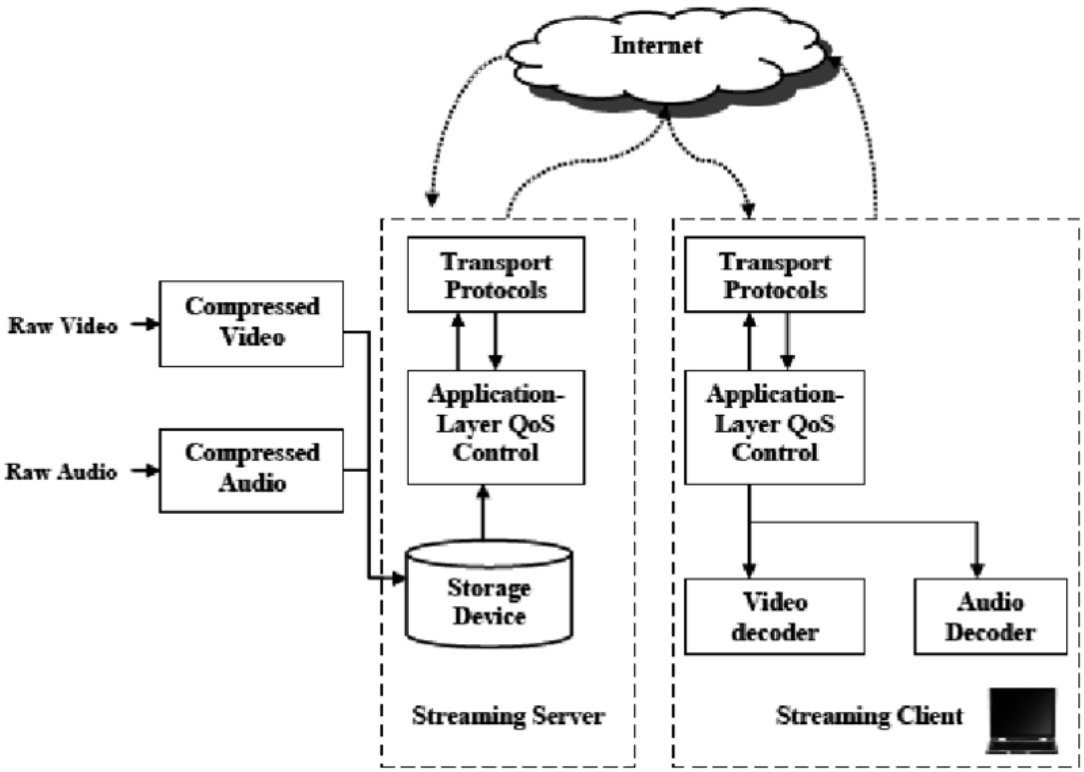
MEDIA STREAMING WITH IBM CLOUD VIDEO STREAMING

DESING:



video streaming and it is divided into six areas as follows: media compression, application-layer QoS control, media distribution services, streaming servers, media synchronization at the receiver side, and streaming media protocols

Media compression:

The large volume of raw multimedia data imposes a stringent bandwidth requirement on the network. Hence, for achieving better transmission efficiency, compression is widely employed. While video needs superior bandwidth requirements (56 Kbps-15 Mbps) than audio (8 Kbps-128 Kbps) and loss of audio is more infuriating to human than video, audio is given higher priority for transmission in a multimedia streaming system. For this reason, only video will be used for alteration so as to meet the QoS requirements [6]. In Figure 1, raw video and audio data are pre-compressed by video compression and audio compression algorithms and then saved in storage devices. Video compression is accomplished by utilizing the resemblances or redundancies that subsists in a normal video signal.

Video Streaming compression:

Internet streaming technology also brings in more interesting applications such as transmission of traditional TV content in a much

more flexible manner. Due to cost considerations, conventional TV networks normally offer channels only if there are enough user bases. For example, a TV network may be willing to offer Hindi programs in New York City where a large Indian population live, but not in many other parts of the country [36]. The introduction of live streaming services enables users to watch several TV channels through the Internet simultaneously. In live streaming, video streams are being generated at the same time as it is being downloaded and viewed by the clients. So, we are dealing with the distribution of a file of unknown and unpredictable length in which the data are only available for a small period of time.

Streaming Video over the Internet

Characteristics of Multimedia Data

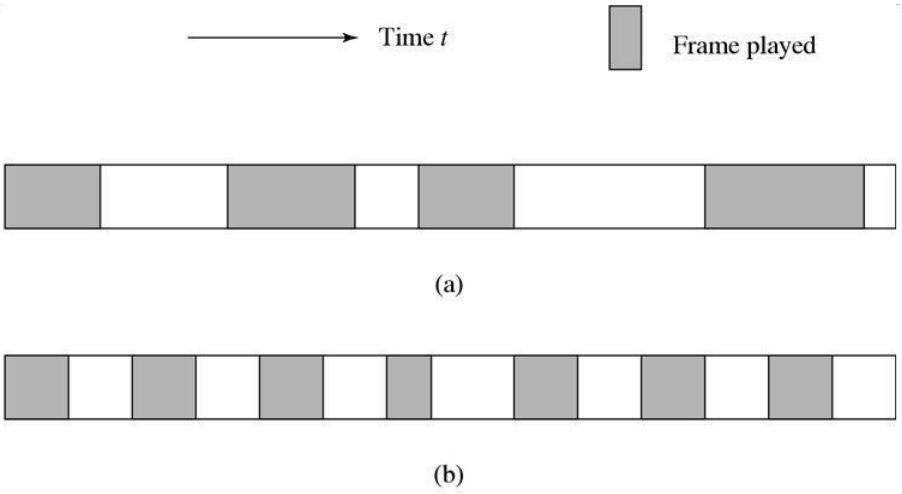
- **Voluminous** — they demand very high data rates, possibly dozens or hundreds of Mbps.

- **Real-time and interactive** — they demand low delay and synchronization between audio and video for "lip sync". In addition, applications such as video conferencing and interactive multimedia also require two-way traffic.

- **Sometimes bursty** — data rates fluctuate drastically, e.g., no traffic most of the time but burst to high volume in video-on-demand.

Quality of Multimedia Data Transmission

- **Quality of Service (QoS)** depends on many parameters:
  - **Data rate**: a measure of transmission speed.
  - **Latency (maximum frame/packet delay)**: maximum time needed from transmission to reception.
  - **Packet loss or error**: a measure (in percentage) of error rate of the packetized data transmission.
  - **Jitter**: a measure of smoothness of the audio/video playback, related to the variance of frame/packet delays.



- Fig. 1: Jitters in frame playbacks. (a) High jitter, (b) Low jitter.

Multimedia Service Classes

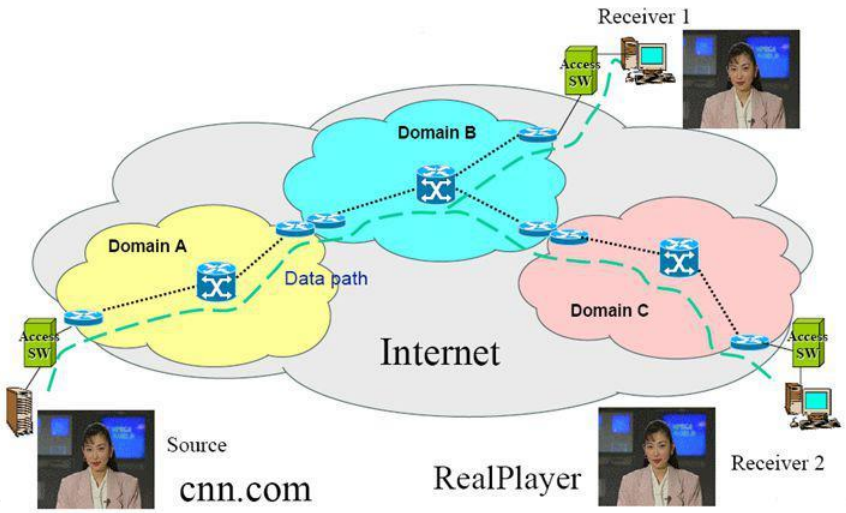
- **Real-Time** (also *Conversational*): two-way traffic, low latency and jitter, possibly with prioritized delivery, e.g., voice telephony and video telephony.
- **Priority Data**: two-way traffic, low loss and low latency, with prioritized delivery, e.g., E-commerce applications.
- **Silver**: moderate latency and jitter. One-way traffic, e.g., streaming video, or two-way traffic (also *Interactive*), e.g., web surfing, Internet games.
- **Best Effort** (also *Background*): no real-time requirement, e.g., downloading or transferring large files (movies).
- **Bronze**: no guarantees for transmission.

- Table 2: Tolerance of Latency and Jitter in Digital Audio and Video

Application	Avg Latency Tolerance (msec)	Avg Jitter Tolerance (msec)
Low-end videoconf. (64 kbps)	300	130
Compressed voice (16 kbps)	30	130
MPEG NTSC video (1.5 Mbps)	5	7
MPEG audio (256 kbps)	7	9
HDTV video (20 Mbps)	0.8	1

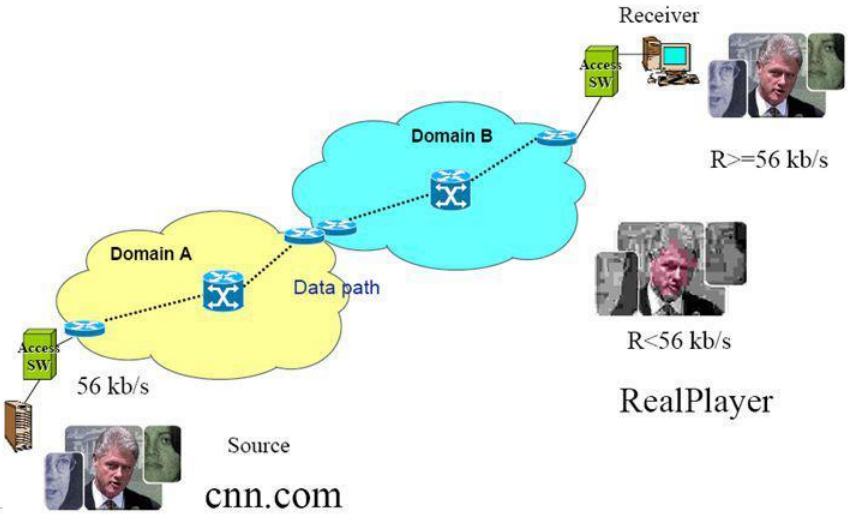
What is Steaming Video

- Download mode: No delay bound
- Streaming mode: delay bound

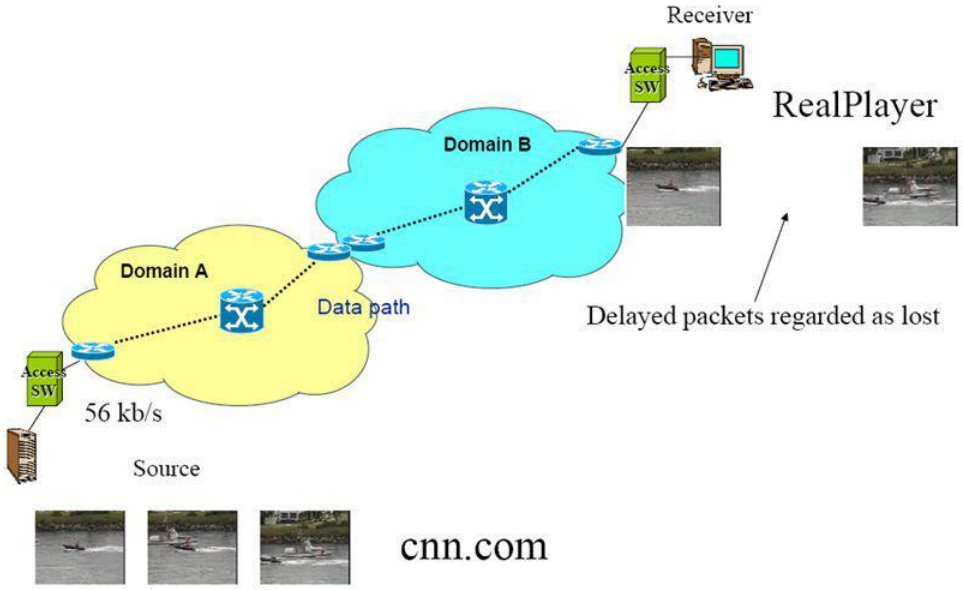


Challenges for Quality Video Transport

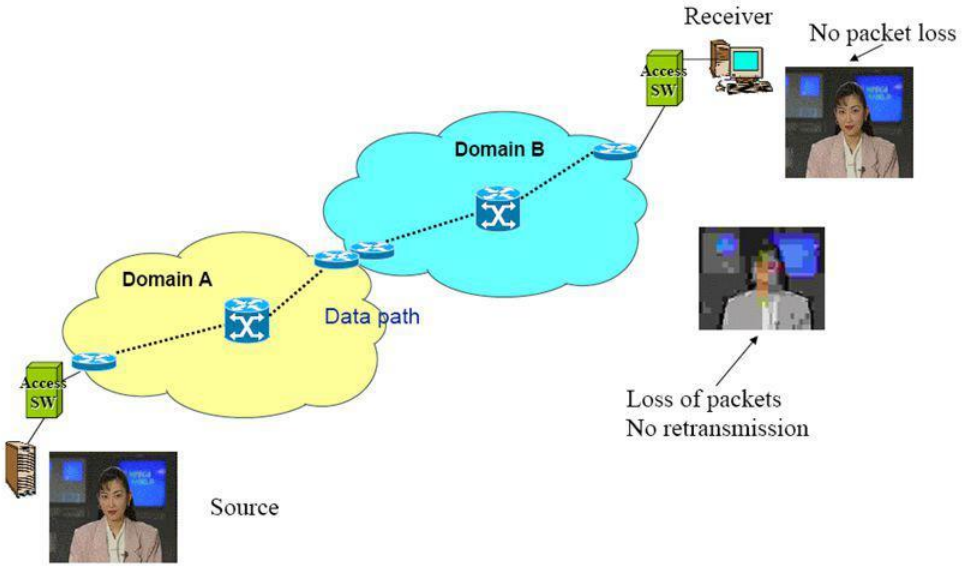
- Time-Varying Available Bandwidth
  - No bandwidth reservation

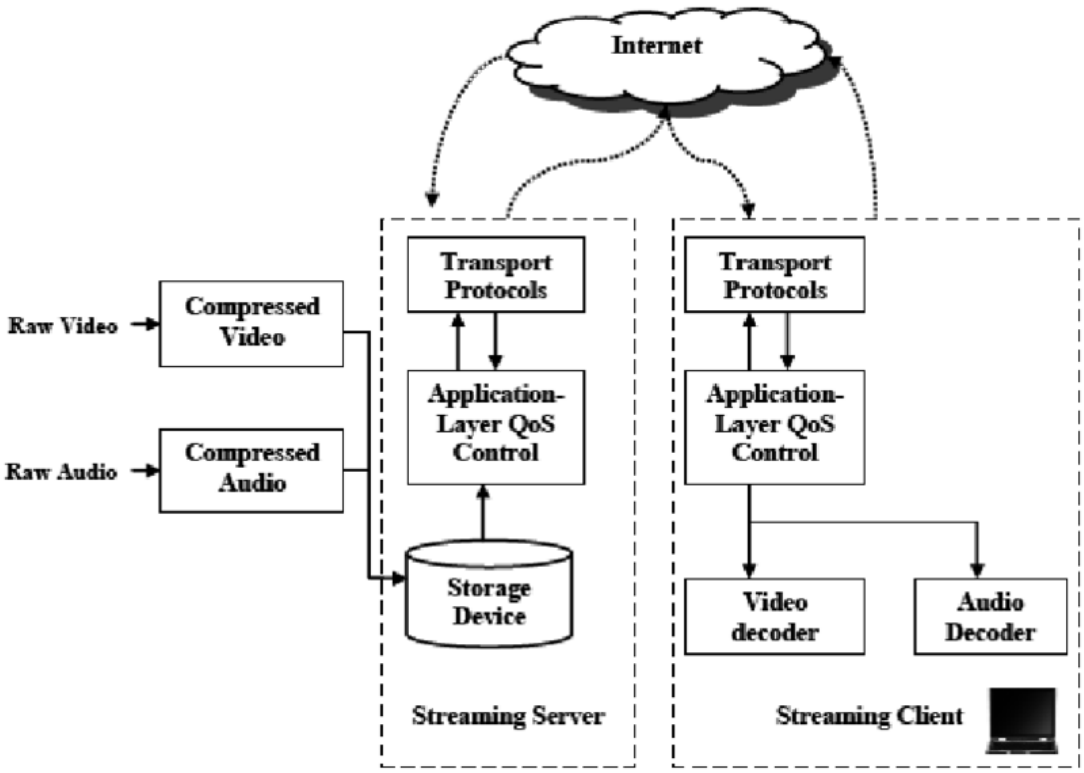


Time-Varying Delay



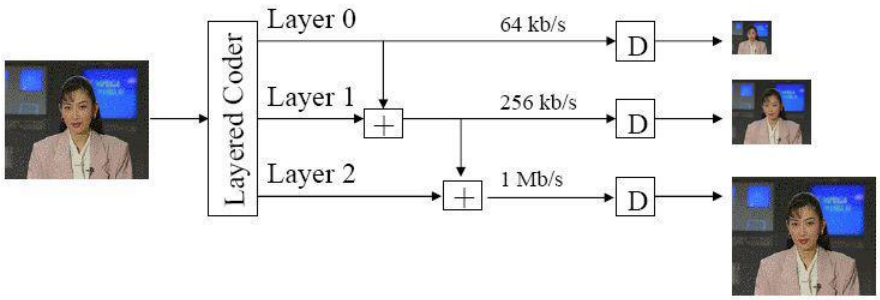
Effect of Packet Loss




Internet
Transport Protocols
Transport Protocols
Raw Video
Compressed Video
Application-Layer QoS Control
Application-Layer QoS Control
Raw Audio
Compressed Audio
Storage Device
Video decoder
Audio Decoder
Streaming Server
Streaming Client

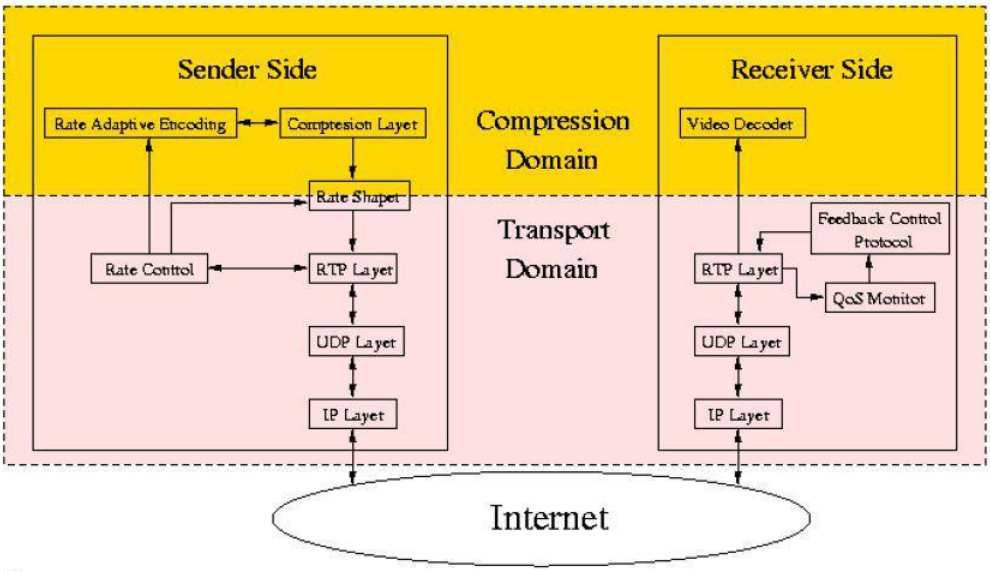
Video Compression

- Layered Video encoding/decoding
- D denotes the decoder



Congest Control

- Source-based Rate Control



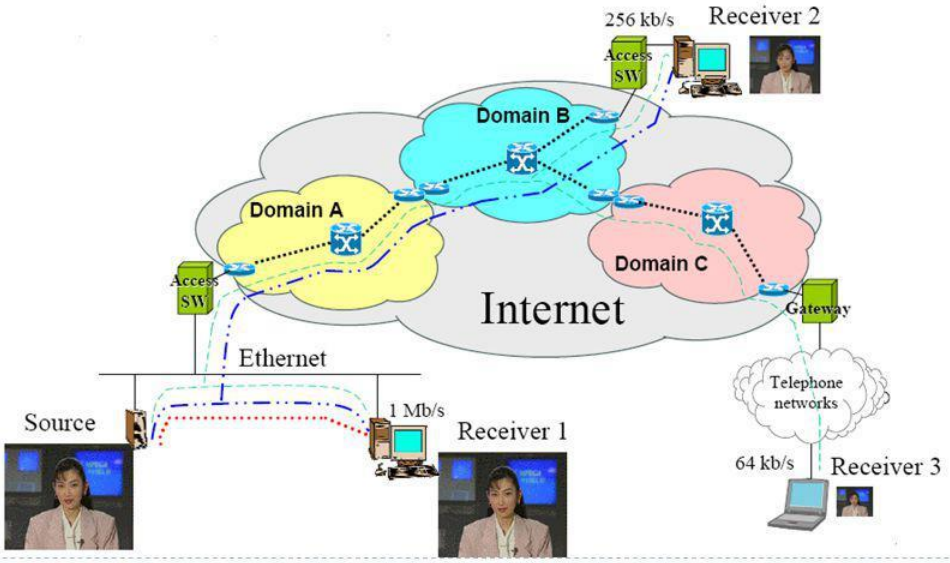
Mode-based Approach

$$\lambda = \frac{1.22 \times MTU}{RTT \times \sqrt{p}}$$

Where is λ the throughput of a TCP connection, MTU (maximum transmission unit) is the packet size used by the connection, RTT is the round-trip time for the connection, and p is the packet-loss ratio experienced by the connection.

Receiver-based Rate Control

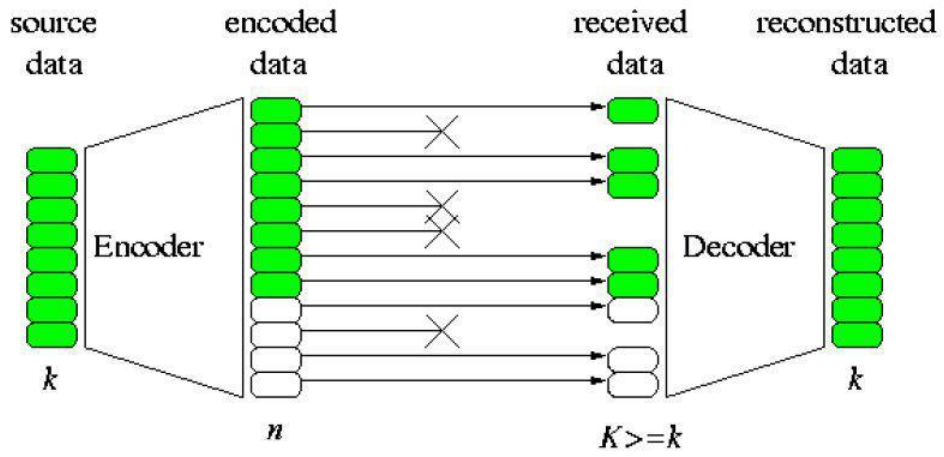
- IP multicast for layered video



Error Control

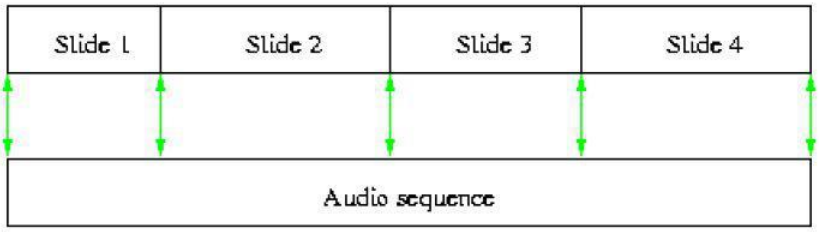
- FEC
  - Channel coding
  - Source coding-based FEC
  - Joint source/channel coding
- Delay-constrained retransmission
- Error resilient Compression
- Error concealment

Channel Coding



Media Synchronization

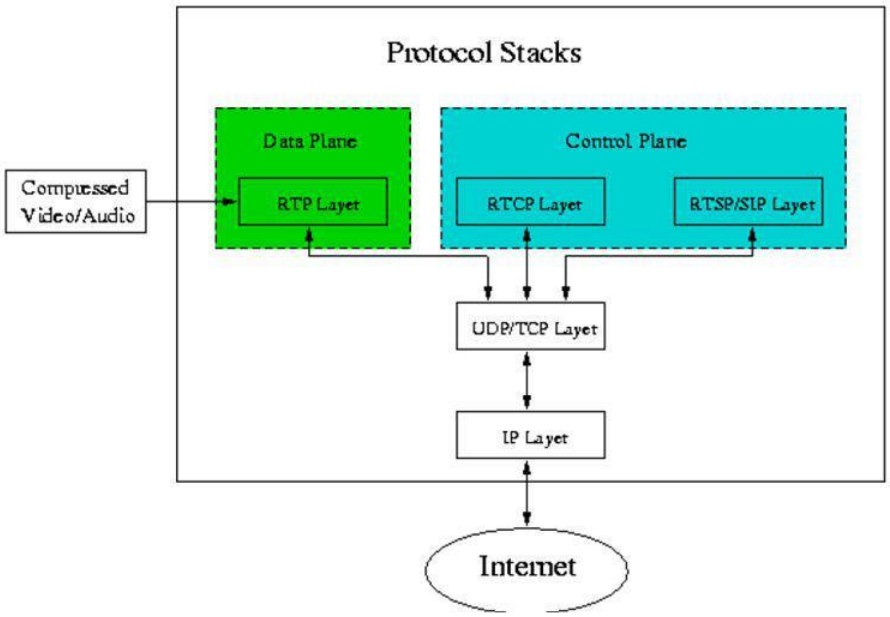
- Why media synchronization
- Example: lip-synchronization (video/audio)



Protocol for Streaming Video

- Network-layer protocol: Internet Protocol (IP)
- Transport protocol
  - Lower layer: UDP & TCP
  - Upper layer: Real-time Transport Protocol (RTP) & Real-Time Control Protocol (RTCP)
- Session control protocol
  - Real-time streaming protocol (RTSP): RealPlayer
  - Session Initiation Protocol (SIP) : Microsoft Windows MediaPlayer; Internet Telephony

Protocol Stacks



Summary

- Challenges for Quality video transport
  - Time-varying available bandwidth
  - Time-varying delay
  - Packet loss
- An architecture for video streaming
- Application-layer QoS control
- Streaming server
- Media synchronization mechanisms
- Protocols for steaming media

```java
package io.john.amiscaray.videosharingdemo.repo;

import io.john.amiscaray.videosharingdemo.domain.Video;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.data.jpa.repository.Query;
import org.springframework.stereotype.Repository;

import java.util.List;

@Repository
public interface VideoRepo extends JpaRepository<Video, Long> {
    Video findByName(String name);

    boolean existsByName(String name);

    @Query(nativeQuery = true, value="SELECT name FROM video")
    List<String> getAllEntryNames();
}
```

COMPLETE..