

Ch4. 2 流媒体网络

4. 2. 1 概述

◆ 定义及特点

- ♣ 流媒体：按时间顺序**传输并播放的连续音视频数据流**
- ♣ 特点：只需部分下载，边下载缓冲边播放；连续性、实时性、时序性
 - ☞ Delay sensitive:收发端端时延<几百毫秒；
 - ☞ Loss tolerant:偶尔包丢失只在音视频回放中出现干扰
- ♣ 结构：**编码器**压缩编码时串流->**传输器**响应请求打包发送->**播放器**解码播放

◆ 流媒体网络传输的形式（C/S、IP组播、CDN、P2P）

- ♣ 流式**后播**（存储）音频/视频：点播：可重新定位，拖动，也可用直播方式播出
 - ☞ 向客户传输存储在服务器上的被压缩的音视频文件
 - ☞ 如演讲、音乐、录音、电影、录制节目、历史事件、卡通...
 - ☞ 特点：VOD
 - 存储媒体（预录）：用户可暂停、倒退，可接受响应时间1—10s
 - 流：接收几秒后开始播放已收、接收后续之循环流动
 - 连续播放：一旦开始，**依初始记录时序进行**，**端端时延要求很高**
- ♣ 流式**直播**音频/视频：用户不可拖动，未来数据不可获取
 - ☞ 网上的电视台/广播台，启动延时<几十秒
 - ☞ 不能实现快进，但存储后可后退、暂停等操作
- ♣ 实时**交互**音频/视频：
 - ☞ 如 IP电话、**视频会议**（任何时刻说话、移动）
 - ☞ 时延要求：语音<**150ms**不被觉察，**150-400ms可接受**，大于则不可理解

因特网的限制与需求

◆ 传输障碍

- ♣ Best effort service: **端端时延无保障**, 5—10s很普遍
- ♣ Packet jitter: **同流中包时延的变化**, **拥塞链路抖动大**

◆ 当前办法

- ♣ 用UDP发送, 避免TCP的**慢启动**阶段的**低吞吐量**
- ♣ 接收方**延迟**100ms (或加大缓冲区) 或更长后**重放**
- ♣ 发送方加**时戳**, 使接收方知道**何时播放**, 减轻**抖动的**影响
- ♣ 客户有额外带宽和存储时, 播放时同时**预取**数据
- ♣ 发送**冗余**包, 减轻网络丢包的影响... ..

◆ 如何演化

- ♣ **预留**派: 预留协议、路由器调度、流量描述、全局保障—>综合服务
- ♣ **自由**派: 更高带宽、更快交换、CDN边缘复制、多播覆盖网络...
- ♣ **中间**派: 边缘收费和监管、网络/传输层小变、划分2种流量—>区分服务

4.2.2 流式存储音频/视频2

◆ 需求

- ♣ 磁盘容量价格比迅速下降, 10^{12} — 10^{15} bytes可用
- ♣ 网络基础设施进步, 高速缓存CDN、ADSL、QOS协议
- ♣ VOD需求: 电视和WEB点播

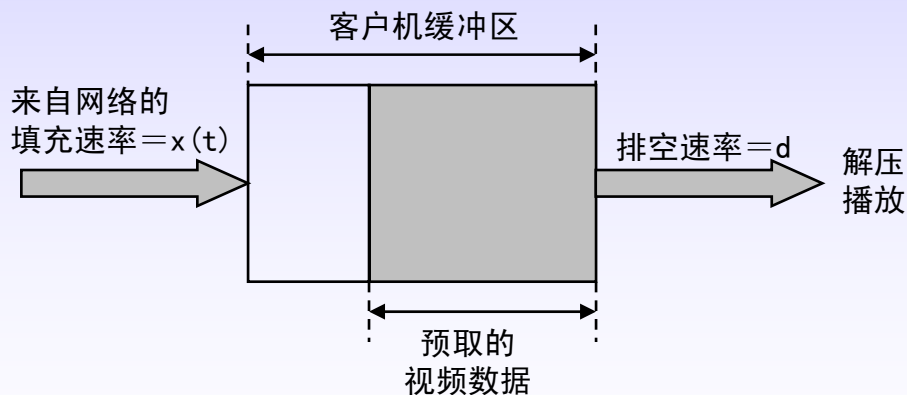
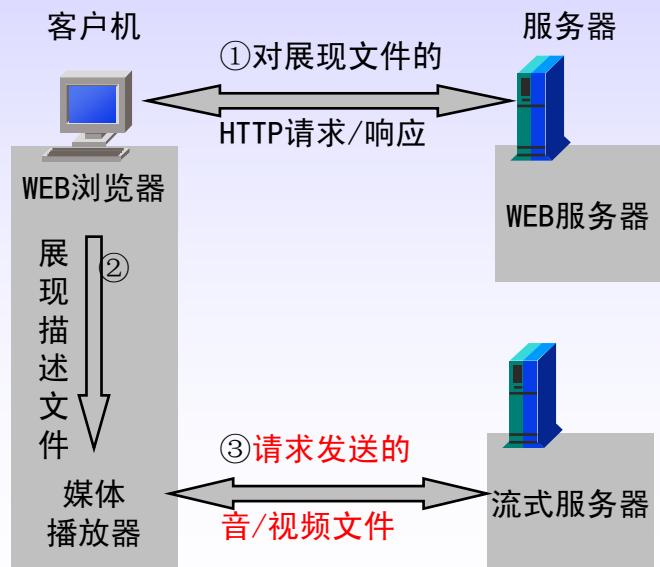
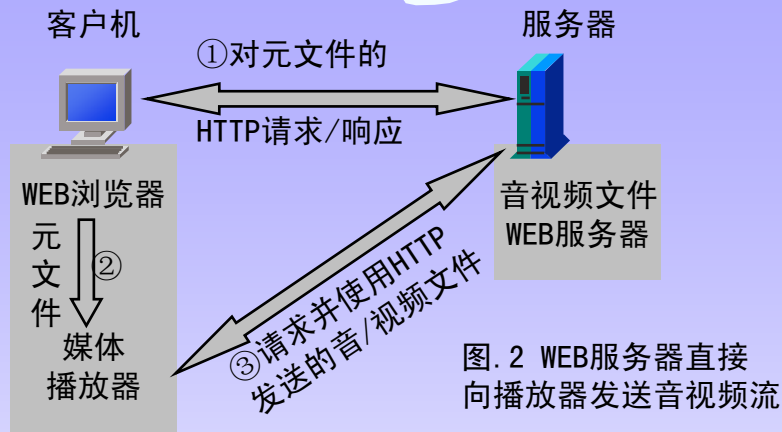
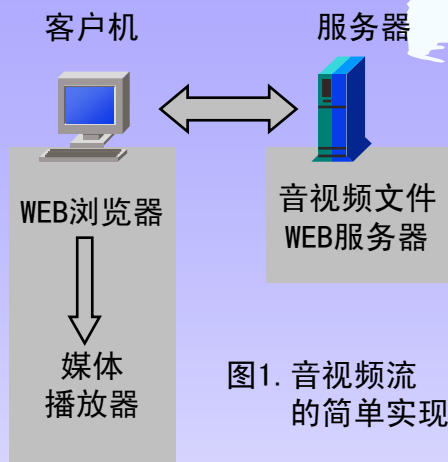
◆ 播放过程

- ♣ 客户机请求服务器
- ♣ 服务器把音视频文件发给TCP/UDP套接字
- ♣ TCP/UDP套接字把文件分段, 合适首部封装后转发给客户机

◆ 功能/性能要求

- ♣ 提供暂停/继续/文件中时间跳跃等交互功能
- ♣ 客户端媒体播放器功能
 - ☞ **解压缩、差错纠正**
 - ☞ **延时抖动消除**: 由冗余包构建丢失包、请求重传丢失包、内插丢失包

VOD 播放的结构



- ◆ 图1：简单，但浏览器成为瓶颈
- ◆ 图2：HTTP发送数据的效率低下
- ◆ 图3：③的三播放
 - ♣ UDP以排空速率恒定发送
 - ♣ 同上，但为消除抖动而延迟2—5秒再播放，且 $x(t) \leq d$
 - ♣ TCP发送，客户机尽可能多读并入缓冲区，且延迟2—5秒再播放。（TCP重传、流控可带来更好效果）

实时流协议RTSP

◆ RTSP: Real Time Streaming Protocol

- ♣ RFC 2326 中定义，在播放器和服务器间**交换播放控制信息**
- ♣ 功能类似**DVD遥控器**，**暂停/继续、重定位时间点、快退/快进**
- ♣ 是带外协议（out of band，类似FTP），端口号：**544**，
- ♣ 媒体流的包结构由**带内协议描述**，端口号不同

◆ RTSP不能做什么

- ♣ **不**定义音视频压缩方案
- ♣ **不**定义音视频在传输中怎样封装（这由RTP完成）
- ♣ **不**限制流媒体如何传输，可在UDP/TCP之上传输
- ♣ **不**限制播放器如何缓冲音视频，可同步、之后、之前播放

RTSP之例

◆ 展现描述文件

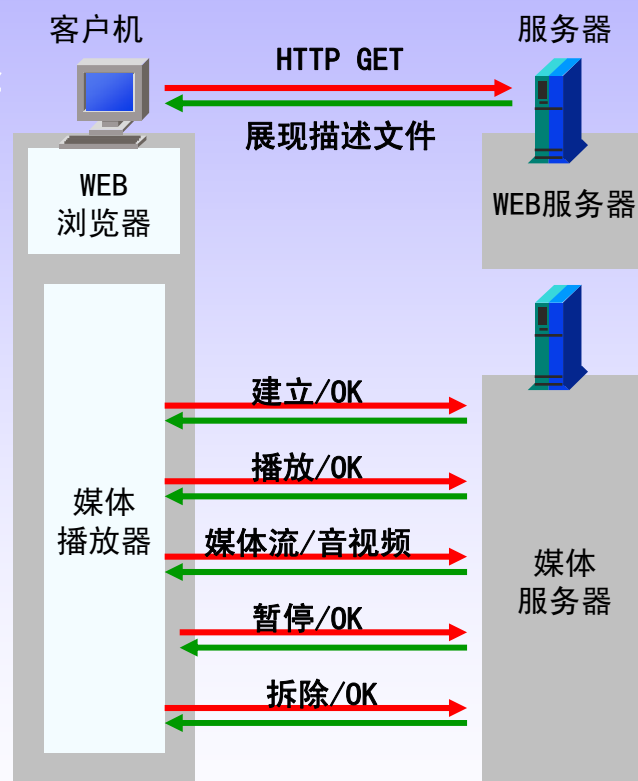
- ♣ 指向几个连续媒体文件的**引用或同步**指示
- ♣ 引用都以URL的rtsp://开始

◆ 下例音视频同步播放（作为一个组）

- ♣ 播放器能在两个音频记录（低保真）间选择

◆ 控制功能

```
<title>Twister</title>
<session>
  <group language=en lipsync>
    <switch>
      <track type=audio
        e="PCMU/8000/1"
        src="rtsp://audio.example.com/twister/audio.en/lofi">
      <track type=audio
        e="DVI4/16000/2" pt="90 DVI4/8000/1"
        src="rtsp://audio.example.com/twister/audio.en/hifi">
    </switch>
    <track type="video/jpeg"
      src="rtsp://video.example.com/twister/video">
  </group>
</session>
```



客户机与服务端之间的RTSP交互

- ◆ **WEB服务器**: 在一个HTTP响应报文中**封装**该展现描述文件，并送给浏览器
- ◆ **浏览器**: 根据该报文字段内容调用**播放器**
- ◆ **播放器和服务器**相互发送一系列报文(要求低保真服务)
- ◆ **简化对话**
 - ♣ **播放器**: 发**SETUP**请求
 - ♣ **服务器**: 回OK 响应
 - ♣ **播放器**: 发**PLAY**请求
 - ♣ **服务器**: 回 OK 来响应
 - ♣ **播放器**: 发**媒体流**请求
 - ♣ **服务器**: 音频送带内
 - ♣ **播放器**: 发**PAUSE**请求
 - ♣ **服务器**: 回OK 来响应
 - ♣ **播放器**: 发**TEARDOWN**求
 - ♣ **服务器**: 回 OK 来响应

```

C:SETUP:rtsp://audio.example.com/twister/audio RSTP/1.0
  Cseq:1
  Transport:rtp/udp;compression;port=3056;mode=PLAY
S:RTSP/1.0 200 OK
  Cseq:1
  Session:4231
C:PLAY:rtsp://audio.example.com/twister/audio RSTP/1.0
  Range:npt=0-
  Cseq:2
  Session:4231
S:RTSP/1.0 200 OK
  Cseq:2
  Session:4231
C:PAUSE:rtsp://audio.example.com/twister/audio RSTP/1.0
  Range:npt=37
  Cseq:3
  Session:4231
S:RTSP/1.0 200 OK
  Cseq:3
  Session:4231
C:TEARDOWN:rtsp://audio.../twister/audio RSTP/1.0
  Cseq:4
  Session:4231
S:RTSP/1.0 200 OK
  Cseq:4
  Session:4231

```

4.2.3 实时交互音频/视频

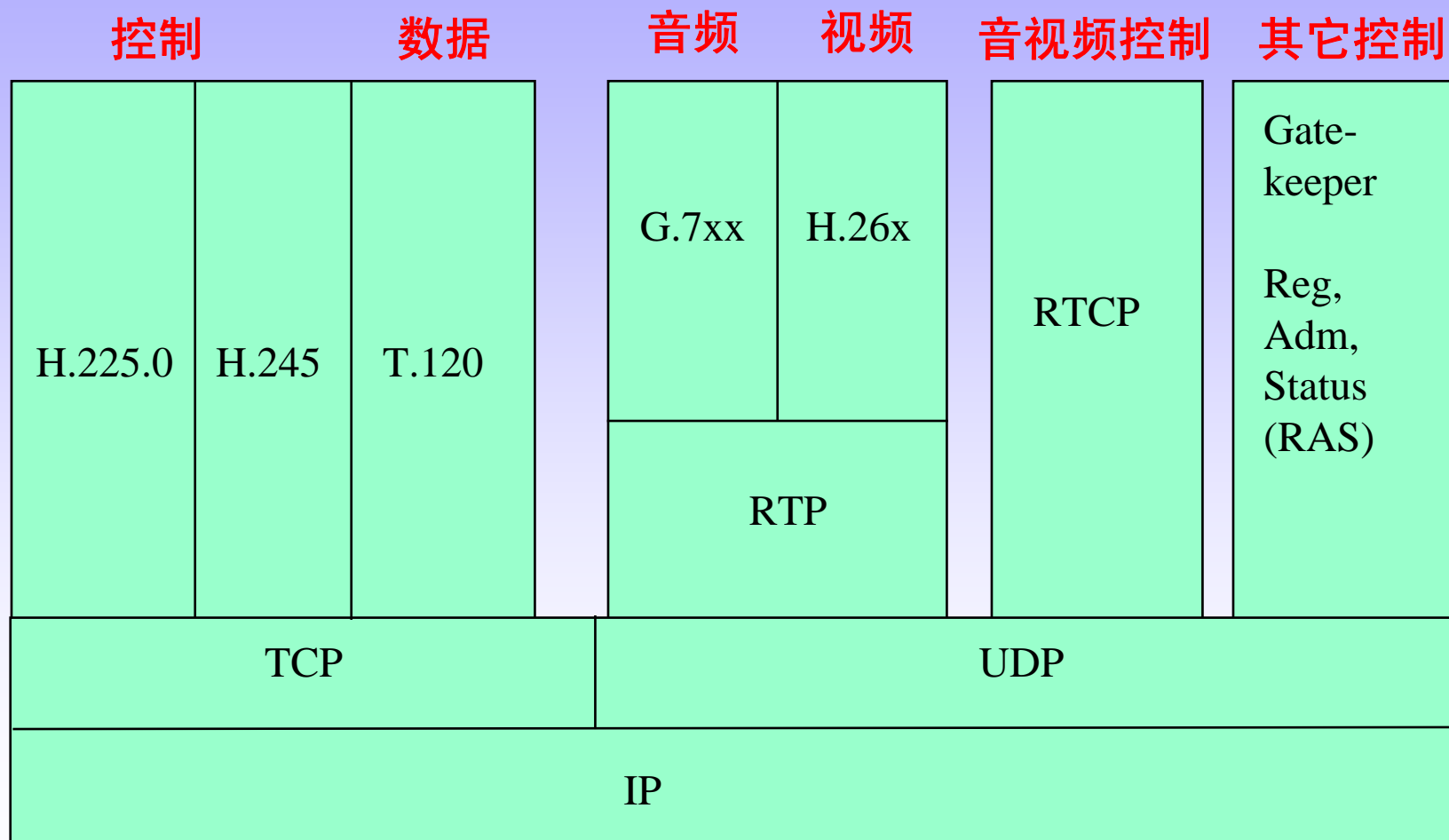
◆ 应用场景

- ♣ 简单的**广播音频会议**：广播地址，IP电话
- ♣ 复杂的**音频和视频广播会议**
- ♣ 混音器(Mixer)：多数据源**合成和编解码**转换
- ♣ 转换器(Translator)：实时数据的防火墙穿透和跨网**中继**
- ♣ 现场编辑

◆ IETF/ITU-T煞费苦心研制和提出标准

- ♣ IETF的**RTP**、**SIP**协议
- ♣ ITU-T的**H. 323**协议

相关协议栈



I) 实时传输协议RTP

- ◆ RTP: **Real-time Transport Protocol** ; RFC 3550
 - ♣ 给音视频数据分块传输之前**封装首部字段**
 - ♣ 包括音视频**数据、序号、时戳、其它有用**字段
 - ♣ 通用格式, 支持音频的**PCM/GSM/MP3**和视频的**MPEG/H. 263**
 - ♣ RTCP是其伴随协议
- ◆ RTP基础
 - ♣ 运行在**UDP之上** ;
 - ♣ 发送端**先用RTP包封装媒体块, 再由UDP封装**
 - ♣ 接收端解封并送播放器解码和显示
- ◆ 特点
 - ♣ 满足多媒体多方会议的需要, 提供**端对端实时数据传输**服务
 - ♣ 允许每个源分配一个独立的**RTP分组**: 如2参与者视频会议可打开4个RTP流, 传不同方向的音/视频分别2个。
 - ♣ 支持**单播和组播**
- ◆ **不能保障**
 - ♣ 数据的发送间隔
 - ♣ QoS

为什么用 RTP 代替 TCP ?

◆ 实时音视频多媒体应用要求

- ✦ 更需要**恒定数传率**,
- ✦ 不计较所有分组的**按序接收**

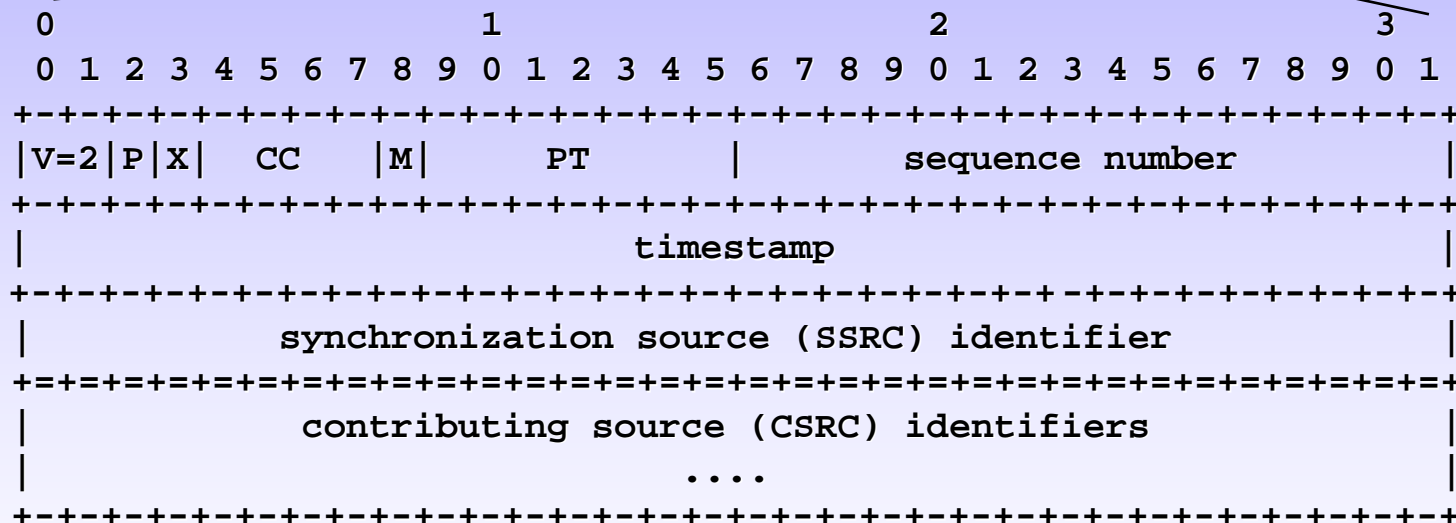
◆ TCP

- ✦ 在保障传输和传输顺序方面是很好的,
- ✦ 但TCP的**重传**协议对实时数据流将引起不可接受的**延迟**

◆ RTP设计关注

- ✦ 支持**恒定数据流**的应用
- ✦ 在连接质量上给应用以反馈, 允许应用改变连接条件以**适应变化**

RTP协议格式



V: Version

P: Padding

X: Header Extension

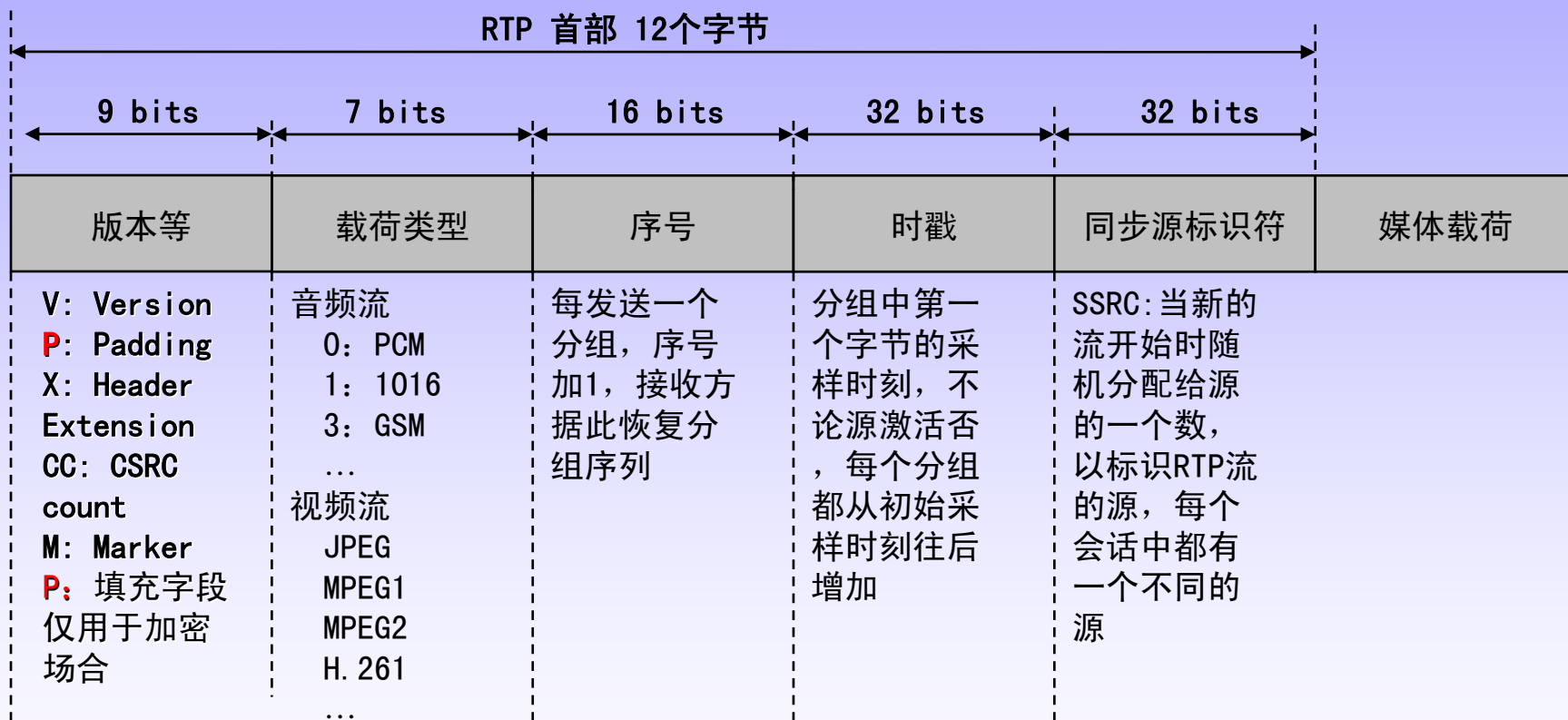
CC: CSRC count

M: Marker

PT: Payload Type

- ◆ **SSRC同步源标识**，可理解为信号的源头，如一个麦克风输入或一个摄像头输入，靠RTCP绑定。
- ◆ **CSRC供流源标识**，只有当MIXER插入时才产生，经MIXER输出的一个流通常由多个分流汇成，每个分流都有一个供流源。

RTP的首部



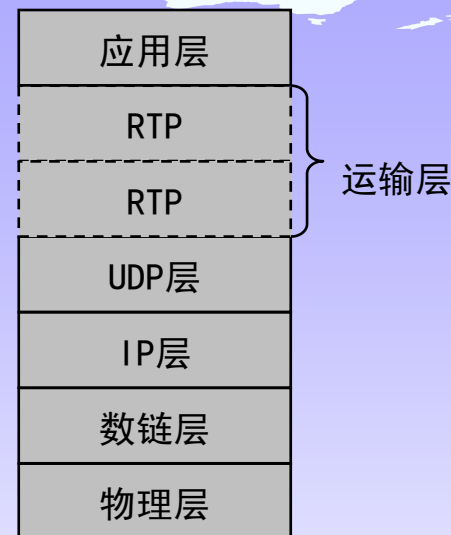
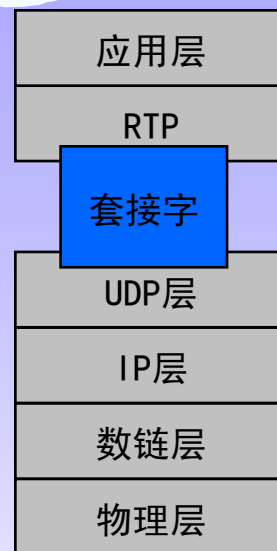
开发RTP的应用

◆ 两种类型

- ♣ 左图，RTP作为**应用程序一部分**，位于UDP之上；
- ♣ 右图，作为**运输层的一个子层**

◆ 两种开发方式

- ♣ 手工加入RTP：自行写出RTP封装、解封代码
- ♣ 利用现有RTP库
 - ♣ C语言中的库
 - ♣ JAVA类
- ♣ API需要给每个UDP报文设置IP地址+端口号



◆ RTP Profile（配置文档+负载格式文档）

- ♣ RTP是**通用实时传输协议**，特定应用需要Profile和Payload RFC来定义
- ♣ RFC 3551: RTP Profile for Audio and Video Conferences
- ♣ RFC 3016: RTP Payload Format for MPEG-4 Audio-Visual Streams

◆ Profile对RTP包的修改

- ♣ Marker, Payload Type, Payload Section, Header Extension（与数据编解码无关的应用），Padding（数据加解密）

II) 实时传输控制协议RTCP

◆ RTCP机制: Real-Time Transport Control Protocol

- ✱ 每隔一段时间, **用IP多播给会话中所有参与者发送一些控制包**

◆ **四大功能**

- ✱ 提供数据传输**状况的反馈**, 监控**RTP服务质量**和网络**拥塞程度**: 如包数量、丢失包、达到延迟抖动
- ✱ 跟踪RTP的**同步音/视频源**, RTCP包中携带着RTP的SSRC存放在CNAME (canonical name) 永久变量中。会话中SSRC可能因冲突或重启而新生, 故可追踪会话的每个参与者
- ✱ 收集每个RTP会话中参与者的状态, 每个参与者必须可以独立地知道会话参与者数量, 来决定RTCP发包的速率
- ✱ 可选, 传递最少的会话控制信息, 如参与者身份, 对加入或离开会话时非常有用, 用于IP多播环境

◆ 报文**种类**和组成

- ✱ SR (Sender Report): **发送报告, 当前活动发送者发送、接收统计。**
- ✱ RR (Receiver Report): **接收报告, 非活动发送者接收统计。**
- ✱ SDES (Source DEscription): **源描述项, 包括CNAME。**
- ✱ BYE/APP: **表示结束/应用特定函数。**
- ✱ 不加密时, 每个RTCP报文至少由一个SR/RR + 含CNAME的SDES组成

◆ 发送方机制

- ♣ 对每个正在发送的RTP流创建RTCP发送方报告SR
- ♣ 并发送到该会话参与者的**多播树上**
- ♣ 该报告能用于同步一次RTP会话中的不同媒体流，如视频会议的音视频流

◆ SR的内容

- ♣ RTP流的SSRC
- ♣ 时戳、该流中最近产生分组的墙上时钟 (wall clock time)
- ♣ 该流中发送的分组数
- ♣ 该流中发送的字节数

◆ 接收方机制

- ♣ 收到每个RTP流都会产生一个**接收报告RR**
- ♣ 将RR集中在单个RTCP包中，并发送到该会话参与者的多播树上

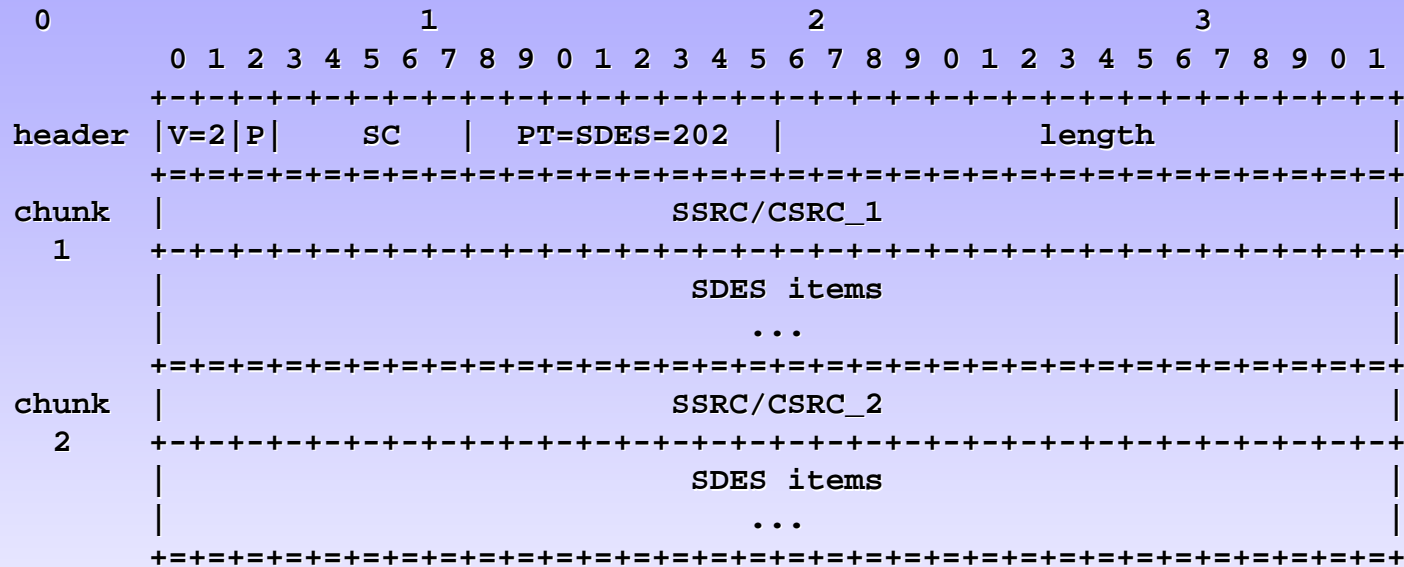
◆ RR的内容

- ♣ RTP流的SSRC
- ♣ 在RTP流中丢失的分组
- ♣ RTP流中收到的最后一个序号
- ♣ 达到延迟抖动，连续包到达时间间隔变化的平均预测

RC:
reception
report count

report
block
2

RTCP SDES报文格式

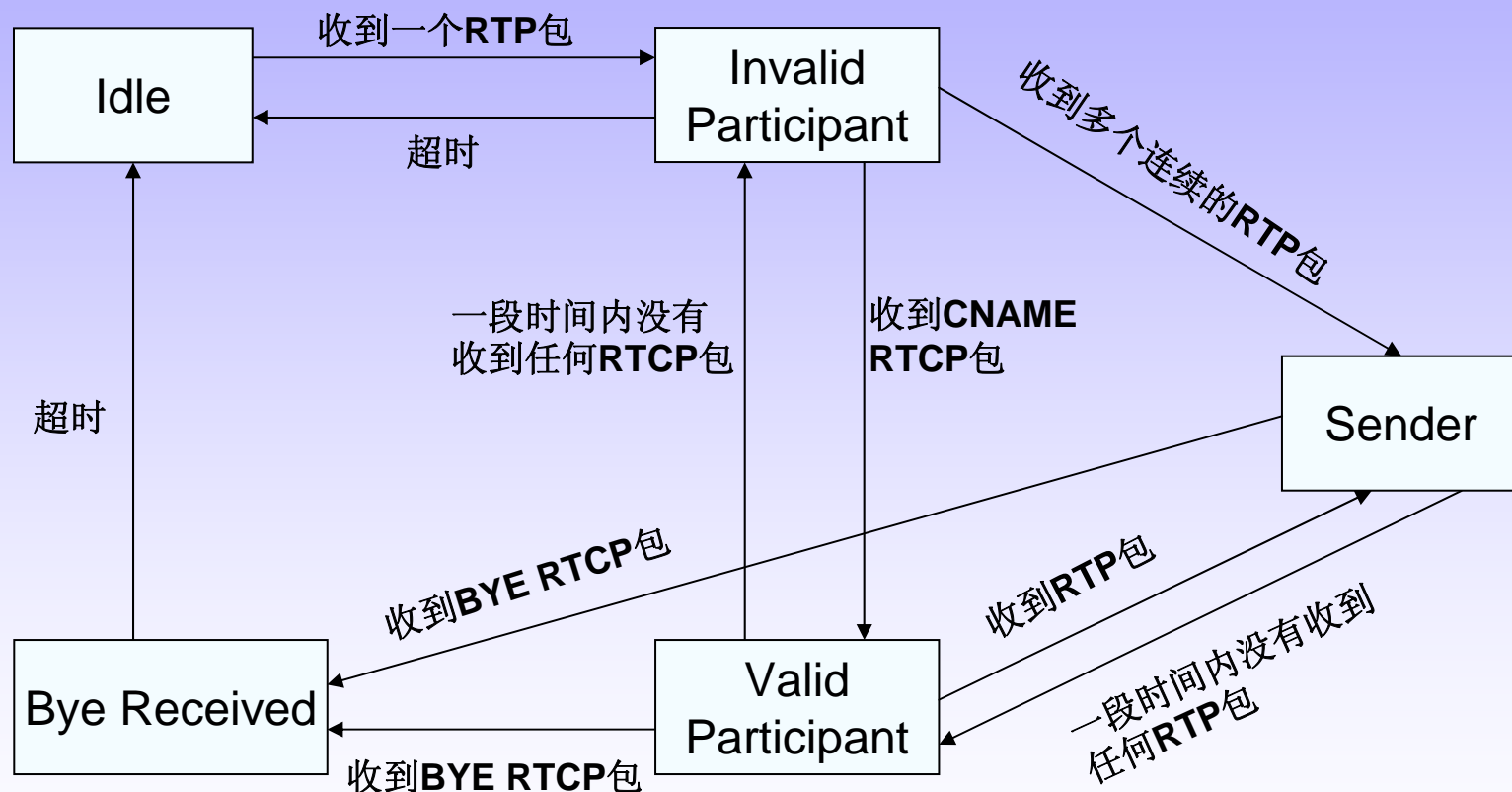


◆ SDES item的种类

- ♣ CNAME NAME EMAIL PHONE LOC
- ♣ TOOL NOTE PRIV



用RTCP维护会话参与者状态



RTCP发送周期的计算

◆ RTCP带宽的扩展问题

- ♣ 考虑1个发送方和 r 个接收方组成的RTP会话
- ♣ 每个接收方周期性产生RTCP分组，则其聚合带宽可能大大超过发送方发送RTP分组的速率，则导致观察到：
 - ☞ 发送到多播树的RTP流量不会随 r 的增加而增加
 - ☞ 但RTCP流量则随 r 的增加而线性增长

◆ 解决方法

- ♣ 因每个参与者要向其它参与者发送RTCP控制分组，可估计 r
- ♣ 使参与者向多播树发送RTCP分组的速率与 r 相关
- ♣ 整个RTCP可限制在会话带宽的5%以内（发送方2M则100Kbps）
- ♣ 100Kbps的25% 留给发送方自己，75% = 75Kbps 给接收方
- ♣ 所有接收方等量分配到 $75K/r$ bps RTCP流量
- ♣ 发送方/接收方的RTCP发送周期。（默认最大间隔5分钟）
 - ☞ $T_s = \text{发送方的数量} / (.25 * .005 * \text{会话带宽}) * (\text{平均RTCP包长})$
 - ☞ $T_r = \text{接收方的数量} r / (.75 * .005 * \text{会话带宽}) * (\text{平均RTCP包长})$

III) 会话发起协议SIP

◆ Session Initiation Protocol

- ♣ IETF定义，**基于IP的、纯文本、轻型应用层控制**（信令）协议
- ♣ 可以管理不同接入网络上的**会话**
- ♣ 参与会话成员可是**单播或多播**

◆ 主要功能

- ♣ **建立呼叫**、名字翻译与用户定位机制：把呼叫定位到任何被叫方
 - ☞ 每个人多个IP设备（电话、手机、PDA、PC），DHCP配置多个IP地址
- ♣ **特征协商**：使双方终端特征达成一致
 - ☞ 约定媒体编码
- ♣ **呼叫管理**：能引入其它用户加入呼叫或取消其连接
 - ☞ 引入新的参与者
 - ☞ 新增流媒体，改变编码
 - ☞ 呼叫过程中改变呼叫特征，呼叫转移或保持

呼叫一个已知IP地址的被叫

Alice

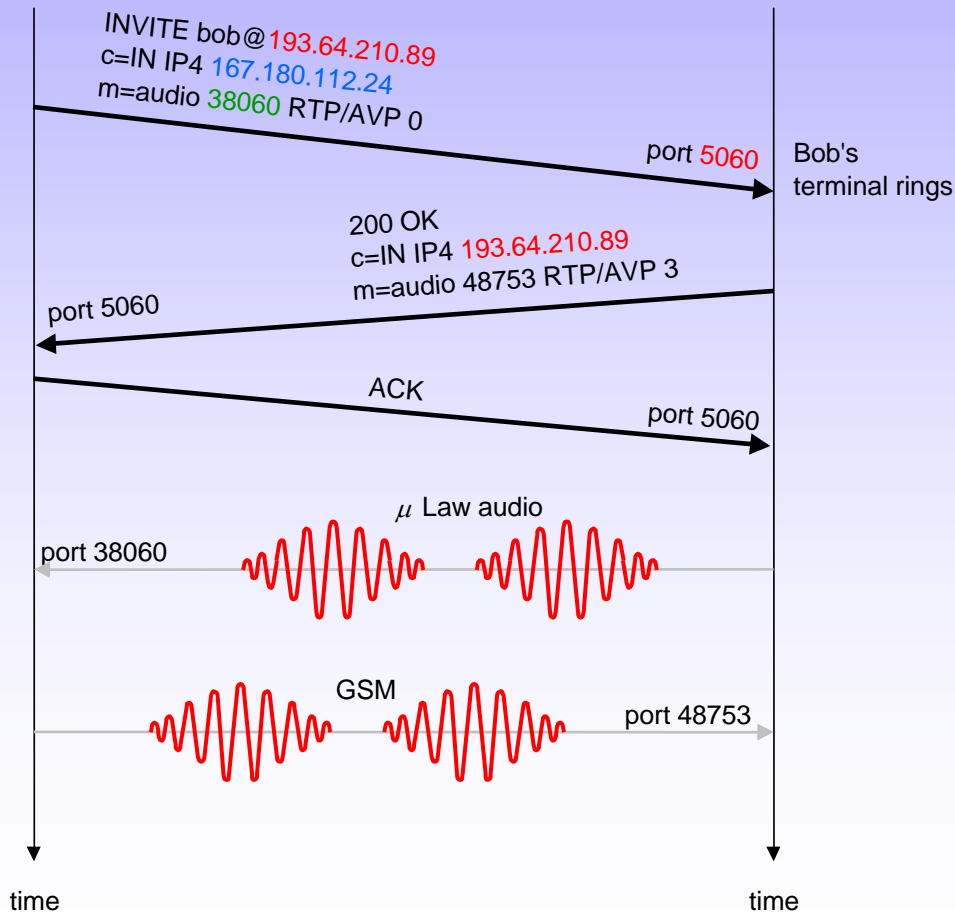


167.180.112.24

Bob



193.64.210.89



◆ Alice给Bob发起报文

- ♣ SIP周知端口号**5056**
- ♣ **Bob标识** bob@193.64.210.89
- ♣ Alice要接收的音频指示 PCM μ law, 在RTP中封装
- ♣ 将在**38060**端口接收RTP

◆ Bob 给Alice响应报文

- ♣ 也发到周知端口号5056
- ♣ 200 OK及Bob IP地址端口号等
- ♣ 希望接收的编码GSM

Setting Up a Call (More)

◆ Codec negotiation:

- ♣ Suppose Bob doesn't have PCM μ law encoder.
- ♣ Bob will instead reply with 606 Not Acceptable Reply and list encoders he can use.
- ♣ Alice can then send a new INVITE message, advertising an appropriate encoder.

◆ Rejecting the call:

- ♣ Bob can reject with replies "busy," "gone," "payment required," "forbidden".

◆ Media can be sent over RTP or some other protocol.

如果只知道邮件地址 ?

```
INVITE sip:bob@domain.com SIP/2.0
Via: SIP/2.0/UDP 167.180.112.24
From: sip:alice@hereway.com
To: sip:bob@domain.com
Call-ID: a2e3a@pigeon.hereway.com
Content-Type: application/sdp
Content-Length: 885

c=IN IP4 167.180.112.24
m=audio 38060 RTP/AVP 0
```

Notes:

- ◆ HTTP/SMTP like message syntax.
- ◆ sdp = session description protocol
- ◆ Call-ID is unique for every call.

- Here we don't know Bob's IP address.

Intermediate SIP servers will be necessary.

- Alice sends and receives SIP messages using the SIP default port number 5060.

- Alice specifies in Via: header that SIP client sends and receives SIP messages over UDP.

名字解析与用户定位

- ◆ Caller wants to call callee, but only has callee's name or e-mail address.
- ◆ Need to get IP address of callee's current host:
 - ♣ User can be mobile.
 - ♣ DHCP protocol may be dynamically assigning different addresses for each session.
 - ♣ User can have multiple IP devices (PC at home or work, PDA, etc.).
- ◆ Result can be based on:
 - ♣ Time of day (work, home).
 - ♣ Caller (don't want boss to call you at home).
 - ♣ Status of callee (calls sent to voicemail when callee is already talking to someone).
- ◆ Service provided by SIP servers:
 - ◆ SIP registrar server
 - ◆ SIP proxy server

SIP 注册器

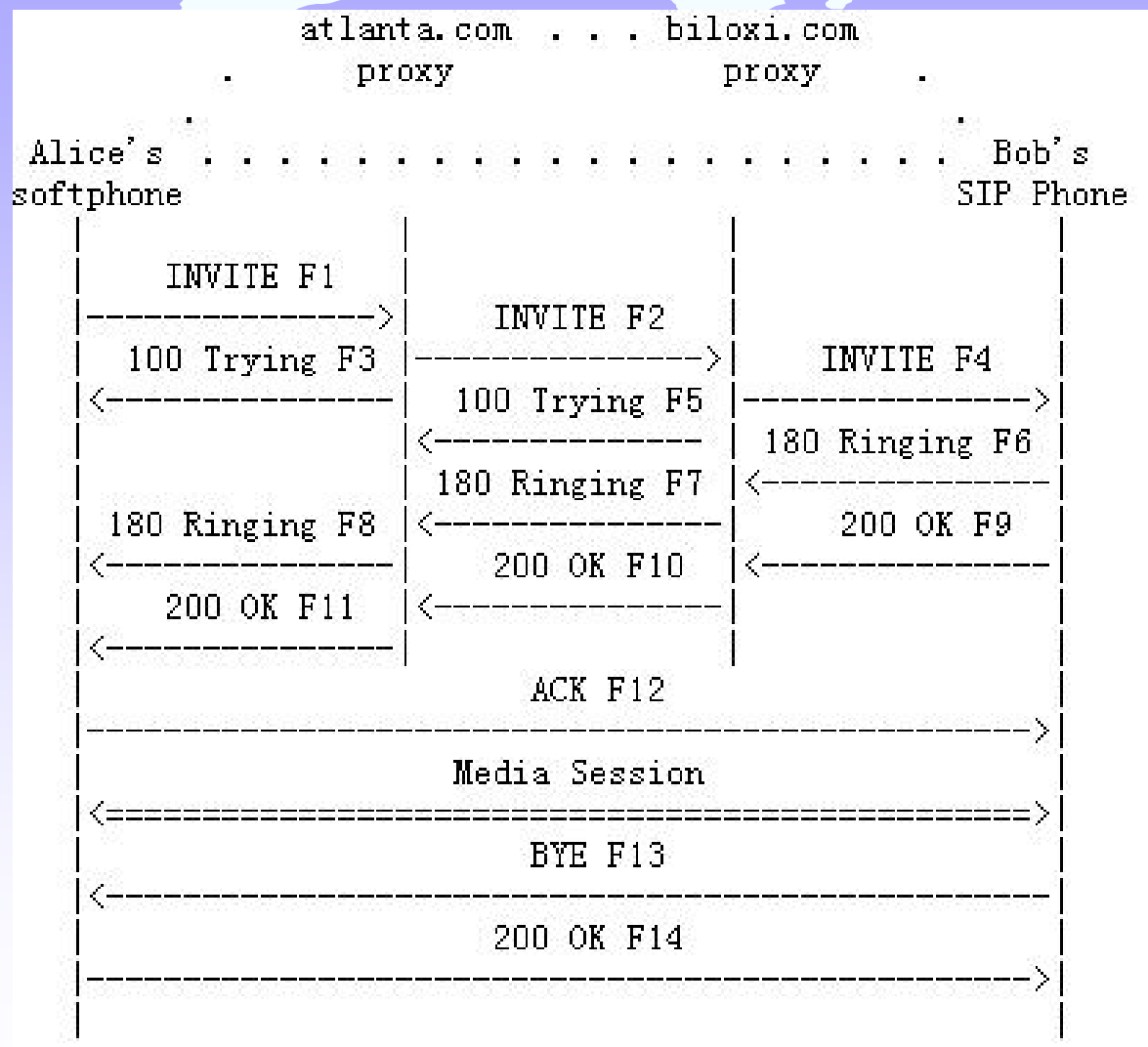
- ◆ When Bob starts a SIP client, the client sends a SIP REGISTER message to Bob's registrar server (similar function needed by Instant Messaging).
- ◆ The SIP registrar is very much like a DNS authoritative name server (translates fixed human identifiers to potentially dynamic IP addresses).
- ◆ 每个用户都有Register Message:
- ◆ 通常与SIP代理运行在同一台主机上
- ◆ 当Bob在其PDA发起SIP时，该应用向注册器发如下报文

```
REGISTER sip:domain.com SIP/2.0; Bob的注册器域名地址
Via: SIP/2.0/UDP 193.64.210.89 ; Bob当前最新IP地址
From: sip:bob@domain.com
To: sip:bob@domain.com
Expires: 3600; 每隔3600秒发来刷新报文，以维护注册器中的IP地址
```



SIP Proxy

- ◆ Alice sends INVITE message to **her** proxy server.
 - ♣ This message contains the address `sip:bob@domain.com`.
- ◆ This proxy is somehow responsible for routing SIP messages to the callee. (ex: Bob's proxy)
 - ♣ Possibly through multiple proxies.
- ◆ The callee sends its response back through the same set of proxies.
- ◆ Proxy returns SIP response message to Alice
 - ♣ This response contains Bob's IP address.
- ◆ Note: **proxy** is analogous to **local DNS server**.



SIP session setup example with SIP trapezoid

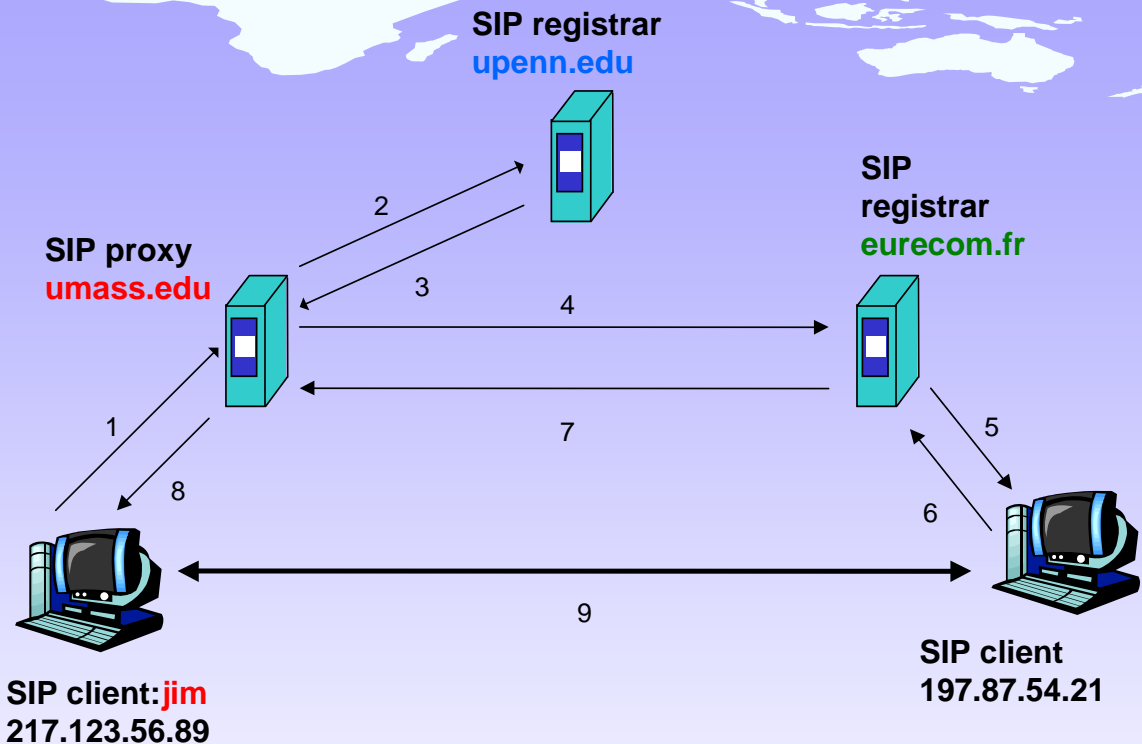
SIP 解析重定位

Caller jim@umass.edu
with places a
call to keith@upenn.edu

(1) Jim sends INVITE message to **umass** SIP proxy.
(2) Proxy forwards request to **upenn** registrar server.
(3) **upenn** server returns redirect response, indicating that it should try keith@eurecom.fr (重定向)

(4) **umass** proxy sends INVITE to **eurecom** registrar. (5) **eurecom** registrar forwards INVITE to 197.87.54.21, which is running keith's SIP client. (6-8) SIP response sent back (9) media sent directly between clients.

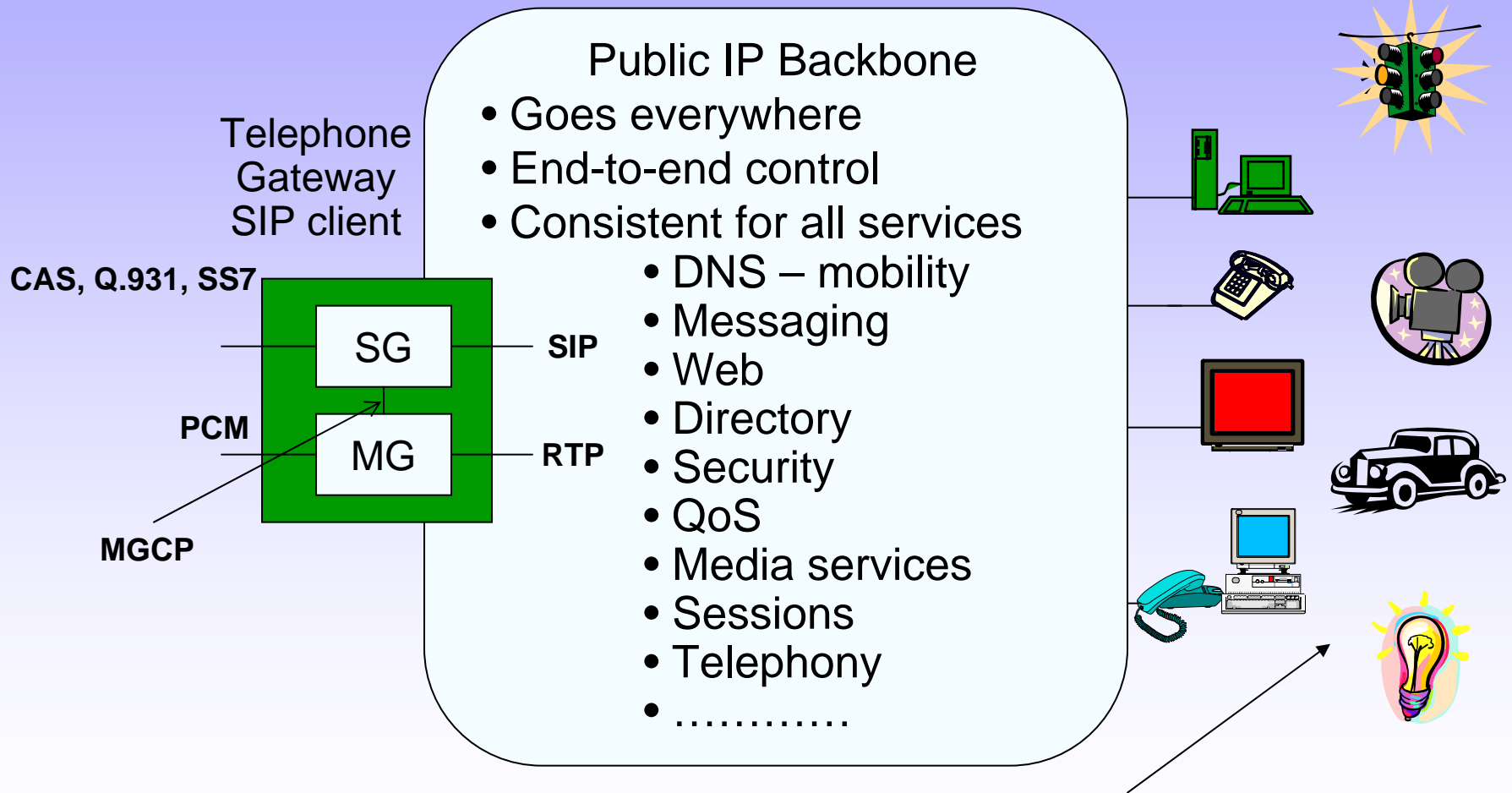
Note: There is also a SIP ACK message, which is not shown.



Telephony on the Internet

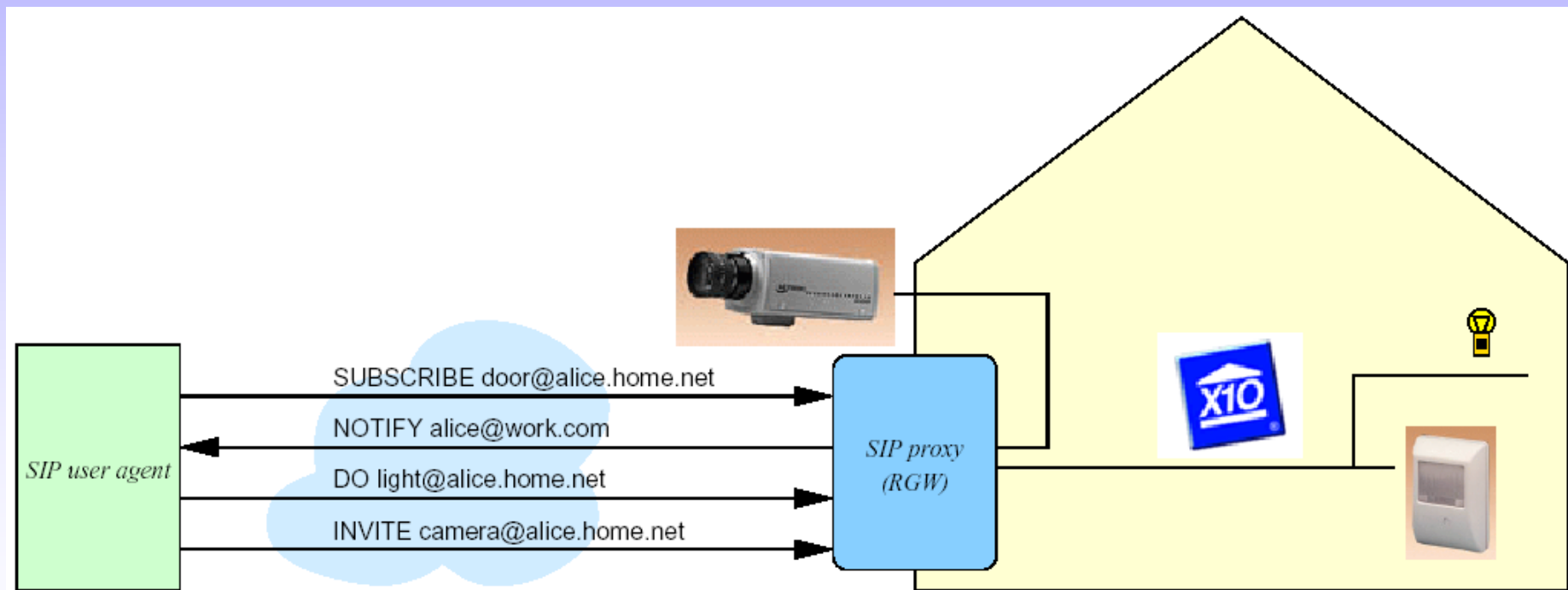
may not be a stand-alone business, but part of IP services

SIP/RTP Media Architecture



Any other sessions

SIP for appliances



IV) H. 323协议

◆ H. 323

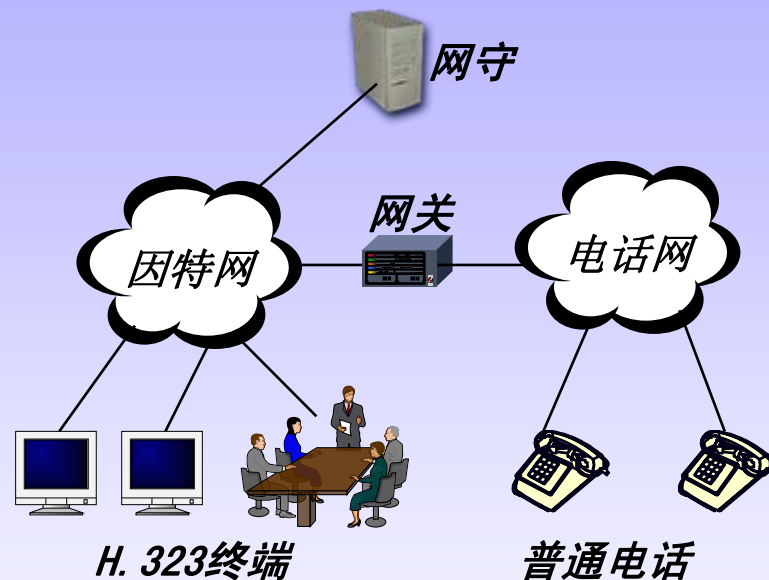
- ♣ 因特网上端系统之间音视频会议的通用标准(SIP的替代)
- ♣ 可和现有电话网技术结合在一起

◆ H. 323 的限制

- ♣ 需要足够的带宽
- ♣ 受网络性能的影响

◆ H. 323 标准包括

- ♣ H. 225.0: 呼叫信号和数据包封装；网守注册、允许/状态功能(RAS)
- ♣ H. 245: 包交换网络音/视频通信的通信控制
- ♣ RTP: 实时传输协议，在H. 323中传输音视频信号
- ♣ RTCP: 实时传输控制协议



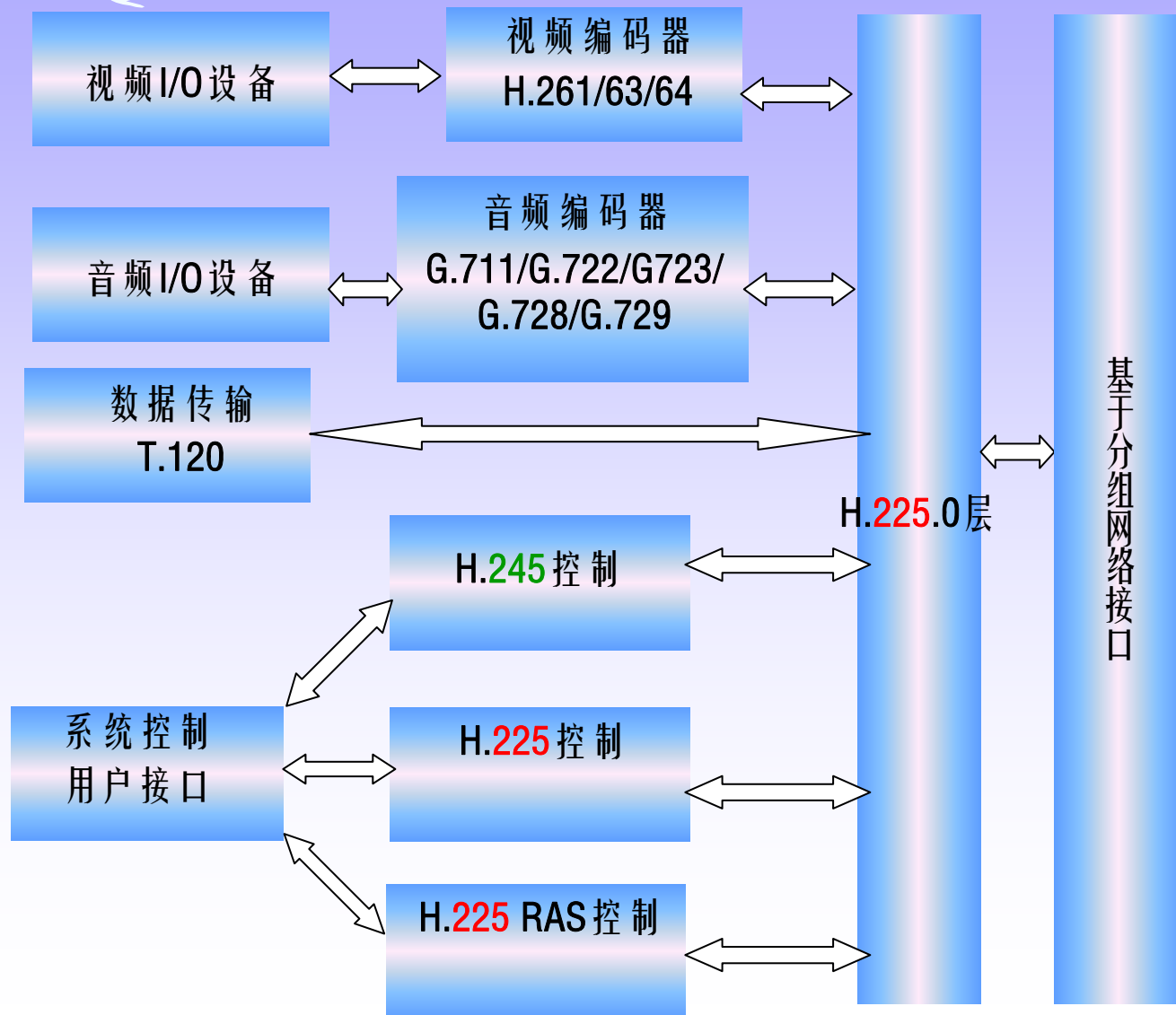
◆ H. 323有伞状规范:包括

- ♣ 端点间协商共同的音/视频编码规范。H. 323支持各种编码
- ♣ 协商音视频块如何封装/如何在网络上发送规范。H. 323强制用RTP
- ♣ 端点如何同其相关的网守通信的规范
- ♣ IP电话如何通过网关与PSTN电话通信的规范
 - ☞ H. 323端点必须支持G. 711语音压缩标准, 它用PCM产生56/64Kbps
 - ☞ 如支持视频(可选), 最低应支持QCIF H. 261 (176*144像素) 视频标准
- ♣ 强制采用H. 245控制协议、 Q. 931信令信道和通过网守注册的RAS协议

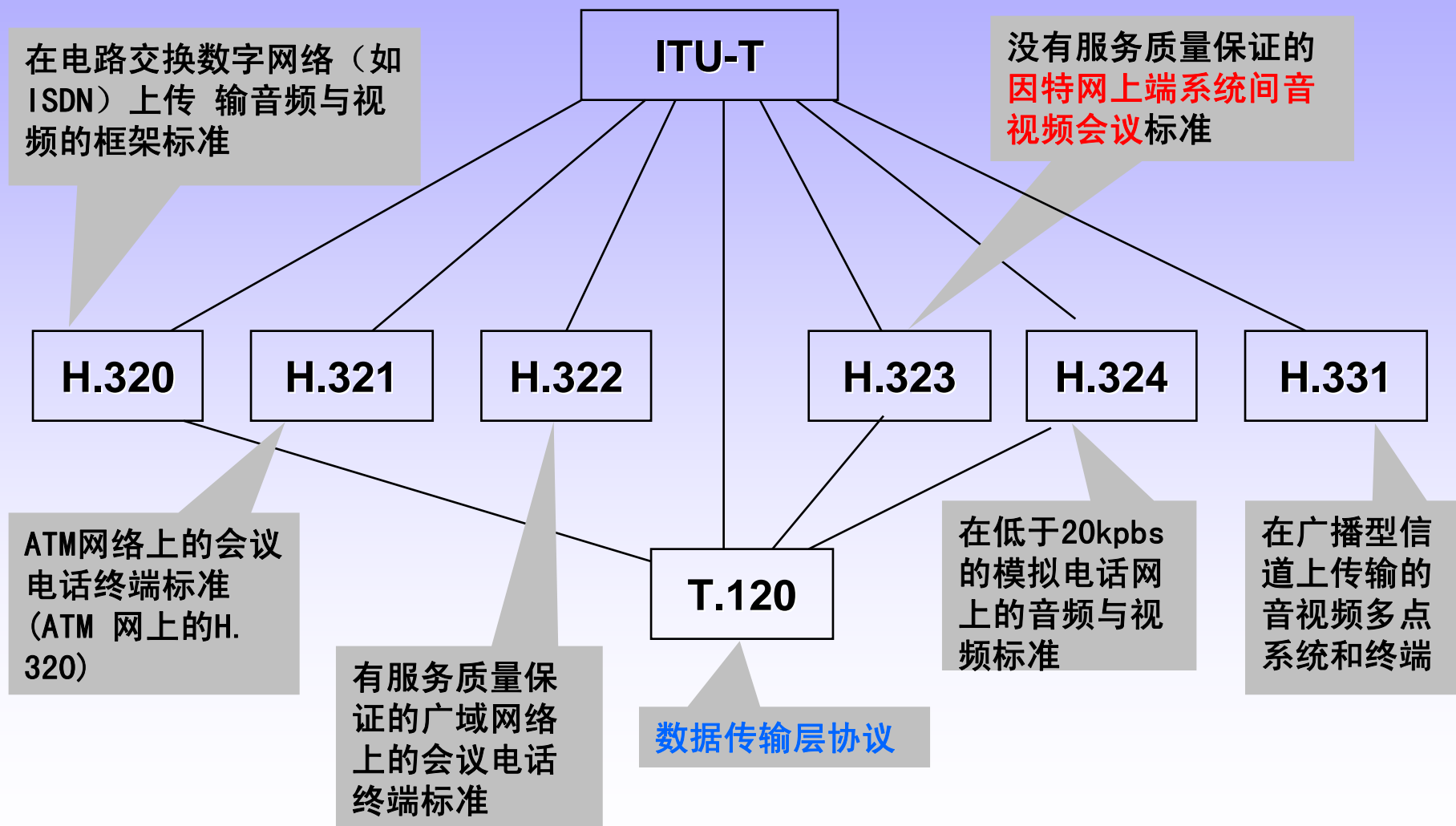
◆ H. 323与SIP的区别

协议区别点	H. 323	SIP
来源	ITU-T; 借鉴电话	IETF; 借鉴WEB/DNS/Email的许多概念
复杂度	完整/垂直集成的多媒体会议协议族: 信令、注册、准入控制、传输和编解码。强制使用RTP/G. 711/ QCIFH. 261。 伞状太复杂	只处理会话发起和管理的单一组件, 使用RTP/G. 711/ QCIFH. 261但不强制. 可与其它协议和服务结合。 KISS原则, 保持简单、傻瓜
编码规则	ANS. 1和压缩编码规则的二进制编码	基于文本的协议的编码
系统结构	多点控制单元MCU集中执行会议的控制功能, 所有参加会议终端都向它控制信息, 不支持多播	分布式呼叫模型, 支持多播
扩展业务	定义专门协议补充: H. 450/450. 2/450. 3等	没有定义专门补充协议, 但支持补充或智能业务
呼叫过程	呼叫通过RAS信令、呼叫信令和H. 245控制三信道; 呼叫 建立时间很长	会话请求过程和媒体协商过程等在一起进行, 只需 1.5个回路 时延来建立呼叫

H.323协议栈结构



与H.323相关的ITU-T标准



V) P2P IP电话: Skype

Skype 使用“快速追蹤”(FastTrack)第三代混合 P2P技术即“全球索引”(GI - Global Index) 技術，建立超级结点重叠网络构成全球分布式用户数据库，结点对接交换资料。不使用服务器等中央控制设备不仅减少搜尋的时间，还可以降低成本。由于这种P2P网络使用了终端本身电脑的处理能力，整个网络的处理能力随着终端数目增加而增加。

Skype自2003年8月推出以来已经使用15种语音在165个国家发展了600万用户。2004年7月提供互联网到传统电话之间的通话廉价商业服务。Skype还有 PDA的软件版本。正在发展移动电话和WiFi 双模手机，用它可以选择用移动电话通话或通过WiFi用Skype通话。

•特性

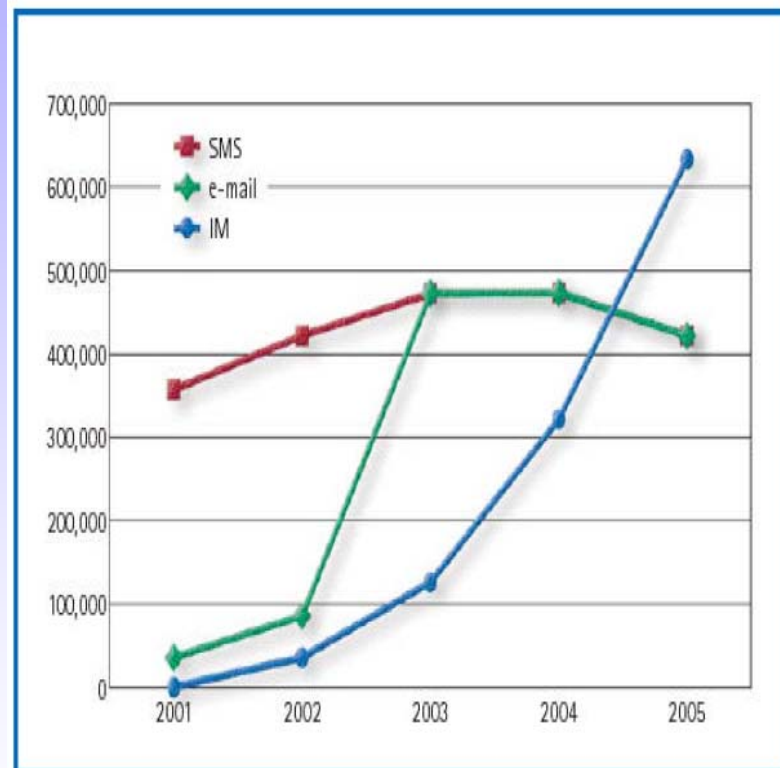
- 在全球范围内与其他 Skype 用户不受限地免费通电话
- 音质优良 — 比普通电话好
- 可以与所有防火墙、NAT 和路由器一起使用 — 无需进行任何配置!
- Skype 用户在线并且准备通话或聊天时，显示朋友列表
- 通话采用“端到端”加密，极具保密性

统一的P2P即时通信系统 IM

设计特征

- ◆ 对等连接模式P2P
- ◆ 自组织网ad-hoc)
- ◆ 消息格式使用 XML
- ◆ 有效的报到管理
- ◆ 异步、并行、可靠和近似实时通信
- ◆ 支持移动报到管理和移动即时通信
- ◆ 使用IETF 标准保证互通互用，如报到和SIP
- ◆ 安全和认证保证

Data source: Cohners In-Stat Group; Wireless Messaging Fusion. August 2001.



IP电话的尽力服务

◆ 发送方

- ✱ 在一个话音突峰期8000 Byte/s产生字节，汇聚1块/20ms;故1块的字节数= $20\text{ms} \times 8000 \text{ Bytes} / 1000\text{ms} = 160 \text{ Bytes}$;
- ✱ 附加特殊首部后UDP封装，经呼叫达到套接字接口，发送1UDP报文/20ms

◆ 接受方

- ✱ 何时播放1个块?
- ✱ 怎样处理丢失的块?

◆ 尽力而为的限制效果

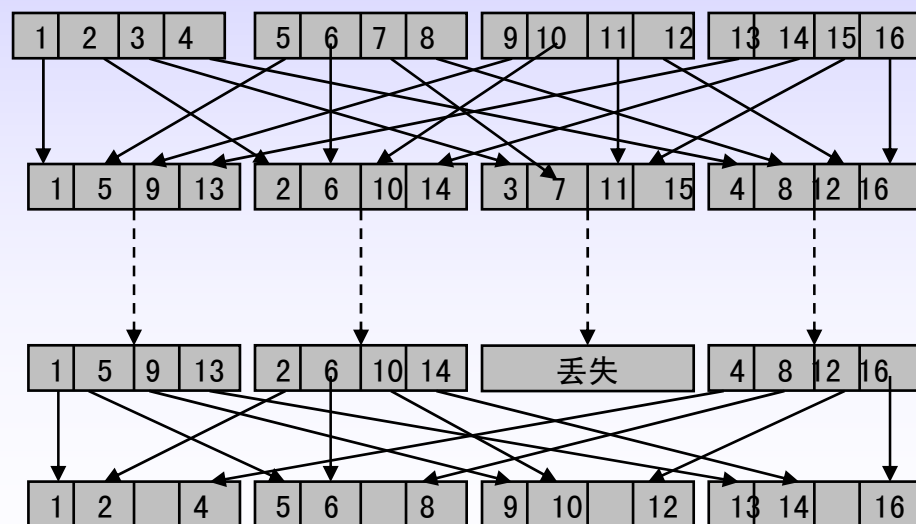
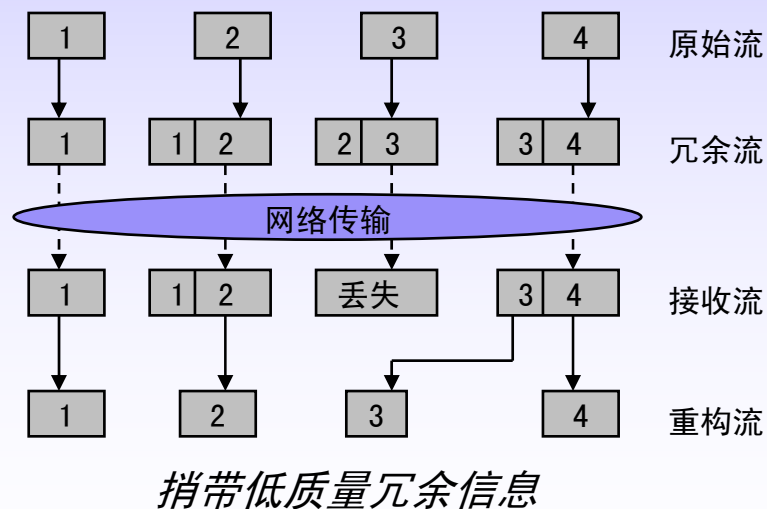
- ✱ 分组丢失（1-20%是可忍受的）
- ✱ 端到端延迟（ $\leq 150\text{ms}$ 觉察不到， $\leq 400\text{ms}$ 可接受）
- ✱ 时延抖动：路由器中的随机排队是时延主部

◆ 接受方消除抖动

- ♣ 每块编定一个序号
- ♣ 每个块编定一个时戳
- ♣ 接受方延迟播放

◆ 丢失分组的恢复

- ♣ ARQ (Automatic Repeat reQuest) 自动重传请求：传输量无法预测
- ♣ FEC (Forward Error Correction) 前向纠错
 - ☞ 每发送n个块后发1冗余编码块（第n+1块=前n个块的异或）；仅丢失1块后可重建
 - ☞ 发送一个分辨率较低的音频流作为冗余（标称64kbpsPCM, 冗余13kbpsGSM）
- ♣ 交错发送：单个分组丢失只导致重建流中的多个小间隙，减轻分组影响
 - ☞ 开销低、不增加带宽；但增加了时延；**适合流式存储音频**



4.2.4 流式直播音频/视频

◆ 流式直播技术（内容分发网CDN）

- ♣ IP单播：如HTTP/FTP，随着Clients增多，服务器端拥塞
- ♣ IP广播：效率很高，盲目性大，造成“广播风暴”
- ♣ IP多播：把包传输到共享同一地址的多播组，无数量和位置限制
- ♣ P2P流媒体分发技术
 - ☞ 基于树状拓扑协议及扩展（Tree-based Protocol and Extensions）节点组成一颗树，父节点为子节点传送数据，用于多播树构造和平衡
 - ☞ 基于Gossip 的协议(Gossip-based Protocol)：节点随机给系统中部分节点发送报文，每个接收报文节点继续这样做，重复该过程，直到报文传输到所有节点。

◆ 流式内容分发方式点评

- ♣ C/S模式：
 - ☞ Server方聚合高带宽、持续时间长、资源随Client增长而线性消耗
 - ☞ CDN通过边界代理Server复制，成本昂贵
 - ☞ IP多播因自身技术限制，很难实现可靠多播和拥塞控制，未广泛应用
- ♣ P2P模式：
 - ☞ 分散带宽、随Peers增加性能反而更好，可扩展性好
 - ☞ 代表新的方向，实用性也好
 - ☞ 分单源/多源模式：前者是应用层多播树。后者更显P2P优势

评价指标

◆ 数据填充率(Window fullness)

- ✱ 某个时间点, 已有数据占整个窗口大小的比例

◆ 丢包率(Segment Miss ratio)

- ✱ 在整个运行过程中, 丢包的数量占有所有应该获取的包的比例

◆ 启动延时(Startup delay)

- ✱ 在决定观看到实际开始之间等待的时间
- ✱ 一般节点获取 $6r$ 个连续的数据块就可以播放了(以码率 r 播放), 一般一个周期为2秒, $6r=12$ 秒钟的数据(6个调度周期)

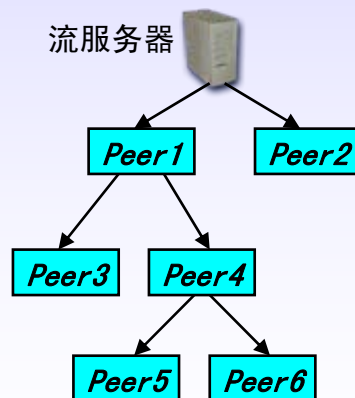
4.3.1 PeerCast

◆ 概述

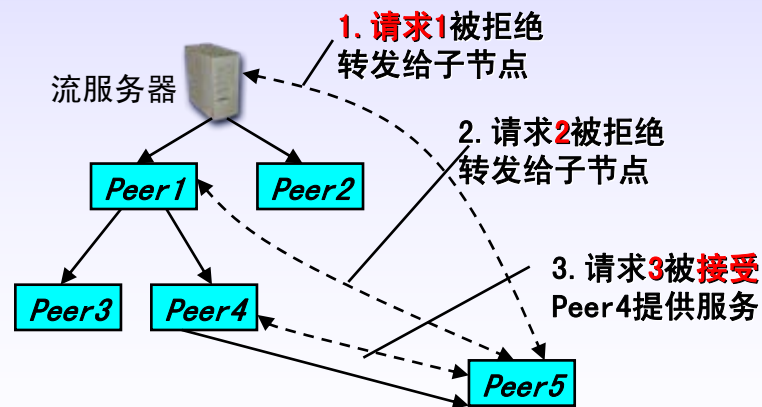
- ♣ 2004成立PeerCast.org非营利站点，免费P2P电台软件
- ♣ 不存在中心服务器，每个Peer同时是客户/服务器
- ♣ 音视频服务器与客户端集合在一起的软件
- ♣ 与其它P2P不同，用户下载的不是文件而是流，并与其它客户交流，没有任何数据存在本地机上

◆ 应用层多播树PeerCast

- ♣ 构造多播树
- ♣ 新节点加入
- ♣ 正常离开
- ♣ 异常离开



PeerCast多播树



新节点加入过程

◆ 转发“节点加入请求”的策略：假设PeerN向PeerE请求

随机选择	PeerE随机选择其子节点进行转发
Round-Robin	PeerE将请求依次转发给自己的子节点队列
Smart-Placement	PeerE将请求 转发给PeerN路由最近的子节点
Smart-Bandwidth	若PeerN带宽好于PeerE，则PeerE将自己的位置给PeerN， PeerE重定向到PeerN

◆ 4种正常离开策略

- ♣ PeerL向其所有**直接子节点**发消息，通知它们通过**PeerL父节点**修复
- ♣ PeerL向其所有直接子节点发消息，通知它们通过**服务器S**修复
- ♣ PeerL向其所有**后续节点**发消息，通知它们通过PeerL父节点修复
- ♣ PeerL向其所有后续节点发消息，通知它们通过服务器S修复

◆ 异常离开策略

- ♣ 每个节点需要周期性向其父节点和子节点发送“自己工作正常”之消息
- ♣ 某节点在若干周期内未收到子节点正常的消息，则认为该子节点非正常离开，且回收该子节点的资源
- ♣ 子节点在若干周期内都没有收到父节点正常的消息，则认为该父节点已经非正常离开，该子节点及所有后续节点都需重新接入网络

4.3.2 BitTorrent类

◆ 改造BitTorrent成为流媒体直播系统

- ♣ 原为一般文件快速分享下载

- ☞ 其分片选择方法令已下载部分非常分散，不适应流媒体

- ♣ 适应流媒体的分片方法：

- ☞ 保障文件未下载完毕前可以实时播放

- ☞ 可以流畅连续收看完整个直播，减少跳帧

- ☞ 设计上改动到仍可兼容BitTorrent协议和现有环境

◆ PPLive

- ♣ Hybird P2P结构：中心是视频源和服务端

- ☞ 视频源提供多种视频直播源

- ☞ 服务器记录当前用户选择的频道（种子信息）

- ☞ 采用Windows的自带播放器（Media Player10），软件体积小

- ♣ 运行方式

- ☞ 节目分成固定长度帧，可从其它节点下载

- ☞ 每个节点要缓冲一定时间内的视频数据，以提供给下一用户

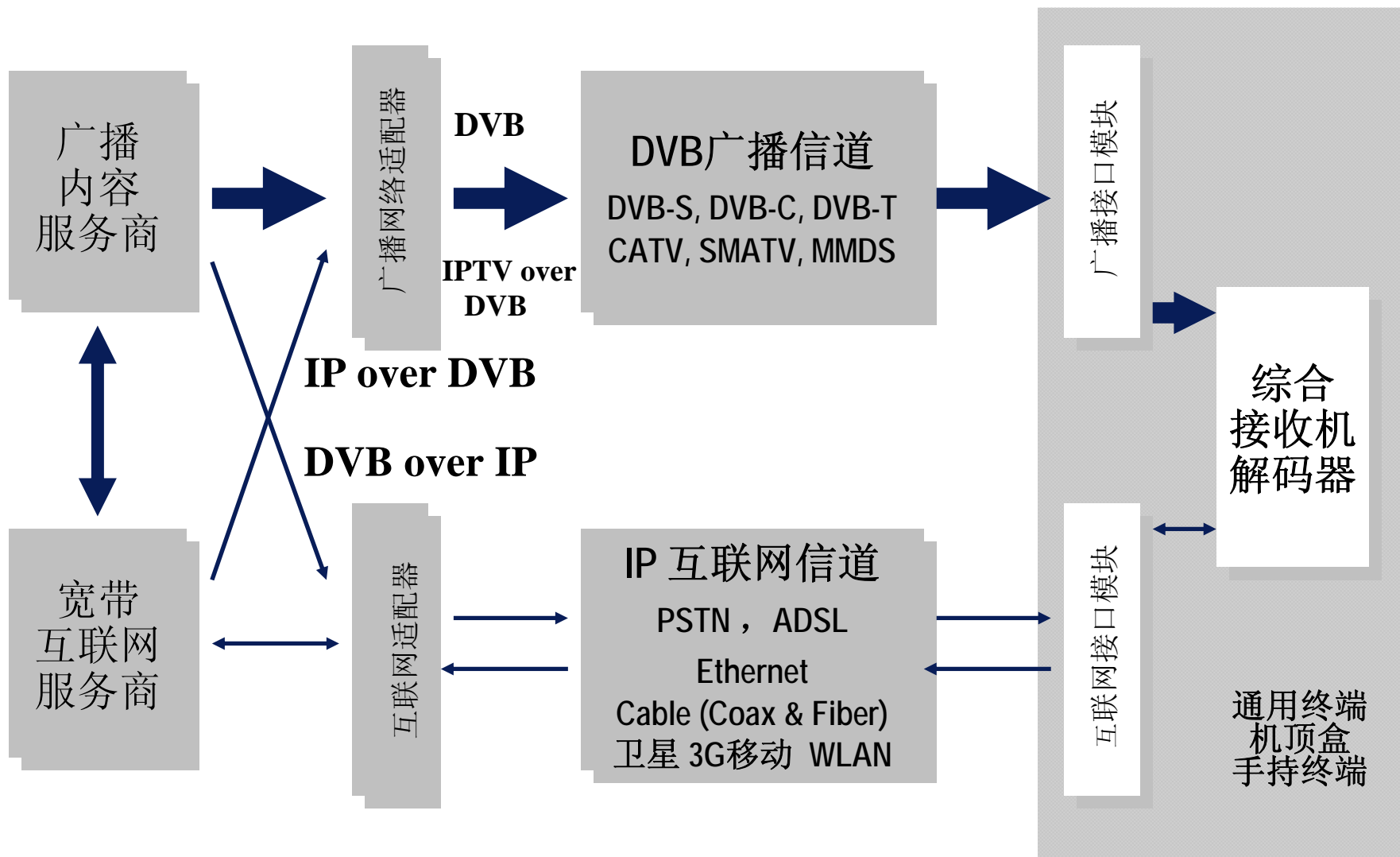
4.3.3 Coolstreaming

- ◆ Coolstreaming 是一种全新的流媒体播放软件，它的**核心技术是P2P**，类似BitTorrent但是**不需要全部下载完了再播放**。是一种P2P流媒体。
 - ✱ 现在支持凤凰卫视，ESPN，HBO电影，CCTV系列等多个电视台的实时转播，同时还提供一些广播节目的转播
- ◆ Coolstreaming可以实现电视视频信号的网络转播。不能大范围接收覆盖的电视台可以利用该技术建立实时的网络转播（免费或DRM付费）提供通达性。
- ◆ 不能安排电视台直播的节目在网上直播。例如新浪等网络门户可以提供独家的媒体见面会，娱乐体育盛事，名人网络聊天的实况转播。大型企业利用该技术辅以其它P2P语音手段实现高质量的网络会议转播等
- ◆ 开辟个人网络电视台/电台，作为私人媒体向网络大众广播
- ◆ 校园IPTV

IPTV推进网络及媒体的融合

- ◆ 网络融合，三网融合并不是一种网络取代其他网络。IP化的数字电视网和宽带互联网将长期共存，互补融合构成同样的宽带交互新媒体。这种融合是发生网络层用户终端处，终端可以根据需要自动选择使用哪个网络。融合也将发生在地面（卫星）数字电视网和宽带无线互联网之间。电视手机将成为移动通信的下一个亮点。
- ◆ 广电和电信的双赢合作，统一的机顶盒和移动终端。

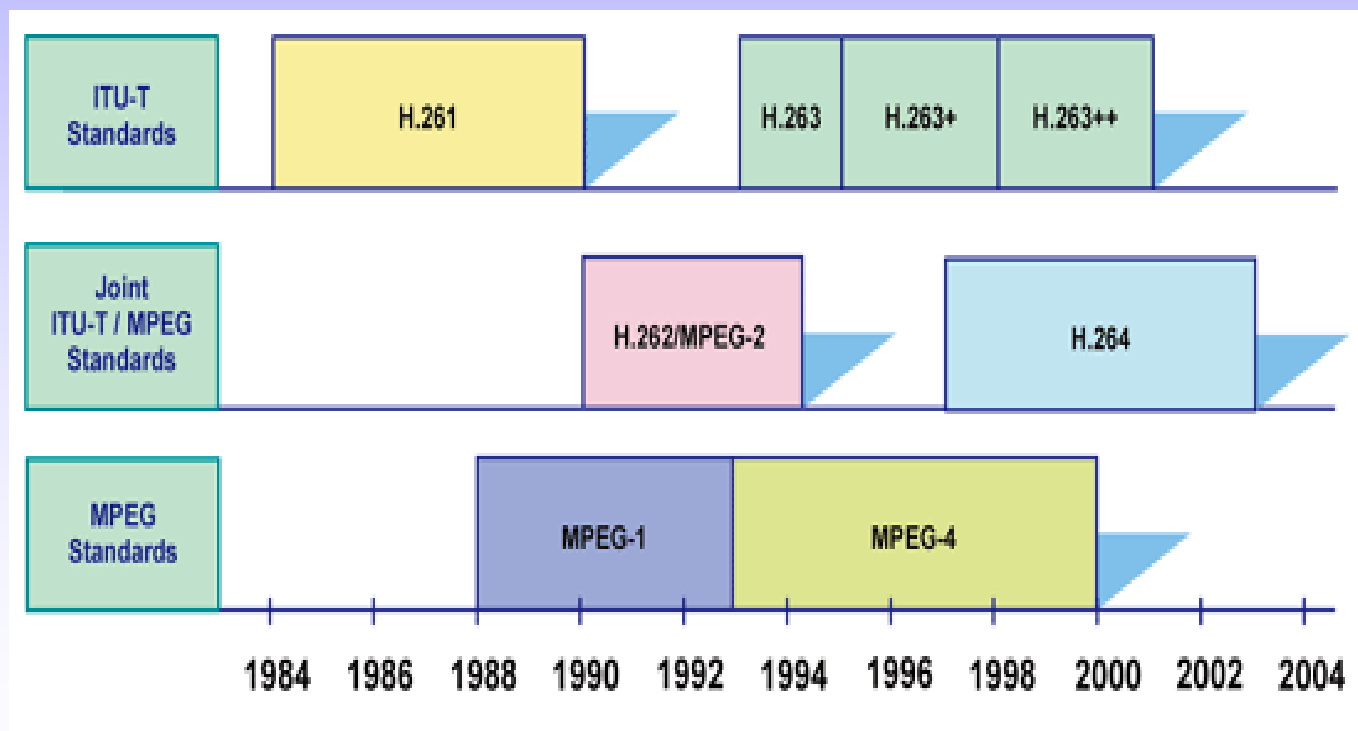
数字电视和宽带IP互联网的融合



媒体融合

- ◆ 媒体融合：网络电视IPTV单纯提供与数字电视类似功能是没有发展前途的。IPTV必须充分发挥互联网的优势将广播**频道媒体和互联网媒体融合**
 - ♣ P2P将网络电视从集中控制的节目表中解放出来
 - ♣ P2P 流媒体降低网络电视直播和转播的门槛。个人网络电视网站成为可能。
 - ♣ P2P网络电视转播大大扩大通达性
 - ♣ P2P结合博客、RSS将产生新的运营模式。
- ◆ 未来统一宽带交互媒体的新运营模式和产业链建立，以及相关政策的制定将成为其能否成功的关键。

数据压缩



ITU-T规范和ISO/IEC MPEG标准的演进历程

Why compression?

- ◆ 无压缩的视频流 (8bit YUV420) 数据量惊人, 无法适用于现有的存储媒介和通讯带宽

例如:

- ◆ VCD (352x288x25FPS): 45分钟的数据量约为10GB, 需要26.5:1的压缩
- ◆ DVD (720x480x30FPS): 2小时的数据量约为100GB, 通常需要15:1的压缩
- ◆ 高清节目广播 (1920x1080x30FPS): 通讯带宽24Mbps (未来希望两路), 需要30:1 (60:1) 的压缩
- ◆ 手机视频聊天 (320x240x15FPS): 单项带宽<128Kbps, 需要>100:1的压缩

视频压缩标准的历史进化...

旧标准:

- ◆ **MOTION-JPEG/SONY-DV** (主要用于视频编辑系统)
- ◆ **H. 261/H. 263** (早期通讯系统)
- ◆ **MPEG-1** (VCD)
- ◆ **MPEG-2** (DVD等), 是目前最成熟使用最广泛的工业标准

过渡性标准:

- ◆ **MPEG-4** (DivX, rm...), 目前网络流传最广的视频格式

视频压缩标准的历史进化 (cont'd)

新一代标准:

- ◆ H. 264 (H. 26L, JVT, MPEG-4 PART10), 下一代高清DVD/广播, MOBILE通讯, WHITE HOT!!!
IC leading companies: WIS, ST, SandVideo (Broadcom)
- ◆ WMV-9 (CORONA, VC-9, VC-1), 微软标准, H. 264的竞争对手, 估计会统治PC世界
IC leading companies: WIS, SigmaDesign
- ◆ AVS, 中国未来数字广播标准?

视频压缩技术组成

- ◆ 静止图像压缩 (JPEG/JPEG-2000)
 - ♣ 色空间 (YUV422/420, RGB, 8/10/12b)
 - ♣ 空间域→频域变换 (离散余弦, 小波)
 - ♣ 量化 (有损)
 - ♣ 无损熵编码 (可变长编码, 算数编码)
- ◆ 运动估计和矢量/残差压缩 (motion estimation)
- ◆ 环内滤波 (in-loop filter)
- ◆ 码流控制 (bit-rate control)
- ◆ 各类上下文预测模型 (context prediction models)
- ◆ 码流-失真模型 (rate-distortion model)

变换 (TRANSFORM)

目的：集中低频能量，数值分布更有规律

◆ 离散余弦 (Discrete Cosine Transform)

特点：分块操作

优点：噪音局部化

缺点：边界误差明显

- ♣ 8x8 浮点 (IDCT mismatch)
- ♣ 8x8/4x8/4x4 微软整点 (WMV-9)
- ♣ 8x8/4x4 H. 264整点

◆ 小波 (wavelet)

♣ 特点：全局分层

♣ 优点：无边界效应

♣ 缺点：时域噪音明显，且不适合块类残差 (block match)

- 5/3基，无损整数变换
- 9/7基，浮点变换
- 8/8, 6/6, ...

熵编码 (ENTROPY CODING)

$$\text{熵} = \ln(1/P_1) + \ln(1/P_2) + \dots + \ln(1/P_n)$$

◆ 可变长编码 (Variable Length Coding)

优点：操作简单

缺点：整数码表位长

- ✦ 各种赫夫曼树

- ✦ 哥伦布码 (H. 264/AVS)

◆ 二值算数编码 (Binary Arithmetic Coding)

优点：理论最高效率 (小数码表位长)

缺点：计算复杂，不可并行

- ✦ QM-coder (JBIG, JPEG-2000)

- ✦ Modulo-coder (H. 264)

◆ 上下文自调整概率模型 (Context Adaptive Models)

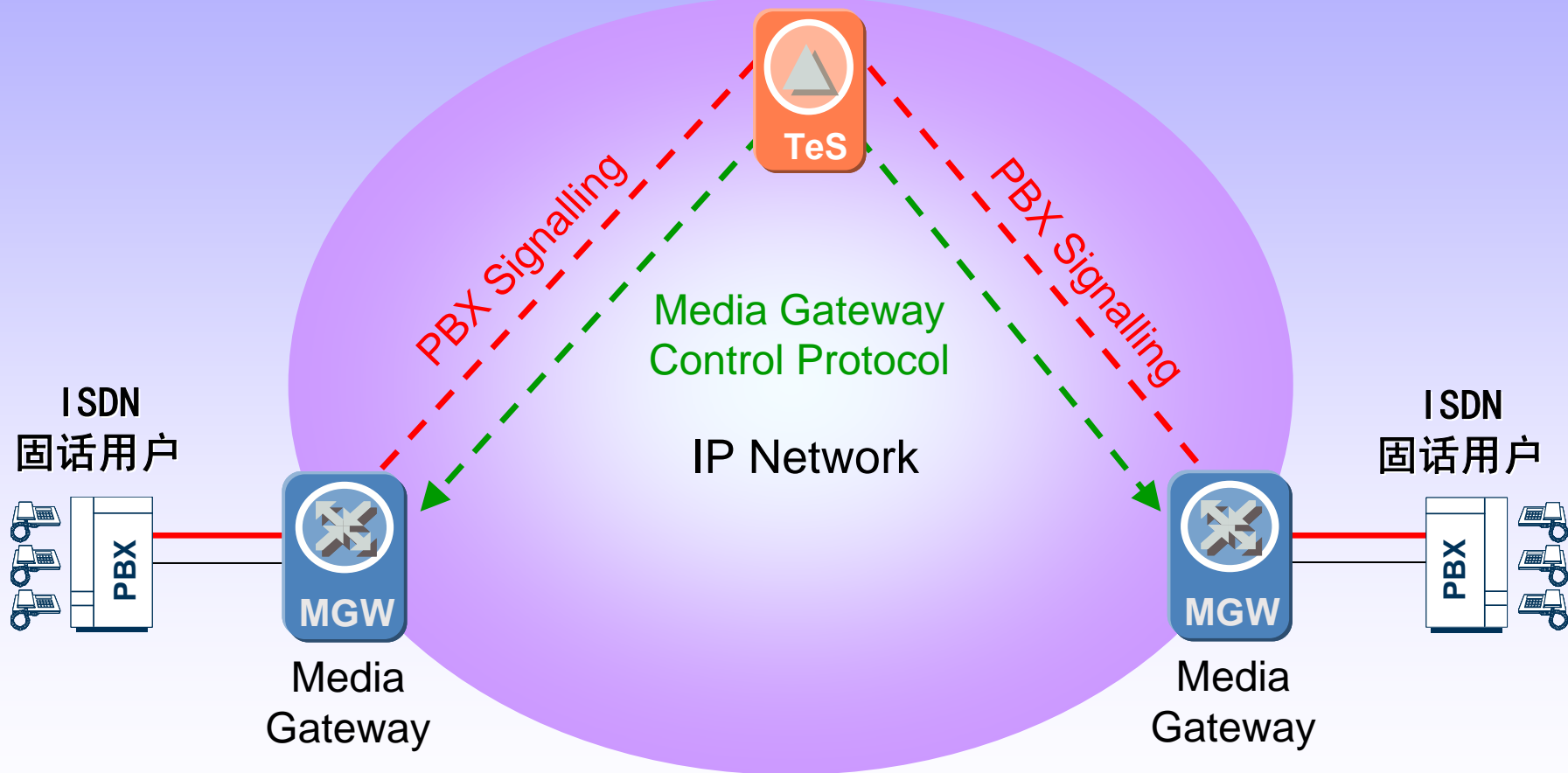
运动估计 (MOTION ESTIMATION)

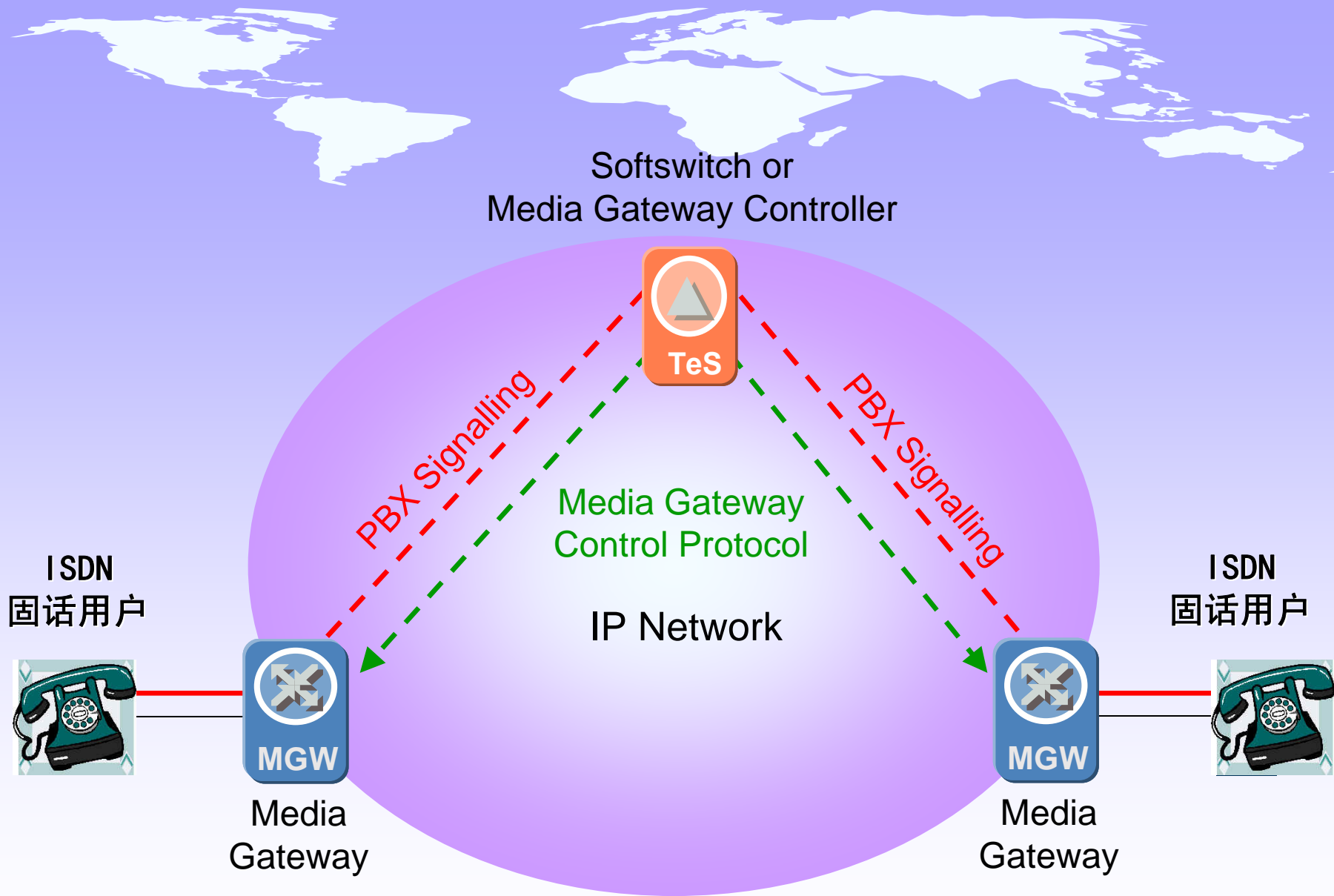
- ◆ 运动模型
 - ♣ 二维矢量块类运动 (2-vector block match)
 - ♣ 多参数仿射变换
 - ♣ 基于可变形态对象的运动预测 (segmentation-based)
- ◆ 运动参数的发展趋势
 - ♣ 精度 (整点→半点→1/4点)
 - ♣ 自适应分块 (16x16→8x8→4x4)
 - ♣ 多帧预测 (multi-reference prediction)
 - ♣ 加权预测 (weighted/intensity prediction)
- ◆ 运动估计的快速算法
 - ♣ 运动矢量逐次细化法 (diamond search, etc.)
 - ♣ 图像空间逐次细化法 (multi-scale search)
 - ♣ 像素深度逐次细化法 (bit-layer search)
- ◆ 不可忽略的模式, 矢量等运动参数的编码代价 (与量化系数有关)
- ◆ 码流-失真模型 (高复杂度的串行操作)

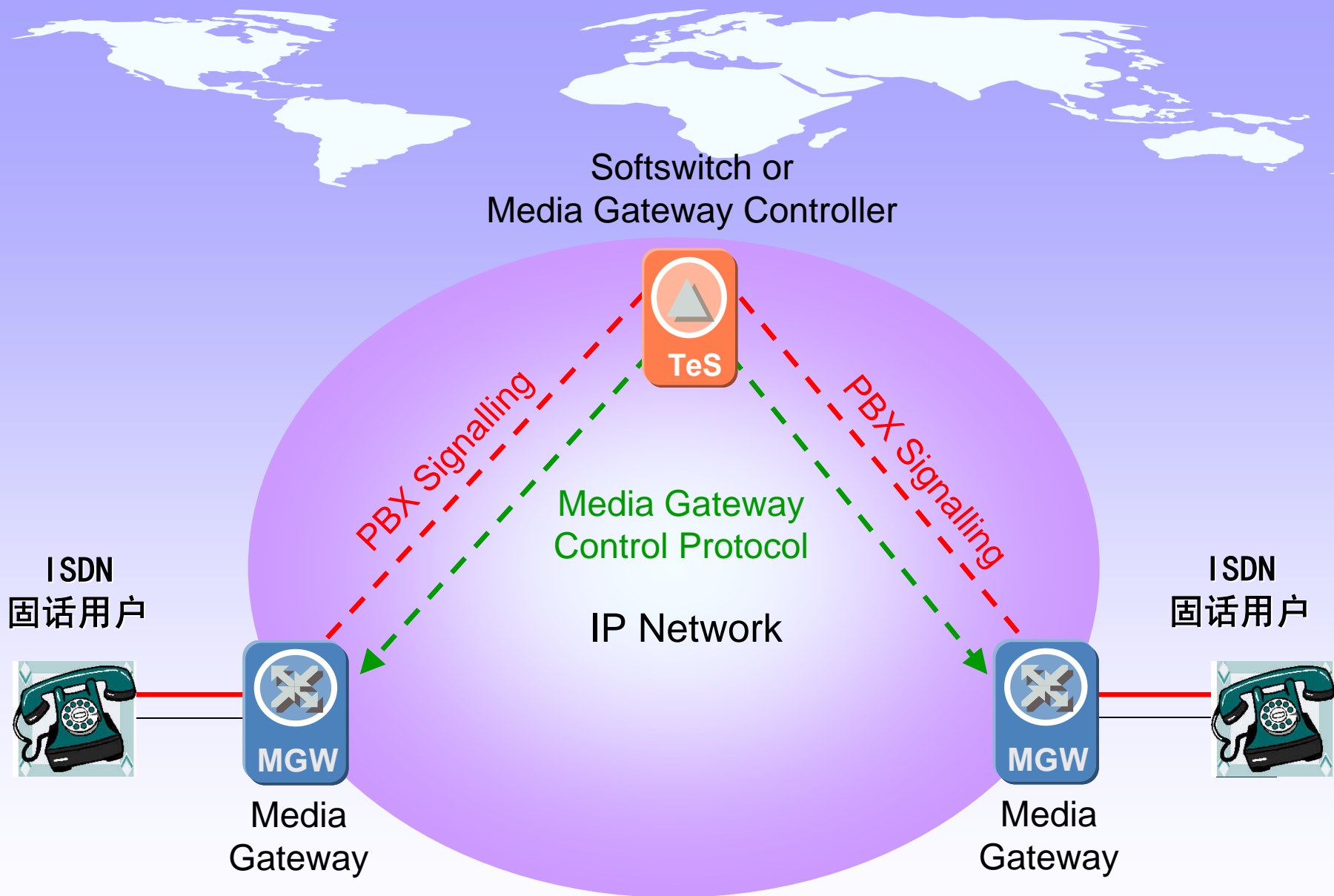
其它相关的视频图像处理

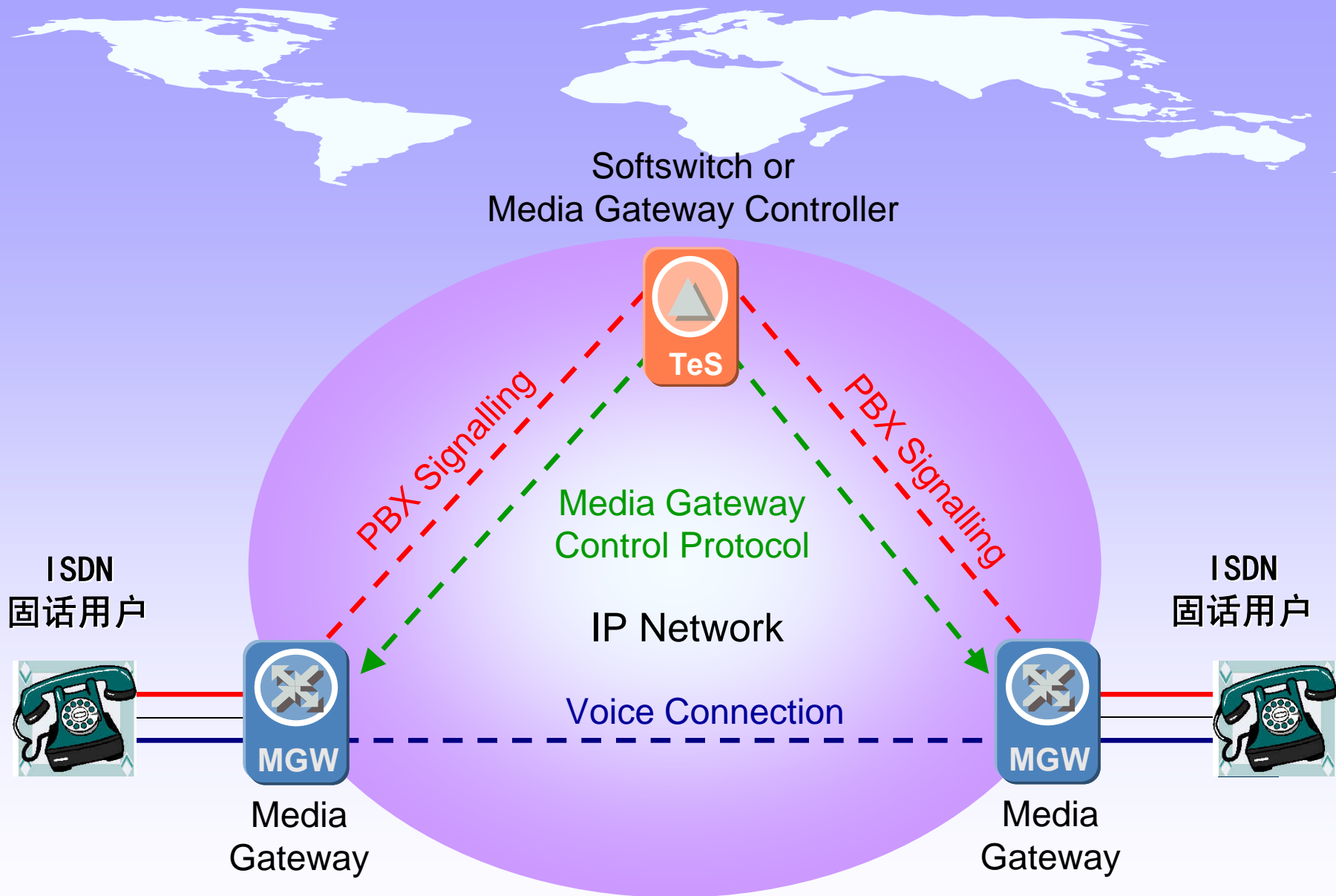
- ◆ 隔行扫描恢复 (de-interlace)
 - ♣ 空间域滤波的副作用: 格边 (alias)
 - ♣ 时间域滤波的副作用: 重影 (ghost)
 - ♣ 加入运动预测的问题: 带宽增加, 高运算量, 稳定性差
- ◆ RGB Bayer 重构 (de-mosaic)
 - ♣ 保持锐边界
 - ♣ 去色边
 - ♣ 减少迭代次数
- ◆ 图像滤波/放缩
 - ♣ 线性滤波器: 简单, 模糊边界, 格边
 - ♣ 保持锐边界的非线性滤波器 (non-linear edge-preserving)

Softswitch or
Media Gateway Controller





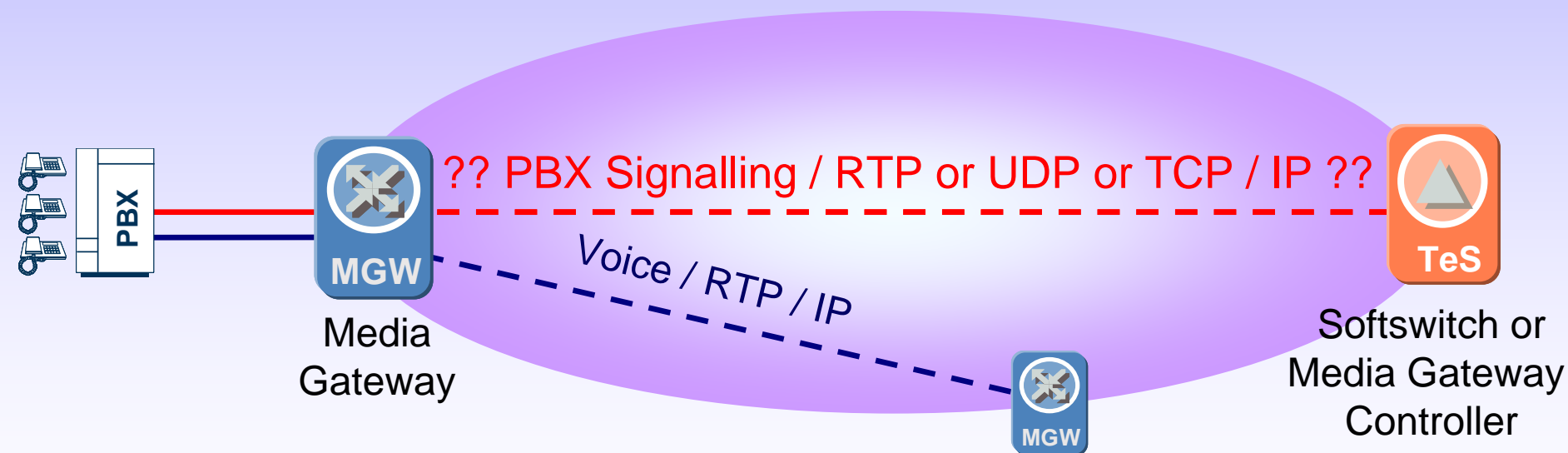






IP是怎样承载电话服务的?

声音在IP 上用 “Real Time Protocol” ,
但是怎样承载拨号信令的呢?





Thank
you!



Questions