PI-Grau
(Internet Protocols)
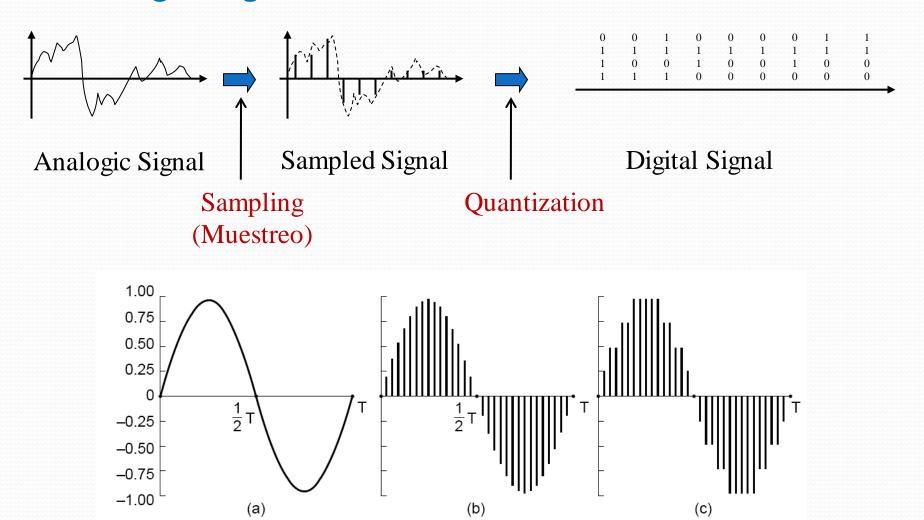
José M. Barceló Ordinas
Departamento de Arquitectura de Computadores
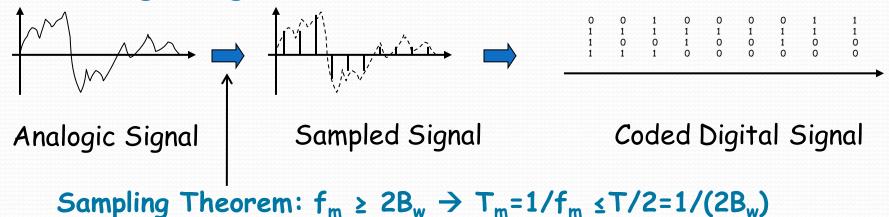**(UPC)**

- **Topic 7: Application and Services II.**

  - Objectives

    - Multimedia coding

    - Identify the main architectures and protocols involved in **multimedia applications**

    - Understand the **QoS architectures** in Internet

- **Multimedia (MM) applications and supporting protocols:**

  - Includes the combination of several content forms such as text, still images, video, audio, etc

  - We will here **introduce** audio, image and video **coding techniques**

  - We will here introduce **a map with the needed protocols** in Internet to communicate multimedia content

  - We will introduce the main **Internet architectures to support QoS** (Quality of Service) content

- ## **Analogic/Digital (A/D) conversion:**

0	0	1	0	0	0	0	1	1
1	1	1	1	1	1	0	1	1
1	0	0	1	0	0	1	0	0
1	1	1	0	0	0	0	0	0

Analogic Signal   Sampled Signal   Digital Signal

Sampling
(Muestreo)   Quantization

- **Analogic/Digital (A/D) conversion:**

Analogic Signal   Sampled Signal   Coded Digital Signal

Sampling Theorem: $f_m \geq 2B_w \rightarrow T_m=1/f_m \leq T/2=1/(2B_w)$

F_m : Sampling Frecuency (samples/seg) $\geq$ 2x Bandwidth
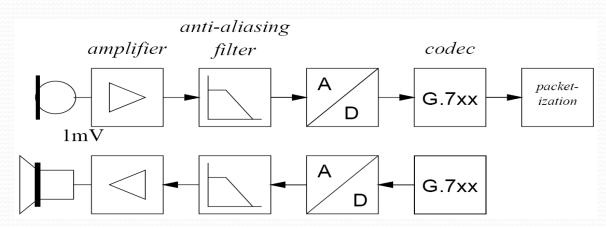b_m: Bits/sample (Quantization process)

For transmitting a digital signal we need $\boxed{R= F_m \times b_m}$ bits/seg

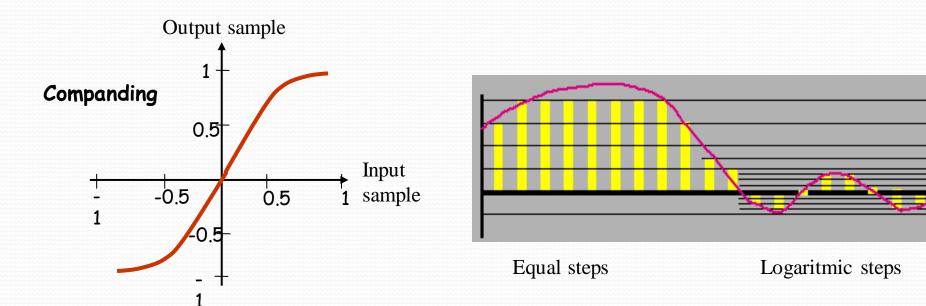Ex: Voice with phone quality. F_m = 8 KHz, b_m = 8 $\rightarrow$ R = 64 Kbps
Ex: Compact Disk (CD): F_m = 44.1 KHz, b_m = 16 $\rightarrow$ R(estéreo) = 1.411 Mbps

- **Audio:**
  - **G.7xx:** group of standards belonging to ITU-T (International Telecommunication Union) to compress voice
    - Sampling at 8KHz, 8-bit/sample for a maximum of 64 Kb/s data rate with a typical delay of 0.125 ms.
    - **G.711:** Logarithmic Pulse Code Modulation
    - **G.721/G.726:** AD-PCM (Adaptive Differential - Pulse Code Modulation)
    - **G.727:** same that G.726 but optimized in order to allow Packet Circuit Multiplexing Equipments (PCME) environemts
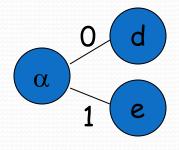
- **G.711: Phone networks.** Use of non-lineal quantization (companding=compressing+expanding): A-law (Europa), μ-Law (USA/Japan).
  - The idea is that high signal amplituds are less probable than low ones, then expand (small steps) the low amplituds (logaritmically) and compress (large steps) the high ones $\rightarrow$ uniform quantization
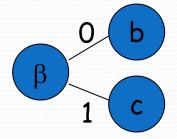

Equal steps   Logaritmic steps

- **Analogic/Digital (A/D) conversion:**

  - The explained coding technique is called **PCM (Pulse Code Modulation).**

  - *Compression* (also called *source coding*) is used to reduce the number of bits needed for transmission or for storing information:

    - **Lossless Compression (without losses):** the original information is exactly recovered (e.g., ZIP, GZIP, GIF, TIFF)

    - **Lossy Compression (with losses):** some info is lost but compression ratios are higher. Used in audio and images (e.g., JPEG, MPEG, MP3, …)
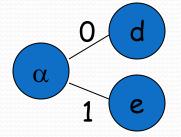
- ## Huffman coding (lossless compression):
  - Optimum technique, but relative symbol frequencies have to be known.
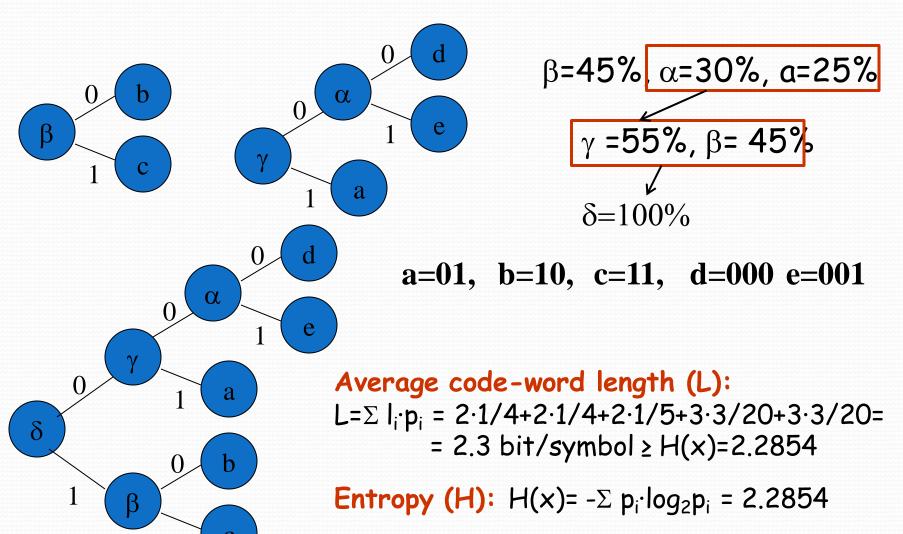 - Ex: a=25%, b=25%, c=20%, d=15%, e=15%

α=30%, a=25%, b=25%, c=20%
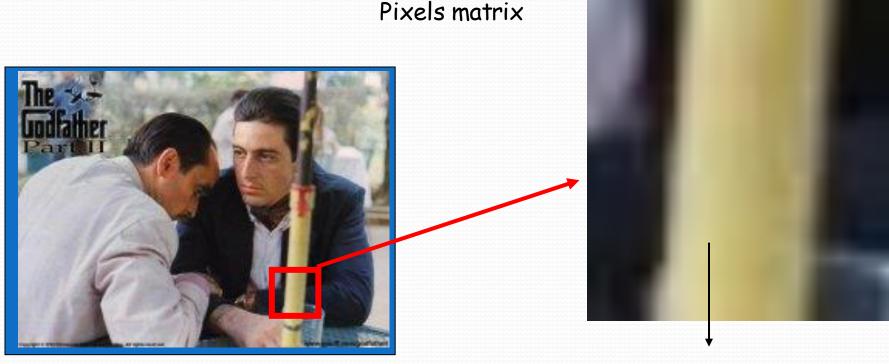
α=15+15=30%

β=45%, α=30%, a=25%

β=25+20= 45%

Huffman coding (lossless compression):

β=45%, α=30%, a=25%

γ =55%, β= 45%

δ=100%

a=01, b=10, c=11, d=000 e=001

Average code-word length (L):
L=Σ $l_i \cdot p_i$ = 2·1/4+2·1/4+2·1/5+3·3/20+3·3/20=
= 2.3 bit/symbol ≥ H(x)=2.2854

Entropy (H): H(x)= -Σ $p_i \cdot \log_2 p_i$ = 2.2854

- **Multimedia (MM) applications and supporting protocols:**

  1. A MM file is loaded before is being played (**stored media**) (*e.g.*: emule).

  2. If the file is big enough to be loaded before being played (disk space, latency), we can download a few seconds before playing. Thus, jitter (in the order of seconds) is compensated (**streaming**) (e.g. WindowsMedia)

  3. The information have to arrive in few miliseconds since it is inter-active or because is played in real-time (**continous playout**) (e.g. VoIP, Internet radio)
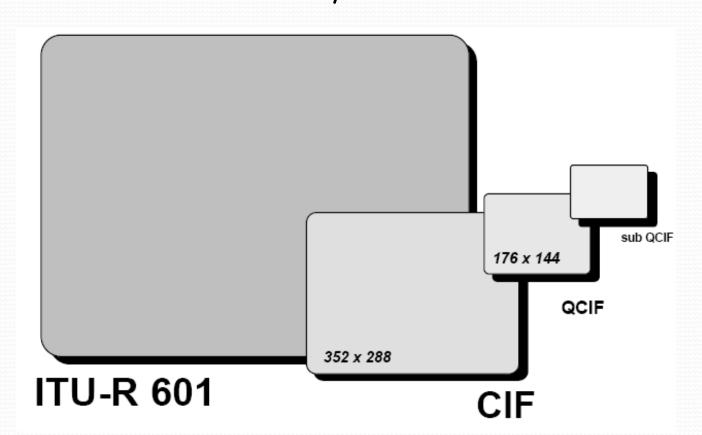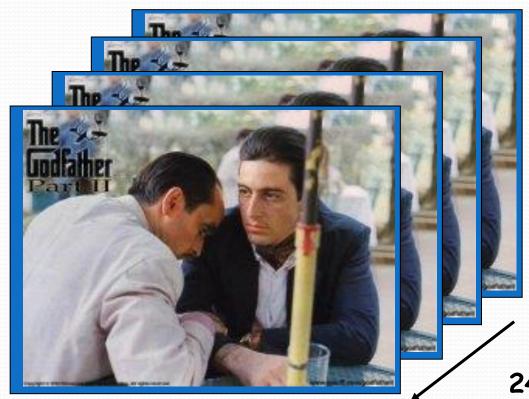
Multimedia (MM) principles:

Pixels matrix

100011101001001001001001001001001011

Each pixel is represented by a set of bits (luminance y colour).

- **Multimedia (MM) applications:**
  - **Formats:** Number of pixels per frame (picture). For example a CIF frame has 352x288 pixels → without compression that means in 24-bit/pixel RGB coding using HSL (hue-saturation-lightness) or HSV (hue-saturation-value) = 3x352x288 = 304.128 Bytes/frame x 30frames/s = 9.123.840 Bytes/s = 72.990.720 b/s.
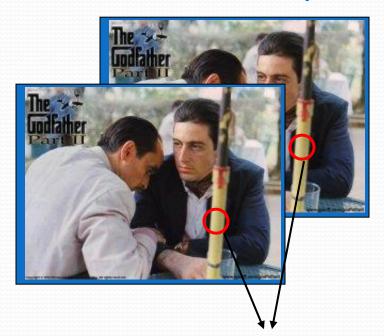
- **Multimedia (MM) principles:**
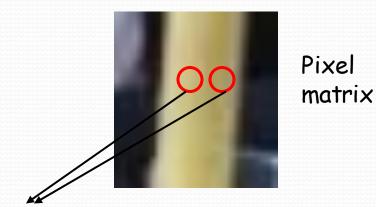
24-30 images per second

- **Multimedia (MM) principles:**

Pixel matrix

Spatial Redundancy : nearby points in the same image have similar luminance and colour

Temporal Redundancy : the same points in consecutive images have similar luminance and colour

Still Imágenes: JPEG, GIF

Video:
MPEG-1 (CD-ROM): 1.5 Mbps
MPEG-2 (DVD): 3-6 Mbps
H.261, H.263 (ISDN, videotelephony):
64Kbps,< 30 Kbps
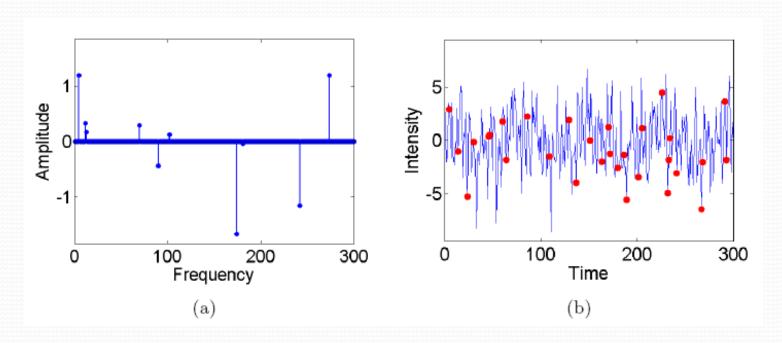
Compression principles:

- Assume a sampled time signal with interval T: e.g. produces samples (coefficients) that later are coded with Huffman coding.

- Assume now that we transform this signal in another domain in such a way, that the number of coefficients that represent the signal is less than the original number of samples !!!!

  - For example take the following time domain signal and see the frequency domain signal.

(a)

(b)

- **Compression principles:**
  - **Transform step:**
    - The source $\{x_n\}$ is divided into blocks of size N. Each block is mapped into a transform sequence $\{e_n\}$ using a reversible mapping.
    - Most of the energy of the transformed block was contained in few elements of the transformed values.
  - **Quantization step:**
    - The transformed sequence is quantized based on the following strategy:
      - The desired average bit rate.
      - The statistics of the various transformed elements.
      - The effect of distortion on the reconstructed sequence.
  - **Entropy coding step:**
    - The quantized data are entropy-coded using Huffman or other techniques.
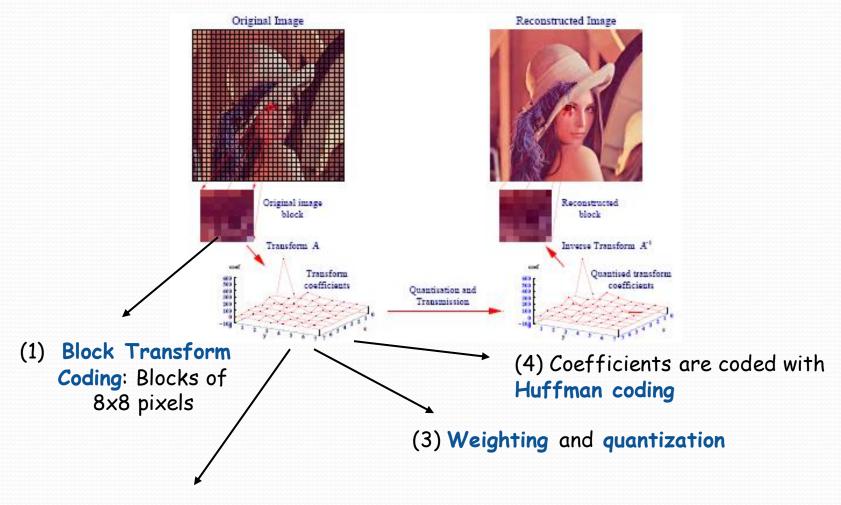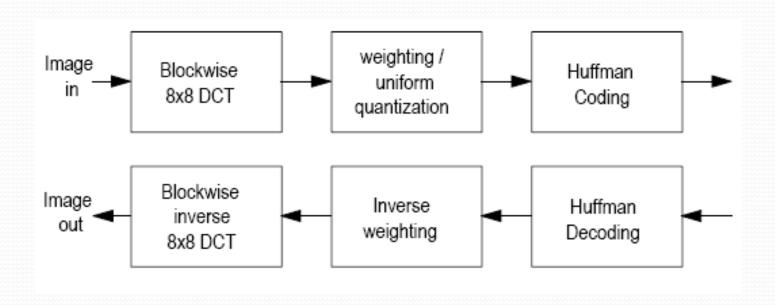
- **Multimedia (MM) principles:**

(1) **Block Transform Coding**: Blocks of 8x8 pixels

(4) Coefficients are coded with **Huffman coding**

(3) **Weighting** and **quantization**

(2) DCT (**Discrete Cosine Transform**) coefficients are calculated

- **Multimedia (MM) principles:**

- **Example of coding in JPEG (wikipedia):**

$$\begin{bmatrix} 52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\ 63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\ 62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\ 63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\ 67 & 61 & 68 & 104 & 126 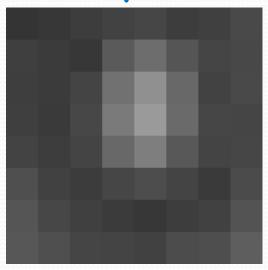& 88 & 68 & 70 \\ 79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\ 85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\ 87 & 79 & 69 & 68 & 65 & 76 & 78 & 94 \end{bmatrix} . \quad \text{8x8 matrix}$$

8x8 sub-image

If the range is on [0,N]=[0,255], then center the coefficients on the middle point 128

$$g = \begin{bmatrix} -76 & -73 & -67 & -62 & -58 & -67 & -64 & -55 \\ -65 & -69 & -73 & -38 & -19 & -43 & -59 & -56 \\ -66 & -69 & -60 & -15 & 16 & -24 & -62 & -55 \\ -65 & -70 & -57 & -6 & 26 & -22 & -58 & -59 \\ -61 & -67 & -60 & -24 & -2 & -40 & -60 & -58 \\ -49 & -63 & -68 & -58 & -51 & -60 & -70 & -53 \\ -43 & -57 & -64 & -69 & -73 & -67 & -63 & -45 \\ -41 & -49 & -59 & -60 & -63 & -52 & -50 & -34 \end{bmatrix} \Big\downarrow y.$$

- ## Example of coding in JPEG (wikipedia):

DCT-II (type II): u,v (0≤u,v<8) are the new coordinates, $g_{x,y}$ are the old coordinates

$$G_{u,v} = \frac{1}{4}\alpha(u)\alpha(v)\sum_{x=0}^{7}\sum_{y=0}^{7} g_{x,y} \cos\left[\frac{(2x+1)u\pi}{16}\right] \cos\left[\frac{(2y+1)v\pi}{16}\right]$$

$$\alpha(u) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } u = 0 \\ 1, & \text{otherwise} \end{cases}$$

Matrix after DCT

$$u \longrightarrow$$

$$G = \begin{bmatrix}
-415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\
4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\
-46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\
-48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\
12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & -2.79 & 3.14 \\
-7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\
-1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\
-0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68
\end{bmatrix} \Bigg\downarrow v.$$

- **Example of coding in JPEG (wikipedia):**

Normalize with respect a quantization matrix Q:

$$B_{j,k} = \text{round}\left(\frac{G_{j,k}}{Q_{j,k}}\right) \text{ for } j = 0, 1, 2, \ldots, 7; k = 0, 1, 2, \ldots, 7$$

$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}.$$
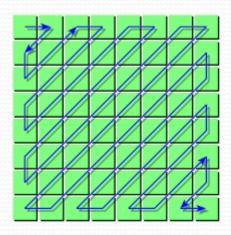
Matrix after quantization

$$B = \begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -3 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

- **Example of coding in JPEG (wikipedia):**

Store using Zig-Zag and apply Huffman compression using a dictionary :

```
-26
-3    0
-3   -2    -6
 2   -4    1    -3
 1    1    5     1     2
-1    1   -1     2     0    0
 0    0    0    -1    -1    0    0
 0    0    0     0     0    0    0    0
 0    0    0     0     0    0    0
 0    0    0     0     0    0
 0    0    0     0     0
 0    0    0     0
 0    0    0
 0    0
 0
```

Values taken in zigzag →

- **Multimedia (MM) principles:**
  - **Predictive coding:** predict the value of the pixels when these are unknown
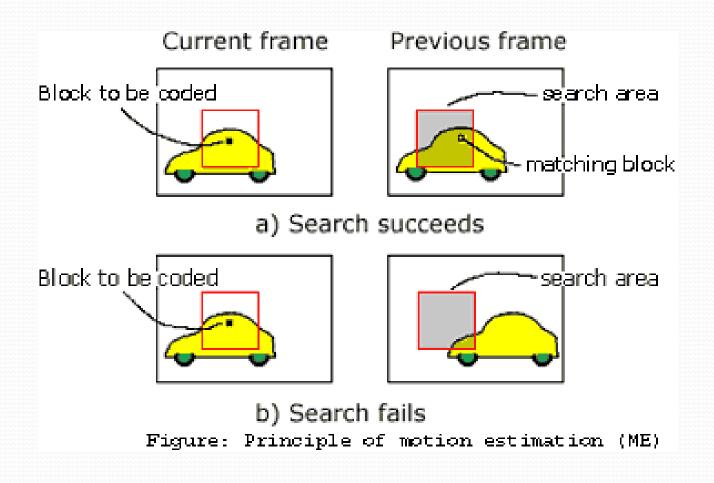
Fig. 2. Predictive coding of image pixels. X: pixel to be predicted and coded. A, B, C: already coded and stored or transmitted pixels. JPEG lossless coding standard specifies the following possible predictors: $X = A, X = B, X = C, X = A + B - C, X = A + (C - B)/2, X = (A + C)/2, X = C + (A - B)/2$.
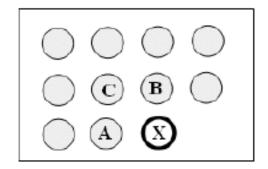
- ## Multimedia (MM) principles:
  - Motion Estimation/Compensation (ME/MC)

Figure: Principle of motion estimation (ME)

- **Multimedia (MM) principles:**

  - Typically first apply Motion Compensation (MC), and on the error frame apply DCT

    ⇒ INTRA coded Frame (**I-Frame**): apply DCT to a block of NxN pixels (N=8)

    ⇒ Predictive Coded Frame (**P-Frame**): apply MC and later DCT over the error frame. Code movement vectors and quantized coefficients of the DCT.

- **Multimedia (MM) applications:**
  - Still Images coding
    - **JPEG** good compression of images (but for example, would poorly compress a fax).
      - Standard for still images black/white and color
      - Lossless coding (predictive) or lossy coding (DCT).
      - Exploits properties taken from eye human perception (example: color changes are less perceptible than bright)
      - Represents colors with 24 bits
      - Variable compression ratio (24:1, acceptable quality loss)

    - **GIF** proprietary method for lossless compression (LZW).
      - Represents colors with 8 bits (**PNG**: non-propietary, 24 bits)

- **Multimedia (MM) applications:**

  - ITU-T standards
    - **H.261**
      - ≤ 128 Kbps (ISDN)
      - Formats CIF or QCIF.
      - 7.5-30 frames/s
      - Videoconference systems
    - **H.263**
      - ≤ 30 Kbps (RTC)
      - CIF, QCIF, Sub-QCIF
      - ≤ 10 fps
      - Internet video streaming
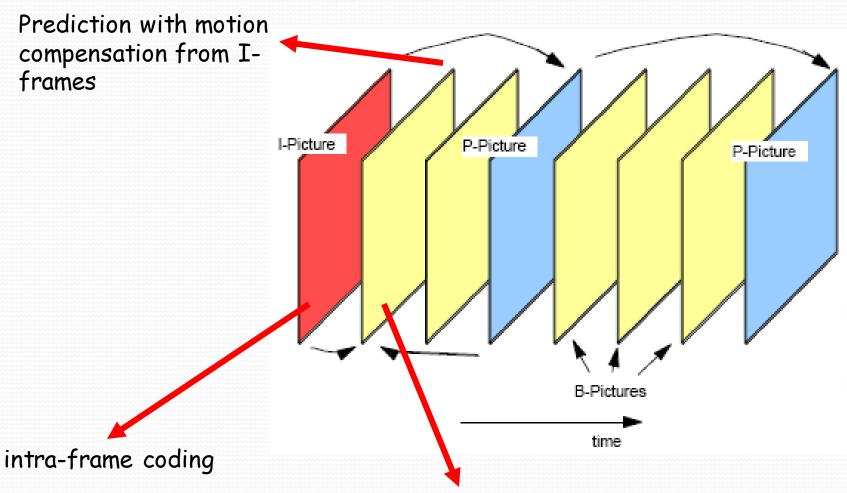
- **Multimedia (MM) applications:**
  - ISO MPEG
    - MPEG-1
      - 1.5 Mbps
      - CIF, no interleaving
      - 24-30 frames/seg
      - Video-storing (CD-ROM)
    - MPEG-2
      - 4-8 Mbps forTV (PAL in Europa)
      - 20 Mbps for HDTV
    - MPEG-4
      - Absorbs many features of MPEG-1 and MPEG-2
    - MPEG-7, MPEG-21, MPEG-A,B,C,D,E,V,M,U,...

- **Multimedia (MM) applications:**

  - **GOP (Group of Pictures):**
    - Sequence of frames that specifies the order in which Intra frames and Inter frames are arranged
    - **I-frames** (Intra Coded Picture): reference frames, uses DCT compression
    - **P-frames** (Predective Coded Picture): contains motion-compensation from the previous I-frame or P-frame
    - **B-frames** (Bidirectionally Predective Coded Picture): contains motion compensation from the preceding and following I or P-frames
    - GOP (M,N):
      - M distance between consecutive P frames
      - N total number of the sequence (distance between consecutive I-frames)
    - GOP(3,12) → IBBPBBPBBPBB IBBPBBPBBPBB IBBPBBPBBPBB ....
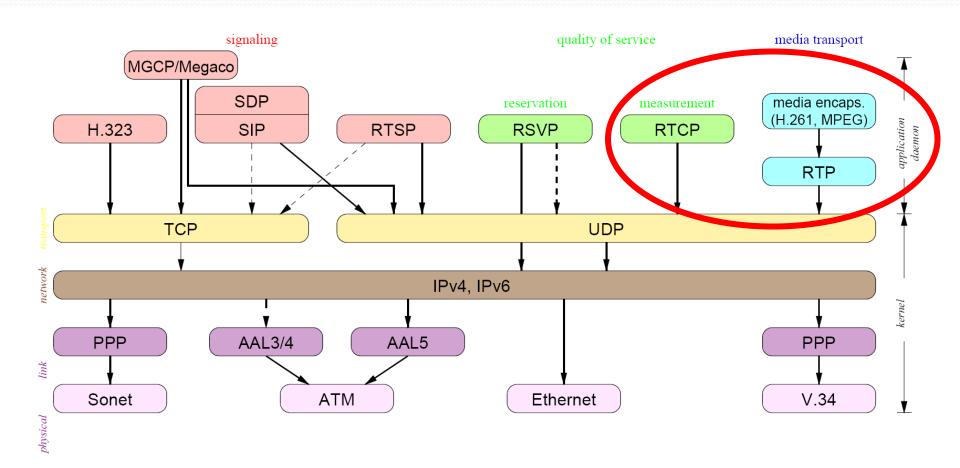
- ## Multimedia (MM) applications:

Prediction with motion compensation from I-frames

intra-frame coding

Prediction with motion compensation from consecutive I-frames and P-frame
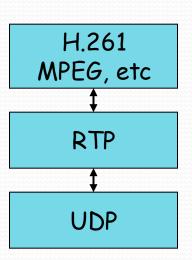
- **Multimedia (MM) supporting protocols:**

- **Multimedia (MM) supporting protocols:**
  - ¿How can the process know:
    - time at which has to replay the MM information
    - lost packets
    - change in packet order
    - type of coding, … ?

  - **RTP (Real-time Transport Protocol):** deals with the transfer of real-time data. Information provided by this protocol include **timestamps** (for synchronization), **sequence numbers** (for packet loss and reordering detection) and the **payload format** which indicates the encoded format of the data.

  - **RTCP (RTP Control Protocol):** used to **specify quality of service** (QoS) feedback and synchronization between the media streams.

RTP (**R**eal-time **T**ransport **P**rotocol):

• RTP allows real-time information encapsulation.
• Adds information that facilitates the task of recovering the correct signal at the receiver:
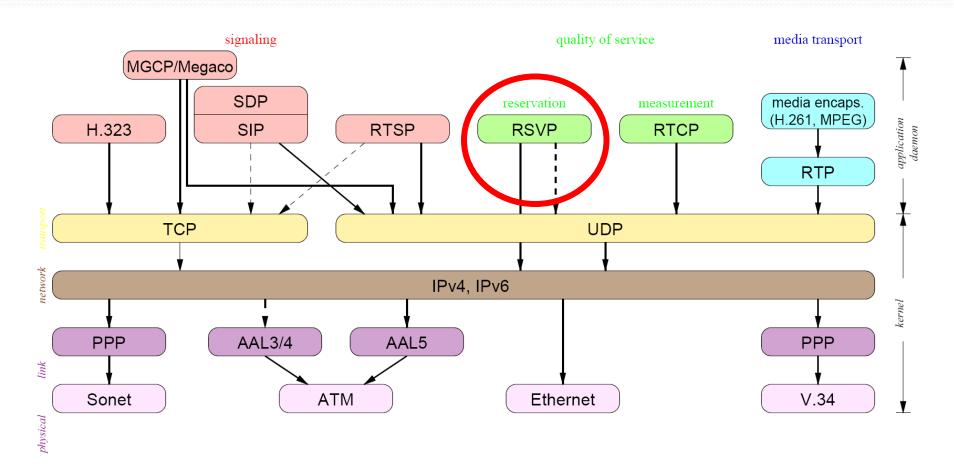
 - Payload Type (PT):

   3: GSM, 14: MPEG audio, 31: H.261,

   32: MPEG1 video, 33: MPEG2 video, etc...

 - Timestamps: Timing
 - Sequence numbers: Losses
 - SSRC: Stream (random number)

H.261 MPEG, etc
↕
RTP
↕
UDP

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|V=2|P|X|   CC  |M|     PT      |       sequence number         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           timestamp                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           synchronization source (SSRC) identifier            |
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

- **RTCP (RTP Control Protocol):**

  - Protocol jointly used with RTP.
  - Allows sending from Transmitter to Receiver statistics that are useful at the time of coding/decoding the transmission/reception of information on real-time
    - SSRC of the RTP flow from which we send the info
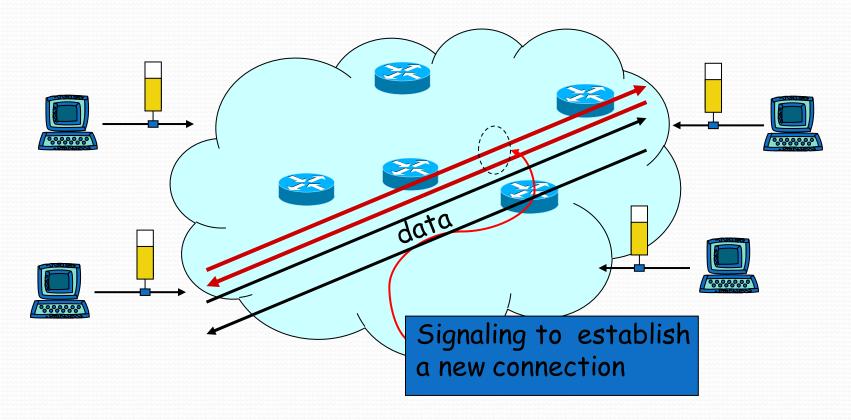   - # of packets sent
    - # of packet lost
    - jitter

- **Multimedia (MM) supporting protocols:**

QoS Architectures:

- **Quality of Service (QoS):** is the ability to provide different priority to different applications, users, or data flow. It is measured as jitter, end-to-end packet delay, packet losses, etc

- **IntServ (Integrated Services):** is an architecture that specifies the elements (**fine-grained**) to guarantee QoS on networks.
    - **Per-flow architecture**: that means that every **flow** (5-tuple form by source/destination IP@ + source/destination ports + transport protocol) is treated specifically at each route to guarantee it a QoS.
    - Uses **RSVP** as main protocol

- **DiffServ (Differentiated Services):** architecture that specifies a simple, scalable and **coarse-grained** mechanism for classifying, managing network traffic and providing QoS.
    - **Per-class architecture:** that means that flows are grouped in classes that received the same treatment in the network
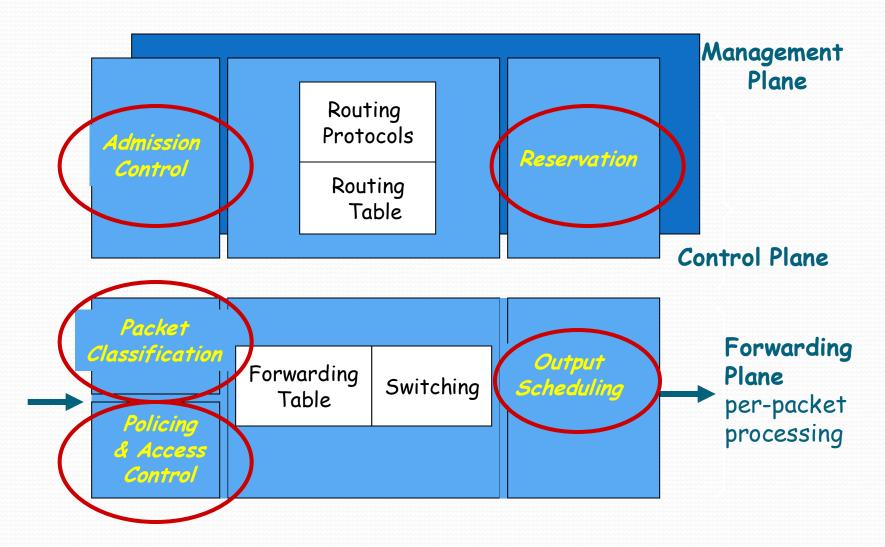
IntServ (Integrated Services):

Signaling to establish a new connection

data

• Signaling for Resource Reservation (RSVP)
• Admission Control of new connections (CAC)
• Policy function (UPC (Usage Parameter Control), ej: TBF: token bucket filter)
• Scheduling policies in the routers (ej: WFQ)

- **IntServ (Integrated Services):** An application wishing to obtain a certain QoS:

    - Uses the **RSVP signaling protocol** for reserving resources in the routers and partners (it supports multicast)

    - The routers will maintain a **reservation for a flow** (defined as a one-way stream of data with defined destination (IP@, Port, Transport protocol)

    - Routers must implement an **Admission Control** function for verifying that the resource needed are available.

    - In the new connection is accepted, Routers must thus **create and maintain** a certain **state** per each flow

    - When a router receives a packet, **classifies** it depending on the flow it belongs to and then processes it accordingly

IntServ (Integrated Services):

Management Plane

Admission Control

Routing Protocols

Routing Table

Reservation

Control Plane

Packet Classification

Forwarding Table

Switching

Output Scheduling

Policing & Access Control

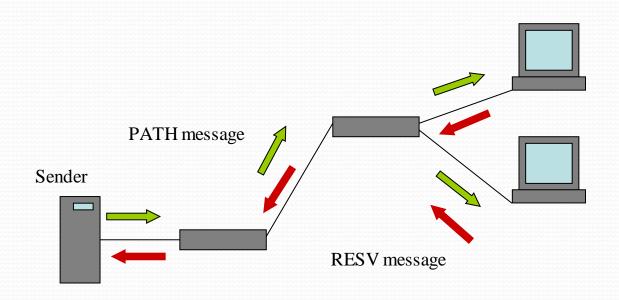**Forwarding Plane** per-packet processing

- ## Resource Reservation Protocol (RSVP)
  - It is a signaling protocol which allows the sender/receiver to reserve resources along a communication path
  - It is used by a host on behalf of an application data flow
  - It provides reservations for bandwidths in multicast trees
  - It is **receiver-oriented**, i.e. the receiver initiates and maintains the resource reservation used for that flow

  - Note that RSVP **does not specify** *how* nodes will provide the reservation it requests

- ## Resource Reservation Protocol (RSVP)

After receiving a PATH message, receivers can request a reservation
By sending back RESV messages. These messages:

- **PATH messages** specify the resource requirements
- **RESV messages** set-up the state of routers along the path

After receiving the RESV, sender can start to transmit packets

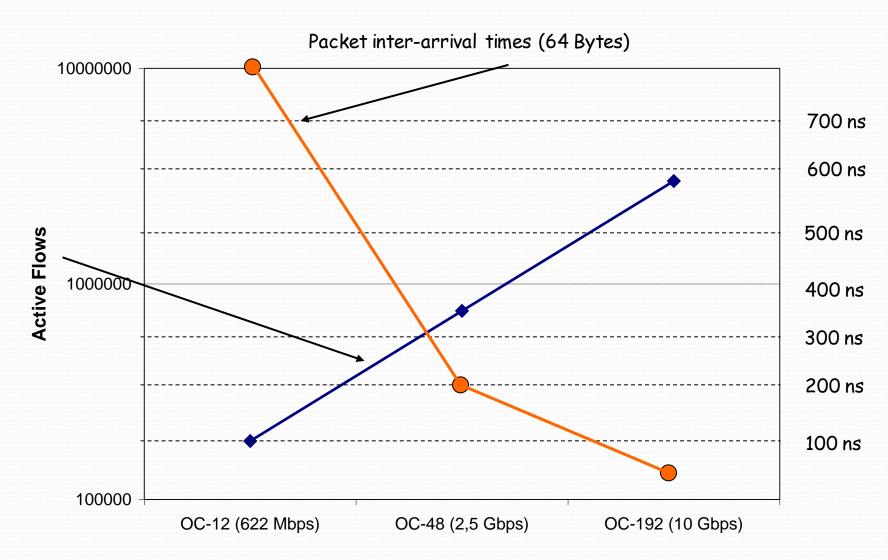PATH message

Sender

RESV message
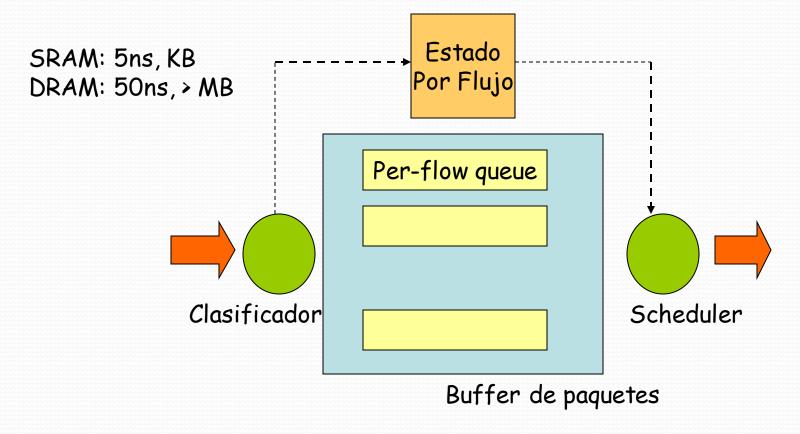
- **Main drawbacks with RSVP:**

  - In the "core network", is where routers should process x100.000 flows for each link, IntServices presents thus problems of scalability:

    - control plane (RSVP)
    - forwarding plane (per-flow scheduling)

Number of active flows:

Packet inter-arrival times (64 Bytes)

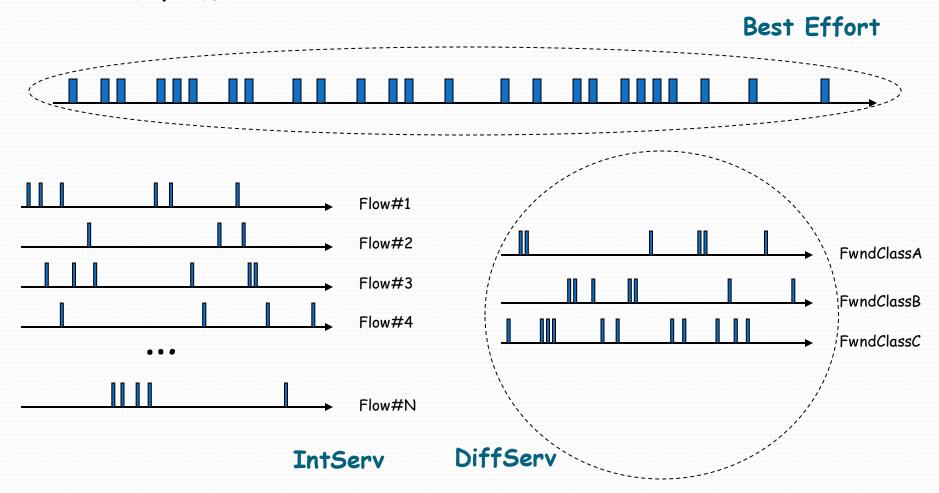|	OC-12 (622 Mbps)	OC-48 (2,5 Gbps)	OC-192 (10 Gbps)

- **Scalability problems in the forwarding plane**
  - Per-flow scheduling: for each new packet read/written the router has to access to the flow variable state (in real-time)

SRAM: 5ns, KB
DRAM: 50ns, > MB

Estado Por Flujo

Per-flow queue

Clasificador

Scheduler

Buffer de paquetes
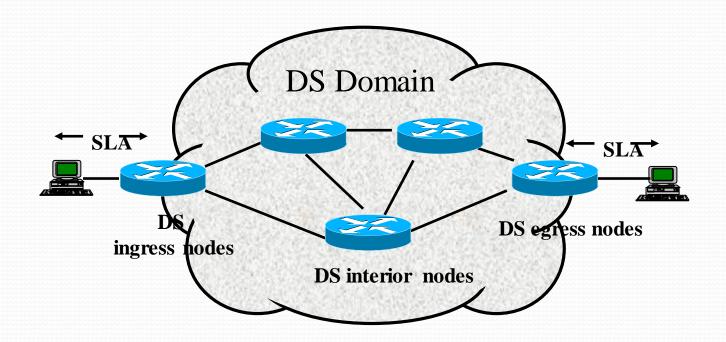
- For each flow, the *control plane* has to process several RSVP packets, that modify state tables, etc.

- If every T seconds, an active flow generates a RSVP message → the *Control plane* has to process LxN/T messages/second (L: number of links, N: number of flows per interface)

- Example:
  - N= 2.000.000, L = 16 T = 180 s → Every 6 μs the *control plane* has to process a RSVP packet …

- Resource reservation is performed as a function of traffic aggregation (**Forwarding classes**) instead than over flows

DiffServ (Differentiated Services):

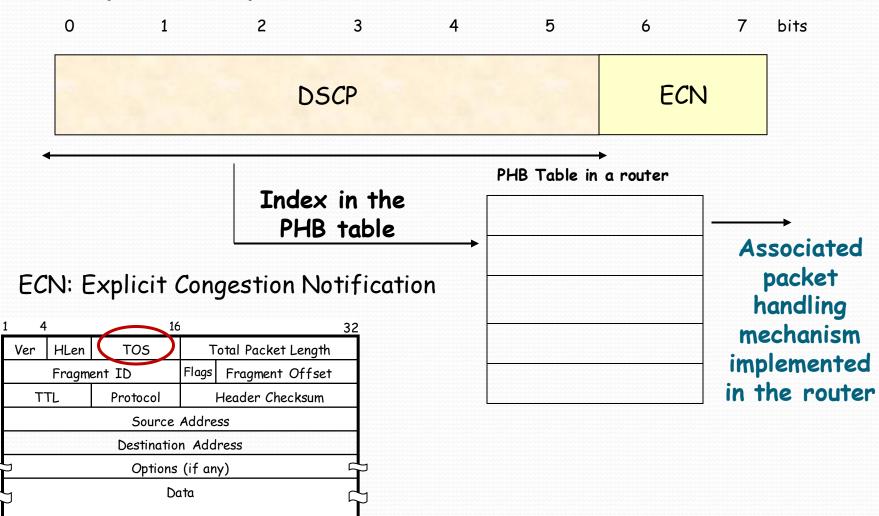- Classes of traffic are classified and marked in the IP header
- Each class of traffic receives a specific treatment in the domain (Autonomous System) defined in the SLA (Service Level Agreement)
- The treatment is given using schedulers, policing functions, etc

- **Core routers** give different ***treatment*** (**PHB: Per Hop Behavior**) to each packet depending of the byte **Type of Service** (ToS) in the IP header. The ToS associate the packet to a specific **Forwarding Class**

- **Access routers** has the responsibility of classifying and marking packets in one of the defined *Forwarding Classes*

  - → The most costly functions are implemented in the edge of the network, while in the ***core network*** the routers only have to **give different treatments** to the packets depending on a reduce number of classes
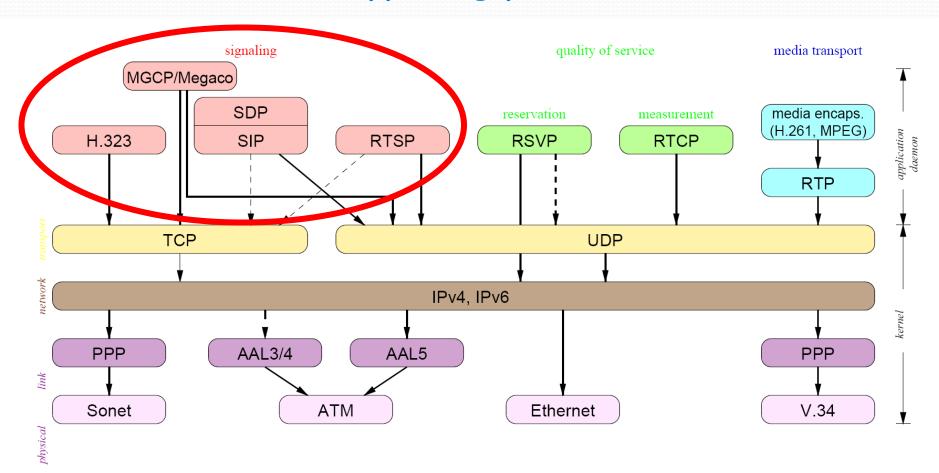
- **DS (DiffServ) field:** substitutes the TOS IPv4 field

0	1	2	3	4	5	6	7	bits

DSCP	ECN

Index in the PHB table

PHB Table in a router

ECN: Explicit Congestion Notification

Associated packet handling mechanism implemented in the router

1	4		16		32
Ver	HLen	TOS	Total Packet Length		
Fragment ID			Flags	Fragment Offset	
TTL		Protocol	Header Checksum		
Source Address					
Destination Address					
Options (if any)					
Data					

- **DiffServ allows defining bussines models, example:**
  - SLA defines a class called **"VoIP"**. The Service Provider (SP) and the client have the following agreement
    - An Ingress Committed Rate (ICR) is defined and an Egress Committed Rate (ECR). Normally, the traffic will be symmetric, so ICR=ECR. The SP guarantees a maximum delay per each direction (i.e, 15 or 30 ms), and a loss ratio (i.e., 0.1 %).
  - The SLA also defines a specific class called **"Business Latency-optimized"**, with specific parameters such as for example, a maximum delay of 30 or 80 ms
  - A third class, **"Business Throughput-optimized"**, does not define a specific guaranteed bandwidth. On the other side, It is agreed that this class will have a 80% of the bandwidth behind of the classes "VoIP" and "Buss-Lat". But there are no delay or loss ratio guarantees with this class.
  - A fourth class, **"standard"**, will receive the rest of the bandwidth (i.e., 20%) after removing "VoIP", "Buss-Lat" and "Bus-throughput"

Multimedia (MM) supporting protocols:

- **Signaling protocols for MM:**
    - before sending MM packets (i.e., encapsulating the MM data using RTP-UDP-IP) the client must communicate with the server and learn:
        - Where is the MM data ? What kind of coding (MPEG1, 2, 4, H261, …) technique is using ? What kind of Session (e.g. RTP) and transport (e.g. UDP) protocols is going to use ?
        - Similar format than HTTP

    - **RTSP (Real Time Streaming Protocol)** for video streaming

    - **SIP (Session Initiated Protocol)** for telephony (**VoIP**) over Internet, although also allows video streaming

- **RTSP (Real Time Streaming Protocol)**

  - *RTSP* is the control protocol for the delivery of multimedia content across *IP* networks.
  - It is based typically on *TCP* for reliable delivery and has a very similar operation and syntax to *HTTP*.
  - *RTSP* is used by the client application to communicate to the server information such as
    - the media file being requested,
    - the type of application the client is using,
    - the mechanism of delivery of the file (unicast or multicast, *UDP* or *TCP*), and
    - other important control information commands such as DESCRIBE, SETUP, and PLAY.
  - The actual multimedia content is not typically delivered over the *RTSP* connection(s), although it can be interleaved if required. *RTSP* is analogous to the remote control of the streaming protocols

- ## RTSP example

  ### C→ S

  - OPTIONS rtsp://video.fooco.com:554 RTSP/1.0 Cseq: 1

  the client will establish a TCP connection to port 554 on the server and issue an OPTIONS command showing the protocol version used for the session

  ### S → C

  - RTSP/1.0 200 OK Cseq: 1

  the server acknowledges the client request

- # RTSP example

  ## C → S

  - DESCRIBE rtsp://video.fooco.com:554/streams/example.rm RTSP/1.0 Cseq:2

  the client issues a DESCRIBE command that indicates to the server the URL of the media file being requested.

  ## S → C

  - RTSP/1.0 200 OK Cseq: 2 Content-Type: application/sdp Content-Length: 210 <SDP Data...>

  The server responds with another 200 OK acknowledgment and includes a full media description of the content, which is presented in either Session Description Protocol (SDP) or Multimedia and Hypermedia Experts Group (MHEG) format.

- **RTSP example**

  C→S

  - SETUP rtsp://video.fooco.com:554/streams/example.rm RTSP/1.0 Cseq: 3 Transport:

    rtp/udp;unicast;client_port=5067-5068

  the client issues a SETUP command that identifies to the server the transport mechanisms, in order of preference, the client wants to use. We won't list all of the available transport options here (the RFC obviously contains an exhaustive list), but we'll see the client request RTP over UDP on ports 5067 and 5068 for the data transport.

  S → C

  - RTSP/1.0 200 OK Cseq: 3 Session: 12345678 Transport: rtp/udp;client_port=5067-5068;server_port=6023-6024

  The server responds with confirmation of the RTP over UDP transport mechanism and the client-side ports and includes the unique Session ID and server port information.

- **RTSP example**

  C→S
    - PLAY   rtsp://video.fooco.com:554/streams/example.rm RTSP/1.0 Cseq: 4 Session: 12345678

  the client is now ready to commence the receipt of the data stream and issues a PLAY command. This simply contains the URL and Session ID previously provided by the server.

  S → C
    - RTSP/1.0 200 OK Cseq: 4

  The server acknowledges this PLAY command, and the RTP stream from the server to client will begin.

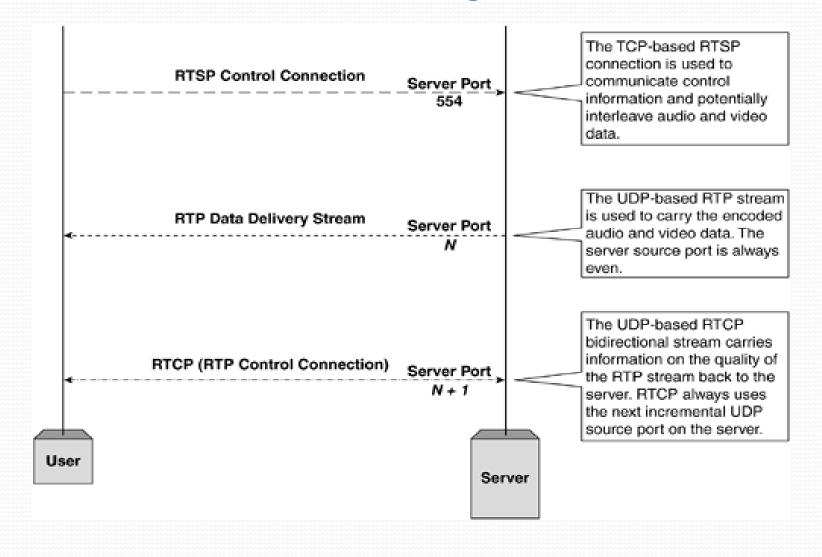 **<DATA>**

- **RTSP example**

C->S

- TEARDOWN rtsp://video.fooco.com:554/streams/example.rm   RTSP/1.0 Cseq: 5 Session: 12345678
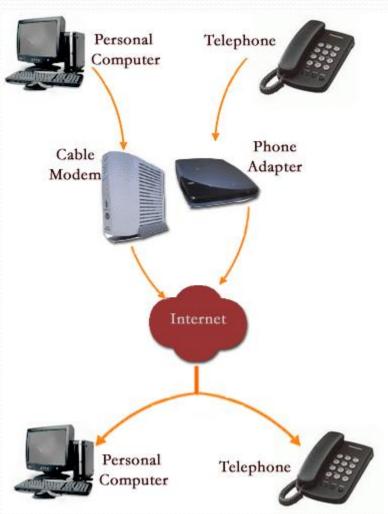
S->C

- RTSP/1.0 200 OK Cseq: 5

Once the client decides that the stream can be stopped, a TEARDOWN command is issued over the RTSP connection referenced only by the Session ID

RTSP (Real Time Streaming Protocol)

RTSP Control Connection — Server Port 554

The TCP-based RTSP connection is used to communicate control information and potentially interleave audio and video data.

RTP Data Delivery Stream — Server Port N

The UDP-based RTP stream is used to carry the encoded audio and video data. The server source port is always even.

RTCP (RTP Control Connection) — Server Port N + 1

The UDP-based RTCP bidirectional stream carries information on the quality of the RTP stream back to the server. RTCP always uses the next incremental UDP source port on the server.

User

Server

• VoIP (Voice iP)

VoIP is a technology that allows you to make telephone calls using a broadband Internet connection instead of a regular (or analog) phone line.

Some services using VoIP may only allow you to call other people using the same service, but others may allow you to call anyone who has a telephone number - including local, long distance, mobile, and international numbers.

Also, while some services only work over your computer or a special VoIP phone, other services allow you to use a traditional phone through an adaptor

- # VoIP (Voice iP)

  - ## H.323 (ITU-T), SIP (IETF)

    - Both describe terminal equipments and other entities that support multimedia communications over packet networks.
    - Possibility of joint operation with telephone networks.

  - ## MGCP, Megaco/H.248

    - Protocols that allow controling devices from a Media Gateway Controller (MGC) or Call Agent (CA).

SIP (Session Initiated Protocol)

- Is an application level signaling protocol for:
  - Setting-up
  - Modifying
  - Terminating
  real-time sessions between participants over an IP data network.

- SIP can support any type of single-media or multi-media session, including teleconferencing

- **Session Description Protocol (SDP):** defines session content using a set of type similar to MIME

- The actual multimedia content is exchanged using appropiate transport protocols (eg: RTP)

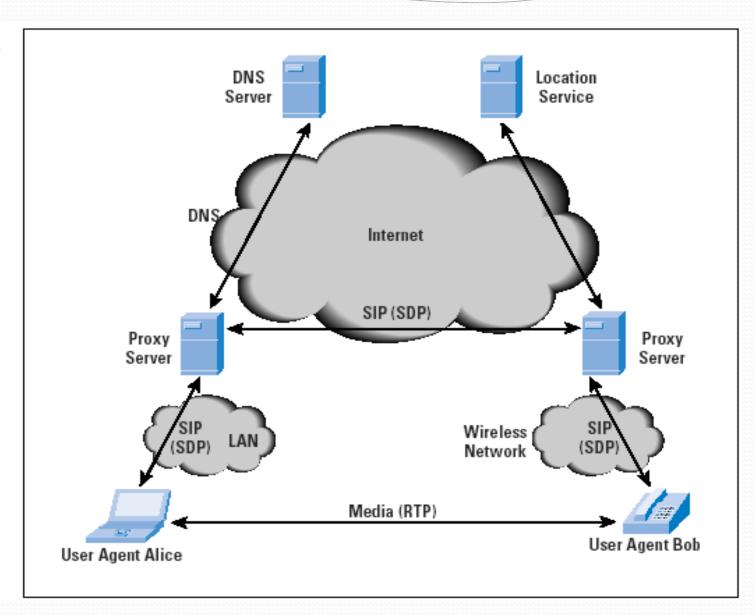- Key driving force: Internet telephony (Voice over IP, VoIP)

Figure 1: SIP Components and Protocols

Figure 2: SIP Successful Call Setup