



# 《多媒体通信》课程专题

## 主流音视频流结构：裸数据的组织

### Encapsulation for Multimedia Data



# 回顾：多媒体数据压缩的思路

## ◆ 思路1：尽量模拟原始信号波形

- 时域：例PCM(G.711)、ADPCM(G.721)、**△M**(SACD);
- 变换域：例DCT(JPEG)、MDCT(MP3)、DWT(J2P)

## ◆ 思路2：用模型来生成信号逼近原始信号波形

- 例，语音生成模型，**滤波器+激励**
- 例，Facial animation parameters (FAPs) define a face

## ◆ 思路3：人的感知特性

- 例，MP3中的临界频带
- 例，RGB→YUV



# 40年来音频压缩标准进展

G.711	A-lawμ-lawPCM	ITU-T	1972
G.721	ADPCM/8kHz	ITU-T	1984
G.722	sub-band ADPCM/16kHz	ITU-T	1988
AC-3	MDCT/32-48kHz	Dolby Laboratories	1992
MP3	MDCT,Hybrid Subband/8-48kHz	ISO/IEC MPEG Audio	1993
AAC	MDCT,Hybrid Subband/8-192kHz	ISO/IEC MPEG Audio	1997
AMR	ACELP/8kHz	3GPP	1999
Vorbis	MDCT/1-200 kHz	Xiph.Org Foundation	2000
AMR-WB	ACELP/16kHz	3GPP	2001
FLAC	Lossless/1-65.5kHz	Xiph.Org Foundation	2001
ALAC	Lossless/1-38.4kHz	Apple Inc.	2004
ALS	MPEG-4 Audio Lossless Coding /up to 384 kHz	ISO/IEC MPEG Audio	2005
CELT	MDCT/32-96 concepts from CELP	Xiph.Org Foundation	2007



# 20年来图像和视频压缩标准的进展

## ◆ 图像

- JPEG, 1992
- JPEG 2000, 2000
- JPEG XR, 2009

## ◆ 视频

- H.261, 1988
- MPEG-1, 1993
- **MPEG-2/H.262, 1996**
- H.263, 1996
- MPEG-4, 1999
- **MPEG-4 AVC/H.264, 2003**
- **HEVC/H.265, 2013**

## ◆ China AVS, 2003

## ◆ Temporal

- Prediction

## ◆ Spatial

- Transform

● DCT(JPEG) → DWT(JPEG2000) → LBT(JPEG-XR)

- 帧内预测(MPEG4 AVC/H.264)

## ◆ Entropy

- Huffman 编码  
(JPEG) → EBCOT(JPEG2000) → VLC(JPEG-XR)

## ◆ Inter-View

- Prediction



# 《多媒体通信》课程专题

## 主流音视频流结构：裸数据的组织

## Encapsulation for Multimedia Data



# 裸数据的组织

## ◆ 现有网络中数据组织的方式

- 计算机：各种数据文件的格式
- 网络：数据包格式
- 哪些信息需要合理表示？
- 交换数据格式

## ◆ 主流音视频流的数据组织方式

- 数字音乐 CD-DA
- 音乐文件 MP3
- 数字音频广播 DAB
- 数字电视 MPEG2
- MPEG-4 AVC / H.264



# 计算机：各种数据文件的格式 压缩标准不同，浏览器如何识别？

◆ MIME的英文全称是"Multipurpose Internet Mail Extensions" 多功能Internet 邮件扩充服务，在1992年最早应用于电子邮件系统，后来也应用到浏览器。服务器会将它们发送的多媒体数据的类型告诉浏览器，而通知手段就是说明该多媒体数据的MIME类型，从而让浏览器知道接收到的信息哪些是MP3文件，哪些是Shockwave文件等等。服务器将MIME标志符放入传送的数据中来告诉浏览器使用哪种插件读取相关文件。

## 用于WAP服务器的MIME类型有：

MRP文件（国内普遍的手机）	.mrp application/octet-stream
IPA文件(IPHONE)	.ipa application/iphone-package-archive
	.deb application/x-debian-package-archive
APK文件(安卓系统)	.apk application/vnd.android.package-archive
CAB文件(Windows Mobile)	.cab application/vnd.cab-com-archive
XAP文件(Windows Phone 7)	.xap application/x-silverlight-app
SIS文件(symbian平台/S60V1)	.sis application/vnd.symbian.install-archive
SISX文件(symbian平台/S60V3/V5)	.sisx application/vnd.symbian.epoc/x-sisx-app
JAR、JAD文件(JAVA平台手机通用格式)	.jar .jad



# 计算机：各种数据文件的格式 压缩标准不同，播放器如何识别？

.gif 无标题.jpg |

```
00000000h: FF D8 FF E0 00 10 4A 46 49 46 00 01 01 01 01 00 60 ; ?JFIF.....  
00000010h: 00 60 00 00 FF E1 00 5A 45 78 69 66 00 00 4D 4D ; .Exif..MM  
00000020h: 00 2A 00 00 00 08 00 05 03 01 00 05 00 00 00 01 ; *.*****
```

无标题.gif |

```
00000000h: 47 49 46 38 39 61 43 03 8B 01 F7 00 00 00 00 00 ; GIF89a.??....  
00000010h: 00 00 33 00 00 66 00 00 99 00 00 CC 00 00 FF 00 ; .3. ....??.?..  
00000020h: 2B 00 00 2B 33 00 2B 66 00 2B 99 00 2B CC 00 2B ; +.+3.+f.+?+?+
```

无标题.bmp |

```
00000000h: 42 4D FA 1D OF 00 00 00 00 00 00 36 00 00 00 28 00 ; BM?.....6...(.  
00000010h: 00 00 43 03 00 00 8B 01 00 00 01 00 18 00 00 00 ; .C ..?.....  
00000020h: 00 00 C4 1D OF 00 00 00 00 00 00 00 00 00 00 00 ; ...?
```

无标题.png |

```
00000000h: B9 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52 ; 始PNG.....IHDR  
00000010h: 00 00 03 43 00 00 01 8B 08 02 00 00 00 8A 86 DE ; .C ..?.....?。  
00000020h: 48 00 00 00 01 73 52 47 42 00 AE CE 1C E9 00 00 ; H....sRGB. ....?
```



# 计算机：各种数据文件的格式

## BMP文件格式

- ◆位图文件(Bitmap-File, BMP)格式是Windows采用的图像文件存储格式。位图文件可看成由4个部分组成：位图文件头(bitmap-file header)、位图信息头(bitmap-information header)、彩色表(color table)和定义位图的字节阵列。





# 计算机：各种数据文件的格式

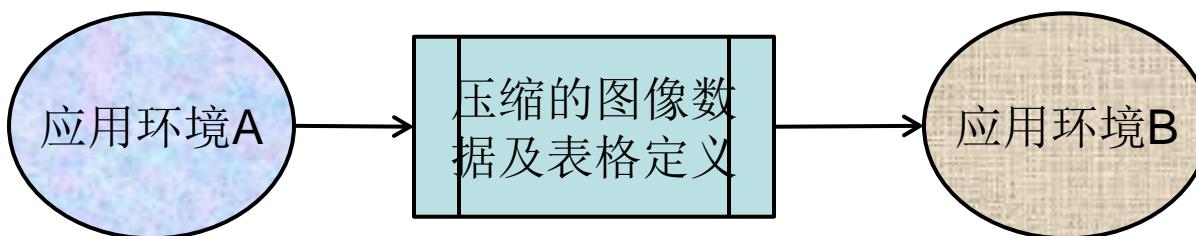
## JPEG标准的三个基本要素

- ◆ 编码器

- ◆ 解码器

- ◆ 交换格式

- 交换格式是压缩图像数据的表示，除要求的标记字段和熵编码段外，还包括了编码中使用的所有表格。交换格式用于不同应用环境之间。





# 计算机：各种数据文件的格式

## JPEG File Interchange Format (JFIF)

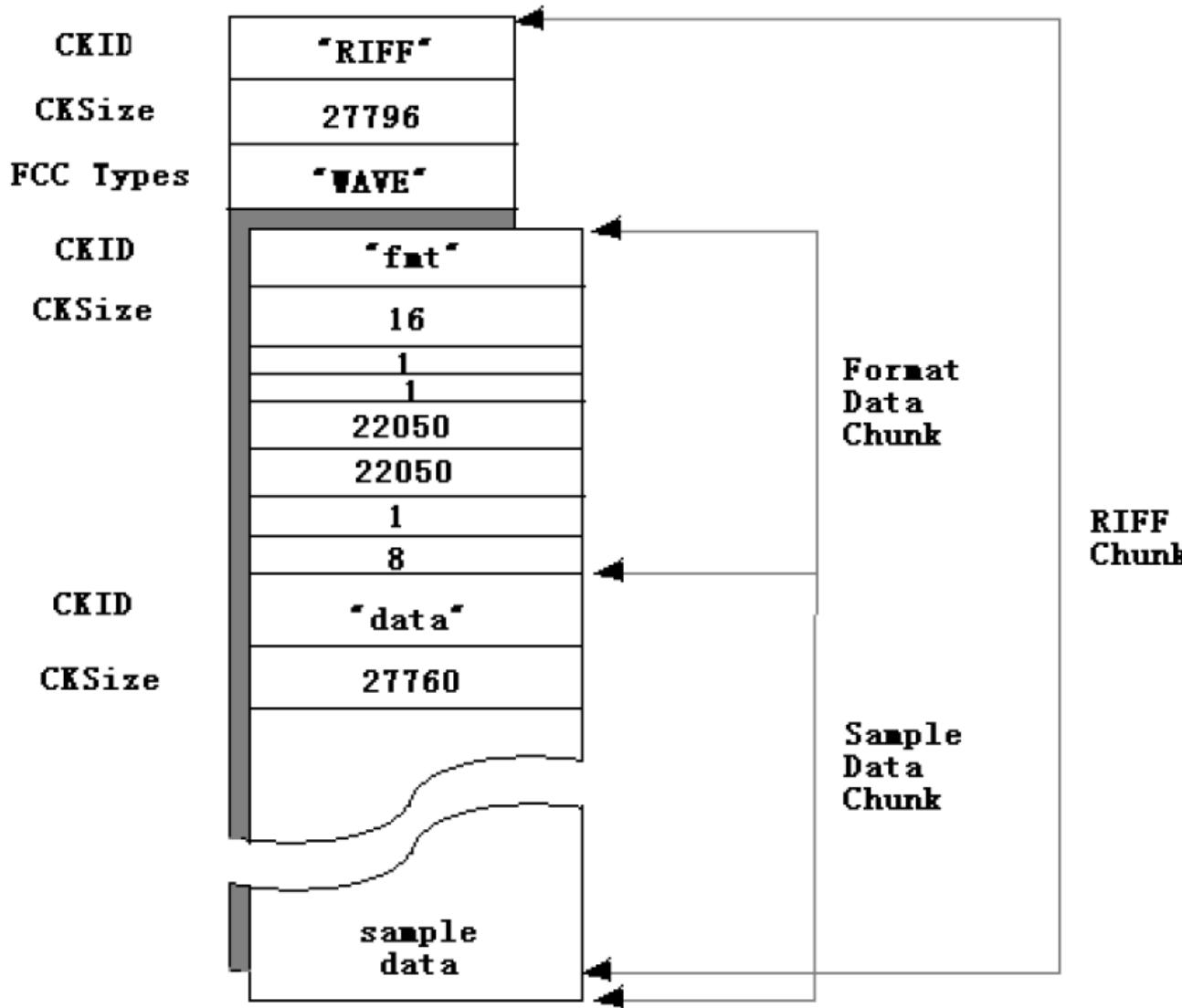
- ◆ A JPEG image consists of a sequence of segments, each beginning with a marker, each of which begins with a 0xFF byte followed by a byte indicating what kind of marker it is.

Short name	Bytes	Name
SOI	0xFF, 0xD8	Start Of Image
SOF0	0xFF, 0xC0	Start Of Frame (Baseline DCT)
SOF2	0xFF, 0xC2	Start Of Frame (Progressive DCT)
DHT	0xFF, 0xC4	Define Huffman Table(s)
DQT	0xFF, 0xDB	Define Quantization Table(s)
DRI	0xFF, 0xDD	Define Restart Interval
SOS	0xFF, 0xDA	Start Of Scan
RSTn	0xFF, 0xD0 ... 0xD7	Restart
APPn	0xFF, 0xEn	Application-specific
COM	0xFF, 0xFE	Comment
EOI	0xFF, 0xD9	End Of Image



# 计算机：各种数据文件的格式

## Resource Interchange File Format (RIFF)





# 小结：计算机中各种数据文件的格式

## Container format

- ◆ A **container** or wrapper format is a **meta-file format** whose specification describes how different data elements and metadata coexist in a computer file.
- ◆ Among the earliest cross-platform container formats were Distinguished Encoding Rules and the 1985 **Interchange File Format**. Containers are frequently used in multimedia applications.
- ◆ Since the **container does not describe how data or metadata is encoded**, a program able to identify and open a container file might not be able to decode the contained data. This may be caused by the program lacking the required decoding algorithm, or the meta-data not providing enough information.



# 裸数据的组织

## ◆ 现有网络中数据组织的方式

- 计算机：各种数据文件的格式
- 网络：数据包格式
- 哪些信息需要合理表示？
- 交换数据格式

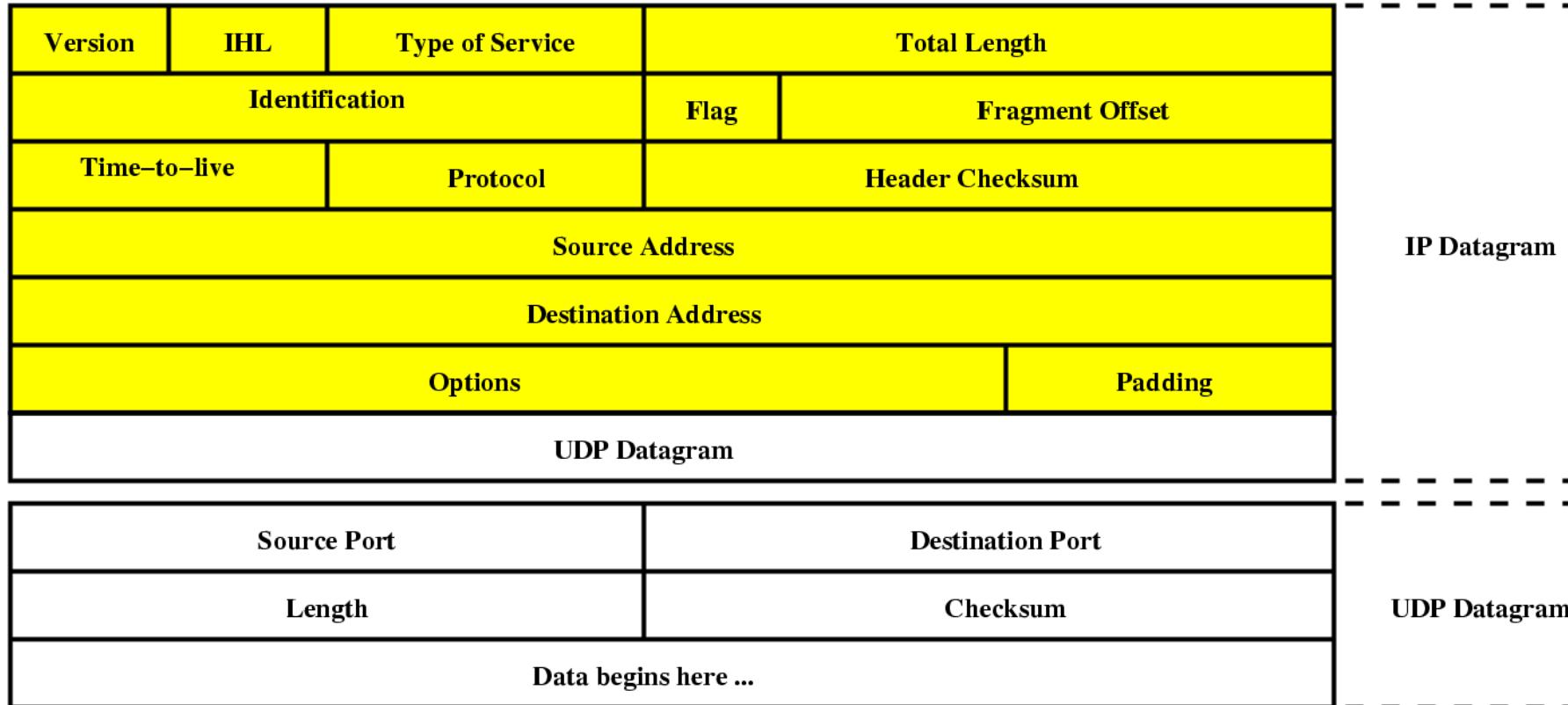
## ◆ 主流音视频流的数据组织方式

- 数字音乐 CD-DA
- 音乐文件 MP3
- 数字音频广播 DAB
- 数字电视 MPEG2
- MPEG-4 AVC / H.264



# 网络：数据包格式

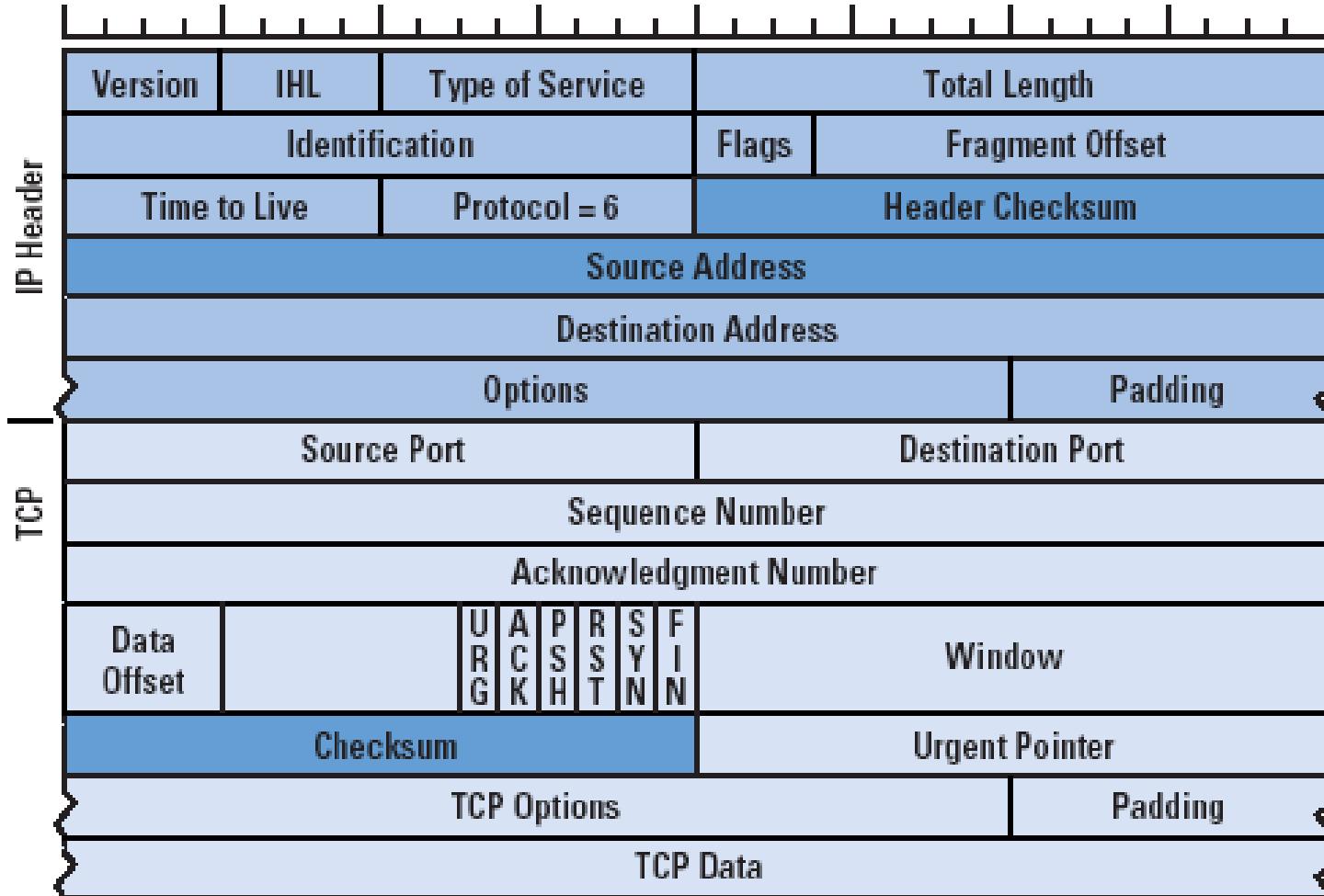
## IP/UDP数据包格式





# 网络：数据包格式

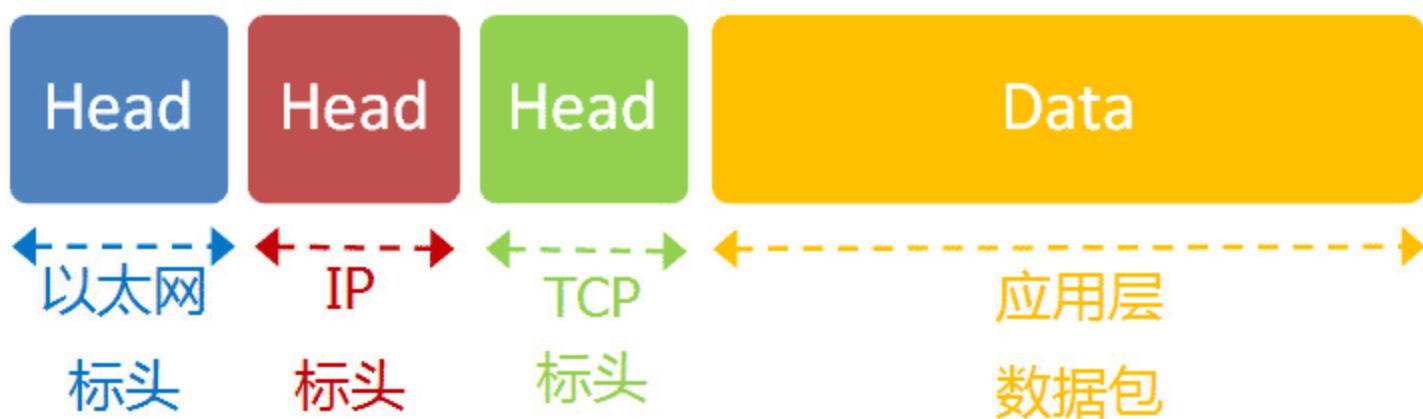
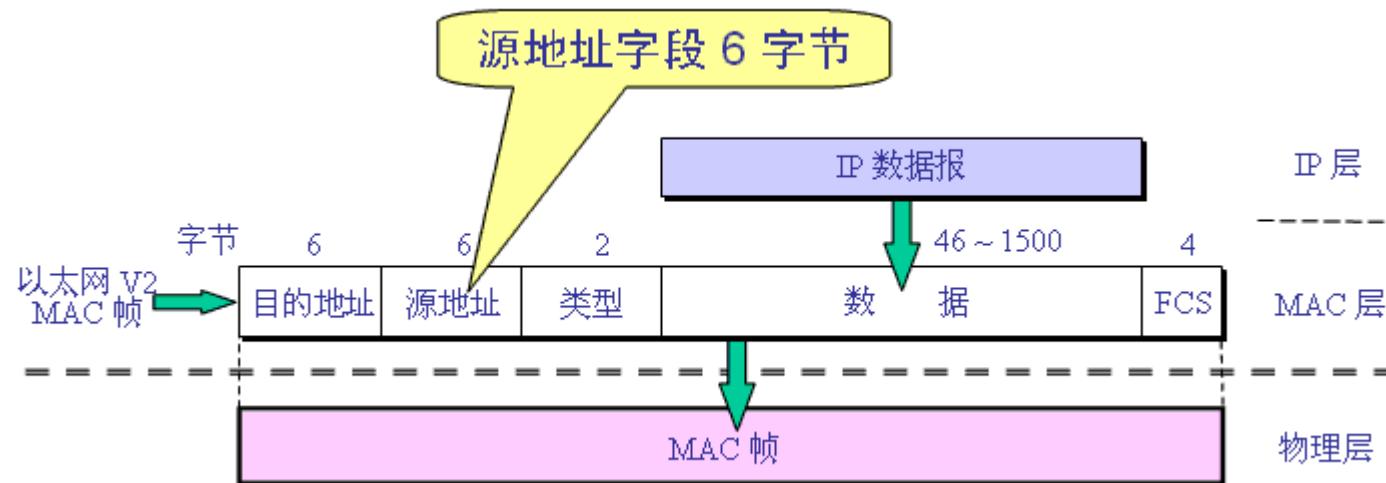
## IP/TCP数据包格式





# 网络：数据包格式

## Ethernet帧格式





# 网络：数据包格式 IPv6头标格式

IPv4 Header				IPv6 Header			
Version	IHL	Type of Service	Total Length	Version	Traffic Class	Flow Label	
Identification		Flags	Fragment Offset	Payload Length		Next Header	Hop Limit
Time to Live	Protocol	Header Checksum		Source Address			
Destination Address							Destination Address
Options		Padding					

**Legend**

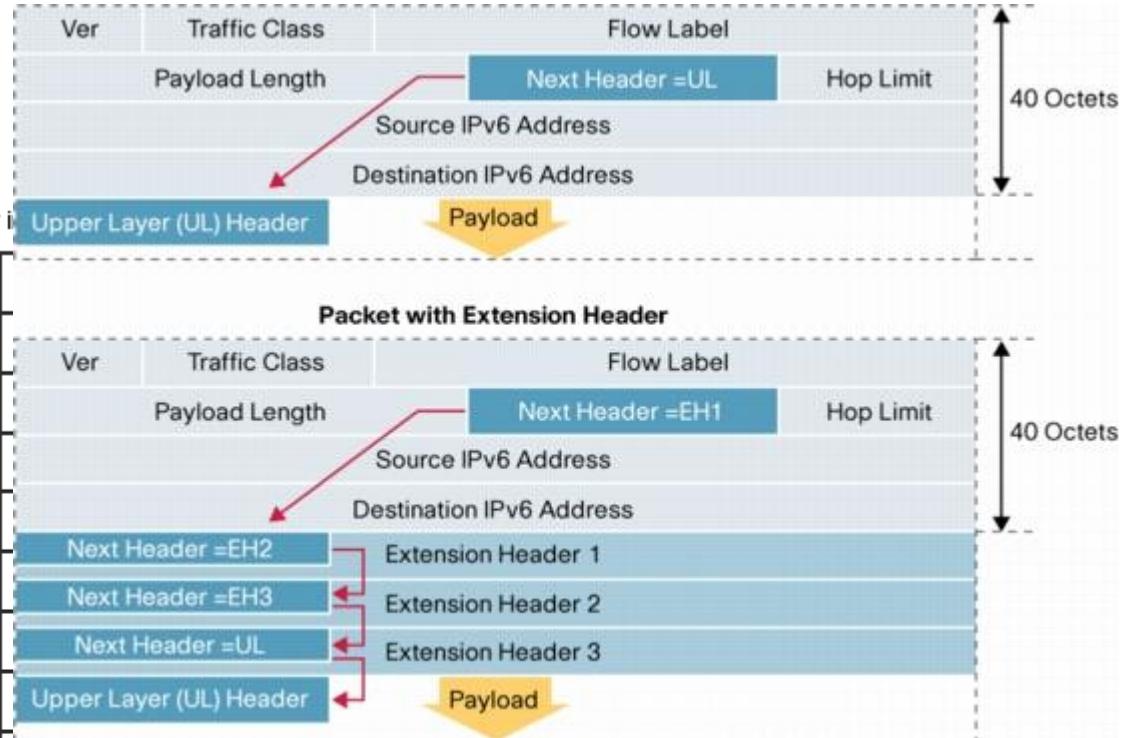
- Field's name kept from IPv4 to IPv6
- Field not kept in IPv6
- Name and position changed in IPv6
- New field in IPv6



# 网络：数据包格式 IPv6扩展头标格式

Table 1. IPv6 Extension Headers and their Recommended Order

Order	Header Type	
1	Basic IPv6 Header	
2	Hop-by-Hop Options	
3	Destination Options (with Routing Options)	
4	Routing Header	
5	Fragment Header	
6	Authentication Header	
7	Encapsulation Security Payload Header	
8	Destination Options	60
9	Mobility Header	135
	No next header	59
Upper Layer	TCP	6
Upper Layer	UDP	17
Upper Layer	ICMPv6	58





# 小结：网络中的数据包格式



# 裸数据的组织

## ◆ 现有网络中数据组织的方式

- 计算机：各种数据文件的格式
- 网络：数据包格式
- 哪些信息需要合理表示？
- 交换数据格式

## ◆ 主流音视频流的数据组织方式

- 数字音乐 CD-DA
- 音乐文件 MP3
- 数字音频广播 DAB
- 数字电视 MPEG2
- MPEG-4 AVC / H.264



# 哪些信息需要合理表示? Windows操作系统的注册表信息

The screenshot shows the Windows Registry Editor window. The left pane displays a tree view of registry keys under '计算机' (Computer). The right pane is a table with columns '名称' (Name), '类型' (Type), and '数据' (Data). The table contains three entries:

名称	类型	数据
(默认)	REG_SZ	(数值未设置)
InstallDir	REG_SZ	C:\Program Files\Bonjour\
Version	REG_SZ	3.0.0.10

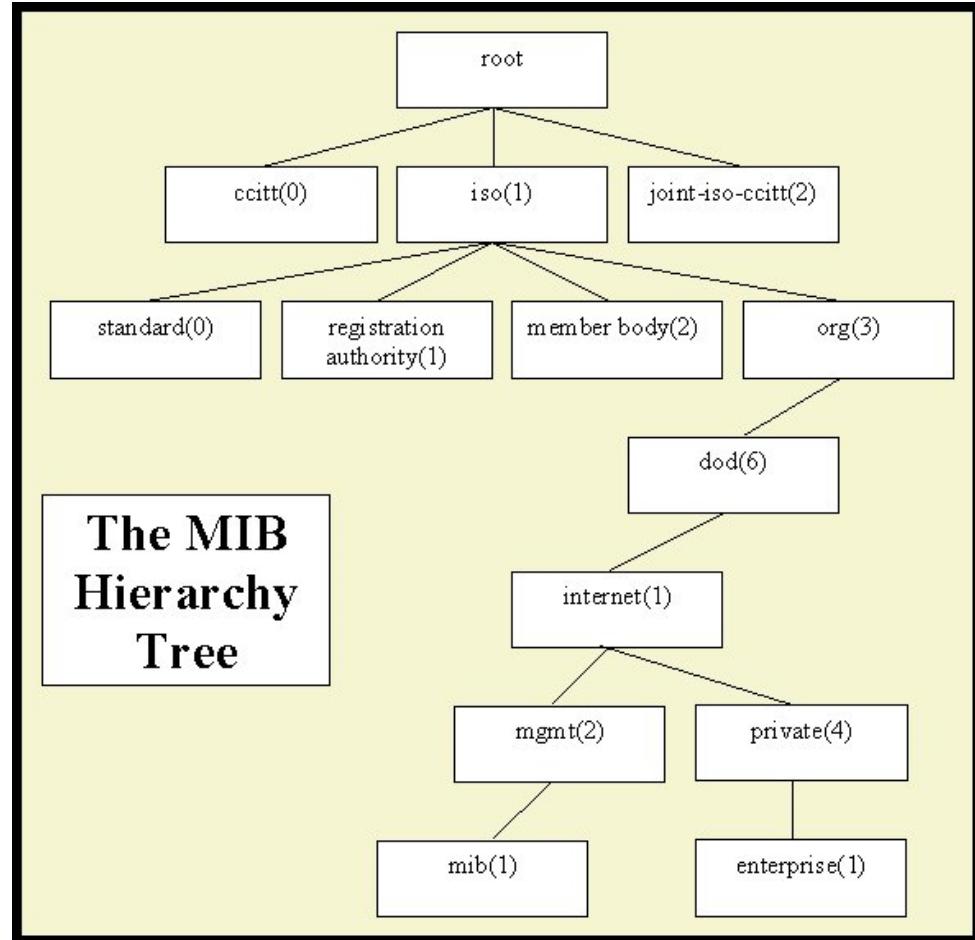
The status bar at the bottom shows the path: '计算机\HKEY\_LOCAL\_MACHINE\SOFTWARE\Apple Inc.\Bonjour'. The URL 'http://www.ustc.edu.cn' is visible in the bottom right corner.



# 哪些信息需要合理表示? 网络设备的管理信息库MIB

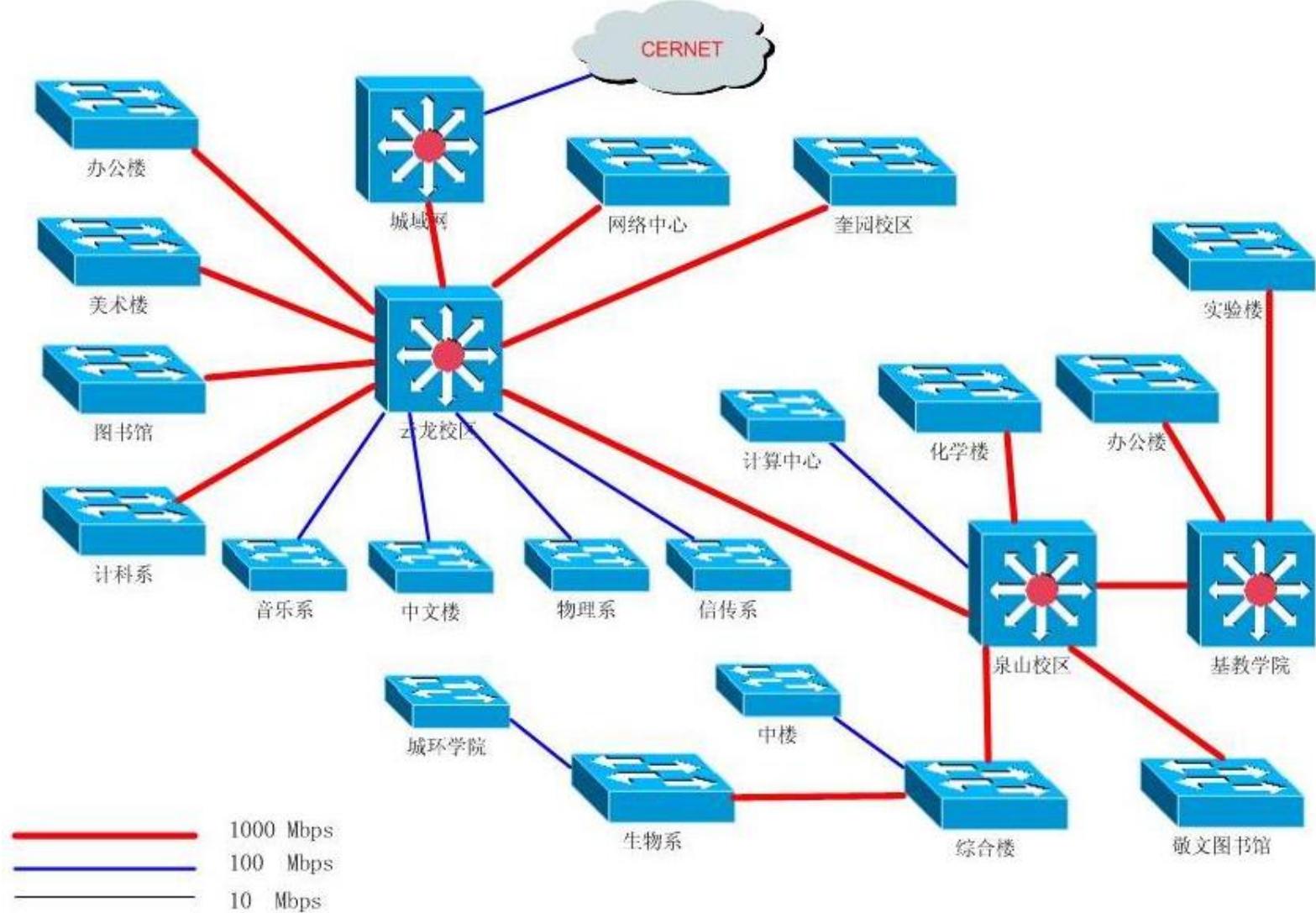
A **management information base (MIB)** is a database used for managing the entities in a communications network. Objects in the MIB are defined using a subset of Abstract Syntax Notation One (ASN.1).

In telecommunications and computer networking, **Abstract Syntax Notation One (ASN.1)** is a standard and flexible notation that describes data structures for representing, encoding, transmitting, and decoding data.



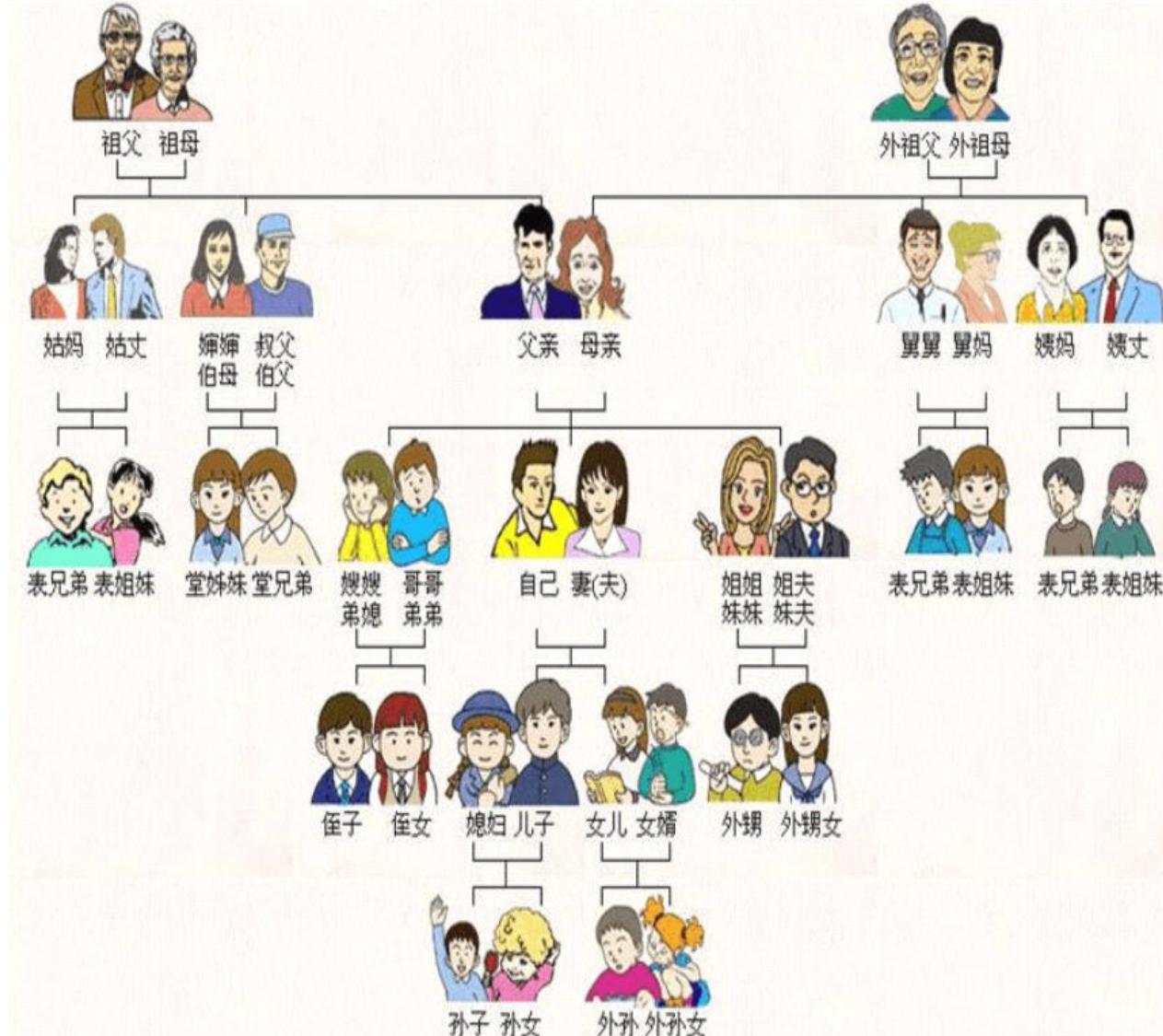


# 哪些信息需要合理表示? 由设备MIB生成拓扑图





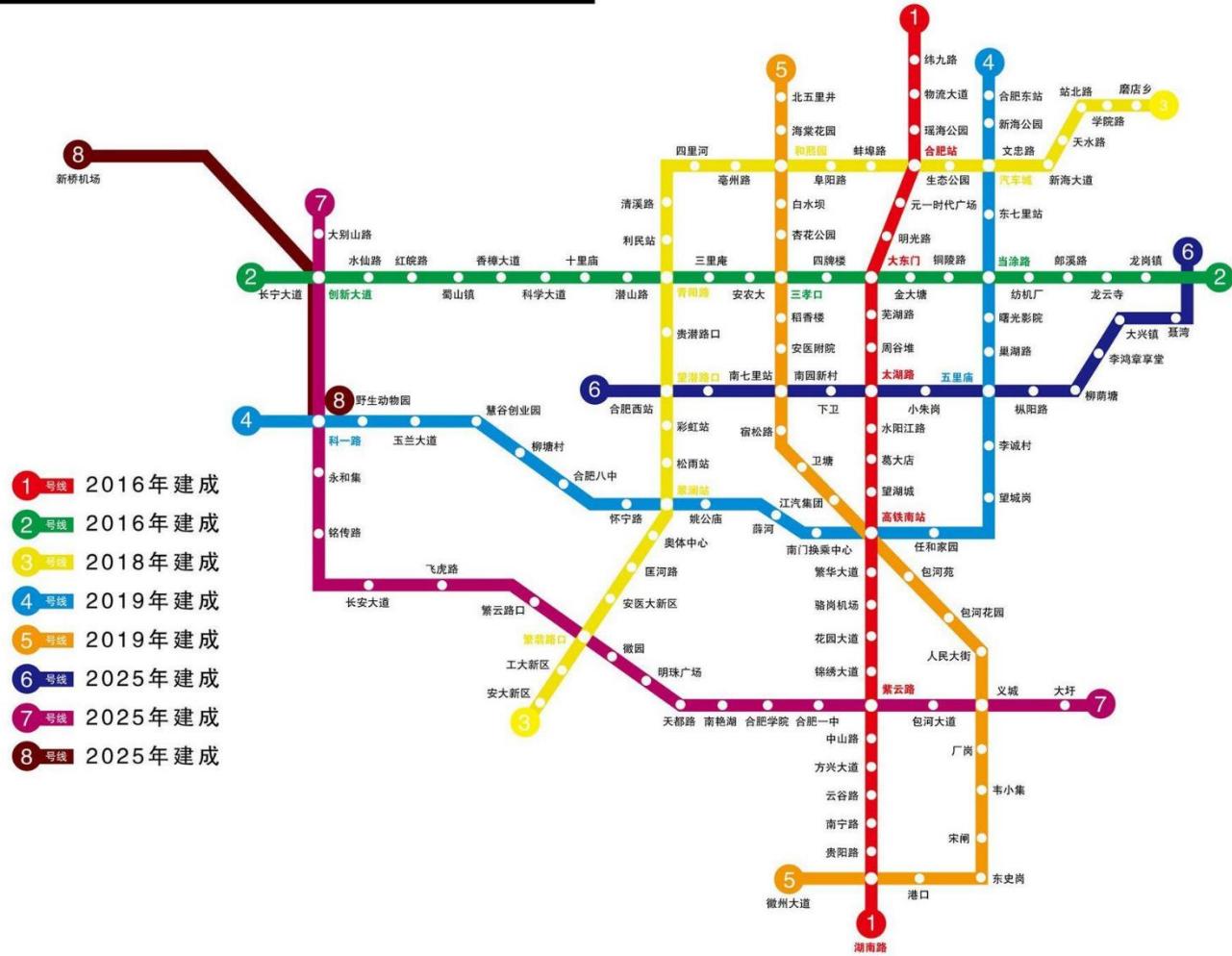
# 哪些信息需要合理表示? 中国亲戚关系称谓图





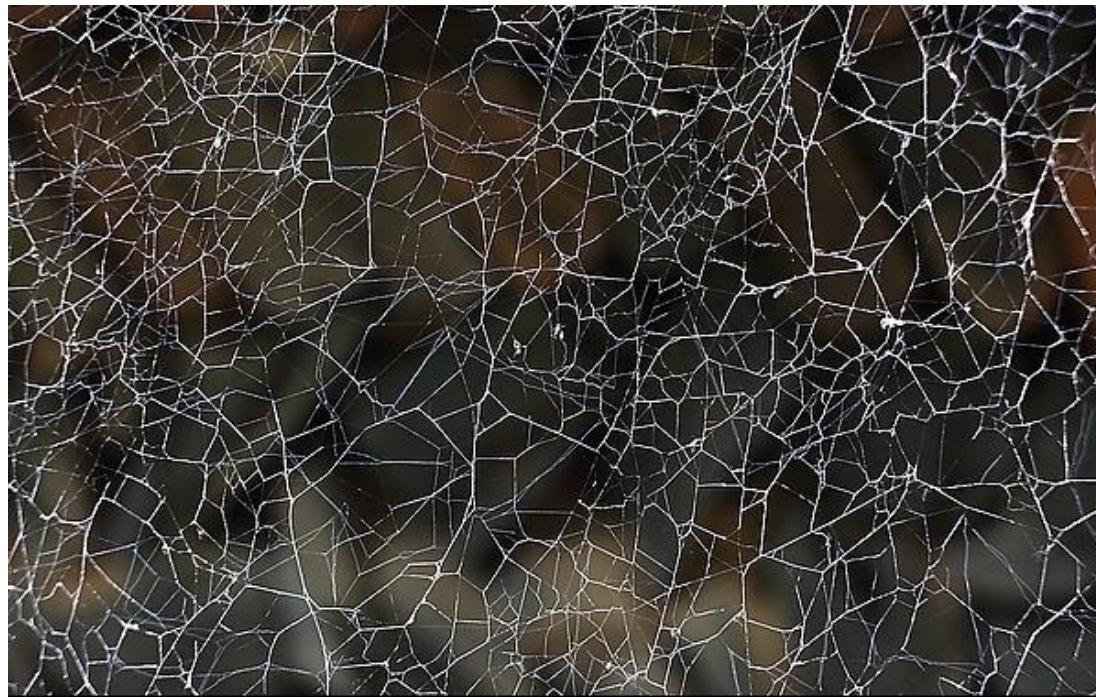
# 哪些信息需要合理表示? 道路交通信息

## 合肥地铁站点信息图





# 小结：哪些信息需要合理表示？



数据项**表示的方式**  
需要表示的数据**总量**  
数据项之间的**关系**





# 裸数据的组织

## ◆ 现有网络中数据组织的方式

- 计算机：各种数据文件的格式
- 网络：数据包格式
- 哪些信息需要合理表示？
- 交换数据格式

## ◆ 主流音视频流的数据组织方式

- 数字音乐 CD-DA
- 音乐文件 MP3
- 数字音频广播 DAB
- 数字电视 MPEG2
- MPEG-4 AVC / H.264



# 交换数据格式

## 为什么需要统一的数据交换格式

- ◆ 计算机诞生之后，各种格式的文档被广泛用于存储信息和数据。由于计算机发展初期并没有统一的规定和标准，所以各种格式的文档层出不穷，即使是今天我们也时常能发现自己并不熟悉的扩展名代表的计算机文档。
- ◆ 计算机技术发展是如此迅猛，以至于今天领先的技术可能很快就是“明日黄花”，更何况这些文档呢？比如，我们早期使用的Word 6.0, Word 97, Word 2000的文档，谁知道10年、20年之后还有没有可以打开这些文档的工具？而对于一些特殊的文档还必须能够经历岁月，比如，档案文档、一些珍贵的多媒体文档等。这样说来，我们需要制订一个在今后相当一段时间内还能够保持一定生命力的文档类型和格式，这就是统一的交换数据格式诞生的背景。



# 交换数据格式 SGML/XML/HTML

- ◆ 标准通用标记语言(Standard Generalized Markup Language, SGML)是1986年出版发布的一个信息管理方面的国际标准。该标准定义独立于平台和应用的本文档的格式、索引和链接信息，为用户提供一种类似于语法的机制，用来定义文档的结构和指示文档结构的标签(tag)。制定SGML的基本思想是把文档的内容与样式分开。
- ◆ XML (Extensible Markup Language)是SGML的一个子集
- ◆ HTML是SGML的一种文档类型



# 交换数据格式 SGML文档的三个层次

- ◆ 结构：用一个单独的文件来描述文档的结构(文档类型定义Document Type Definition， DTD文件)。
- ◆ 内容：这里指的内容就是信息本身。内容包括信息名称(标题)、段落、项目列表和表格中的具体内容，具体的图形和声音等。
- ◆ 样式：用专门的标准来描述显示格式方面的信息，即文档样式语义学和规范语言(Document Style Semantics and Specification Language， DSSSL)。



# 交换数据格式 XML的基本语法

```
<?xml version = "1.0" encoding="GB2312" standalone = "no"?>
<!DOCTYPE 菜单 SYSTEM “test.dtd”>
<?xmlstylesheet type="text/xsl" href=“test.xsl”?>
<菜单目录>
<徽菜系列>
    <名称>平地一声雷</名称>
    <厨师>王XX</厨师>
    <饭店>金满楼</饭店>
    <价格>
        <正常价格>38</正常价格>
        <优惠价格>28</优惠价格> <!--只对VIP用户-->
    </价格>
</徽菜系列>
</菜单目录>
```

此文件中记录了所有的数据

Test.xml

{ynh,cxh}@ustc.edu.cn



# 交换数据格式 文档类型定义文件DTD

```
<?xml version="1.0" encoding="GB2312"?>
<!ELEMENT 菜单目录 (徽菜系列)*>
<!ELEMENT 科技书籍 (名称,厨师,饭店,价格)>
<!ELEMENT 价格 (正常价格,优惠价格)>
<!ELEMENT 名称 (#PCDATA)>
<!ELEMENT 厨师 (#PCDATA)>
<!ELEMENT 饭店 (#PCDATA)>
<!ELEMENT 正常价格 (#PCDATA)>
<!ELEMENT 优惠价格 (#PCDATA)>
```

此文件中记录了结构信息

**Test.dtd**

{ynh,cxh}@ustc.edu.cn



# 交换数据格式 XML数据的显示控制

```
<xsl:template match = "菜单目录">
<xsl:for-each select="徽菜系列">
    <UL>
        <LI><xsl:value-of select="名称"/></LI>
        <LI>厨师:<xsl:value-of select="厨师"/></LI>
        <LI>饭店: <xsl:value-of select="饭店"/></LI>
        <LI>正常价格: <xsl:value-of select="价格/正常价格"/></LI>
        <LI>优惠价格: <xsl:value-of select="价格/优惠价格"/></LI>
    </UL>
</xsl:for-each>
</xsl:template>
```

此文件中记录  
了显示格式

Test.xls

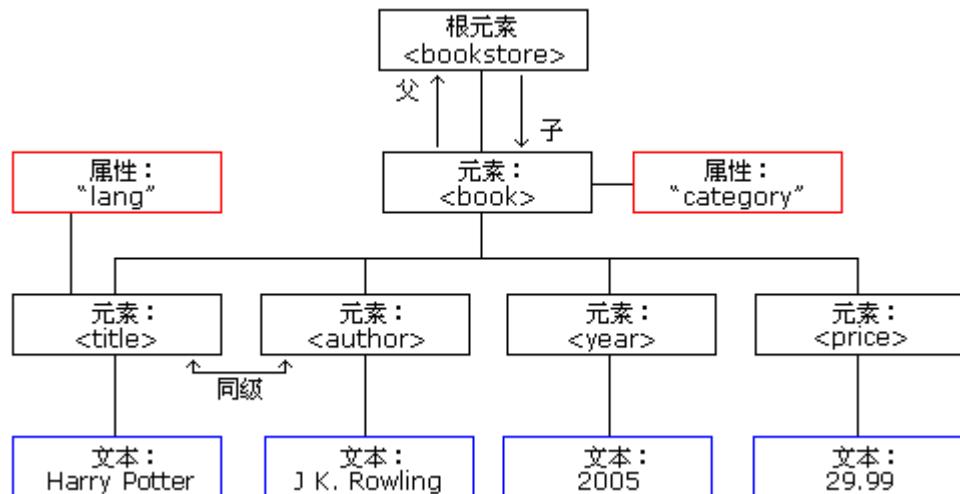
{ynh,cxh}@ustc.edu.cn



# 交换数据格式

## XML表示树形结构的数据示例

```
<bookstore> Future Publisher
<book category="COOKING">
<title lang="en">Everyday Italian</title>
<author>Giada De Laurentiis</author>
<year>2005</year>
<price>30.00</price>
</book>
<book category="CHILDREN">
<title lang="en">Harry Potter</title>
<author>J K. Rowling</author>
<year>2005</year>
<price>29.99</price>
</book>
<book category="WEB">
<title lang="en">Learning XML</title>
<author>Erik T. Ray</author>
<year>2003</year>
<price>39.95</price>
</book>
</bookstore>
```





# 交换数据格式

## JSON(JavaScript Object Notation)

- ◆ XML(eXtensible Markup Language) 在1998年发布为W3C的标准（XML1.0）。
- ◆ JSON(JavaScript Object Notation) 是一种轻量级的数据交换格式。它基于JavaScript（Standard ECMA-262 3rd Edition - December 1999）的一个子集。
- ◆ JSON采用完全独立于语言的文本格式，但是也使用了类似于C语言家族的习惯（包括C, C++, C#, Java, JavaScript, Perl, Python等）。这些特性使JSON成为理想的数据交换语言。易于人阅读和编写，同时也易于机器解析和生成。



# 交换数据格式 XML与JSON对比示例

```
1  <?xmlversion="1.0"encoding="utf-8"?>
2  <country>
3  <name>中国</name>
4  <province>
5  <name>黑龙江</name>
6  <cities>
7  <city>哈尔滨</city>
8  <city>大庆</city>
9  </cities>
10 </province>
11 <province>
12 <name>广东</name>
13 <cities>
14 <city>广州</city>
15 <city>深圳</city>
16 <city>珠海</city>
17 </cities>
18 </province>
19 <province>
20 <name>台湾</name>
21 <cities>
22 <city>台北</city>
23 <city>高雄</city>
24 </cities>
25 </province>
26 <province>
27 <name>新疆</name>
28 <cities>
29 <city>乌鲁木齐</city>
30 </cities>
31 </province>
32 </country>
```

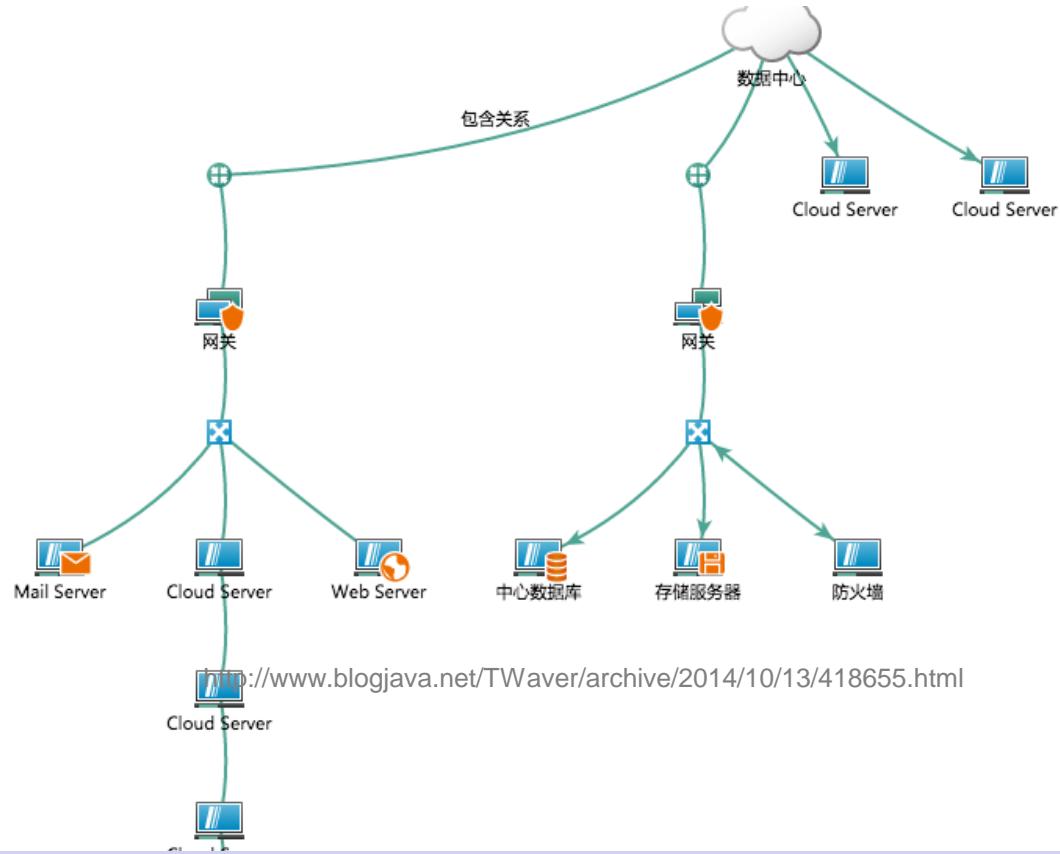
编码的可读性，**XML**有优势，人类的语言更贴近这样的说明结构。**JSON**读起来更像一个数据块，比较费解了，但适合机器阅读。

**XML**可在记录信息数据项的同时记录信息的结构信息，显示时的格式信息。**JSON**则仅仅是一个数据的集合。

```
1  {
2  "name": "中国",
3  "province": [
4  {
5  "name": "黑龙江",
6  "cities": {
7  "city": ["哈尔滨", "大庆"]
8  }
9  },
10 {
11 "name": "广东",
12 "cities": {
13 "city": ["广州", "深圳", "珠海"]
14 }
15 },
16 {
17 "name": "台湾",
18 "cities": {
19 "city": ["台北", "高雄"]
20 }
21 },
22 {
23 "name": "新疆",
24 "cities": {
25 "city": ["乌鲁木齐"]
26 }
27 }
28 ]
29 }
```



# 交换数据格式 JSON表示拓扑信息示例



```
var topo_data = [
    {"element": "node", "name": "网关", "id": "gateway1", "image": "group", "icon": "icon_wall"},  

    {"element": "node", "name": "网关", "id": "gateway2", "image": "subnetword", "icon": "icon_wall"},  

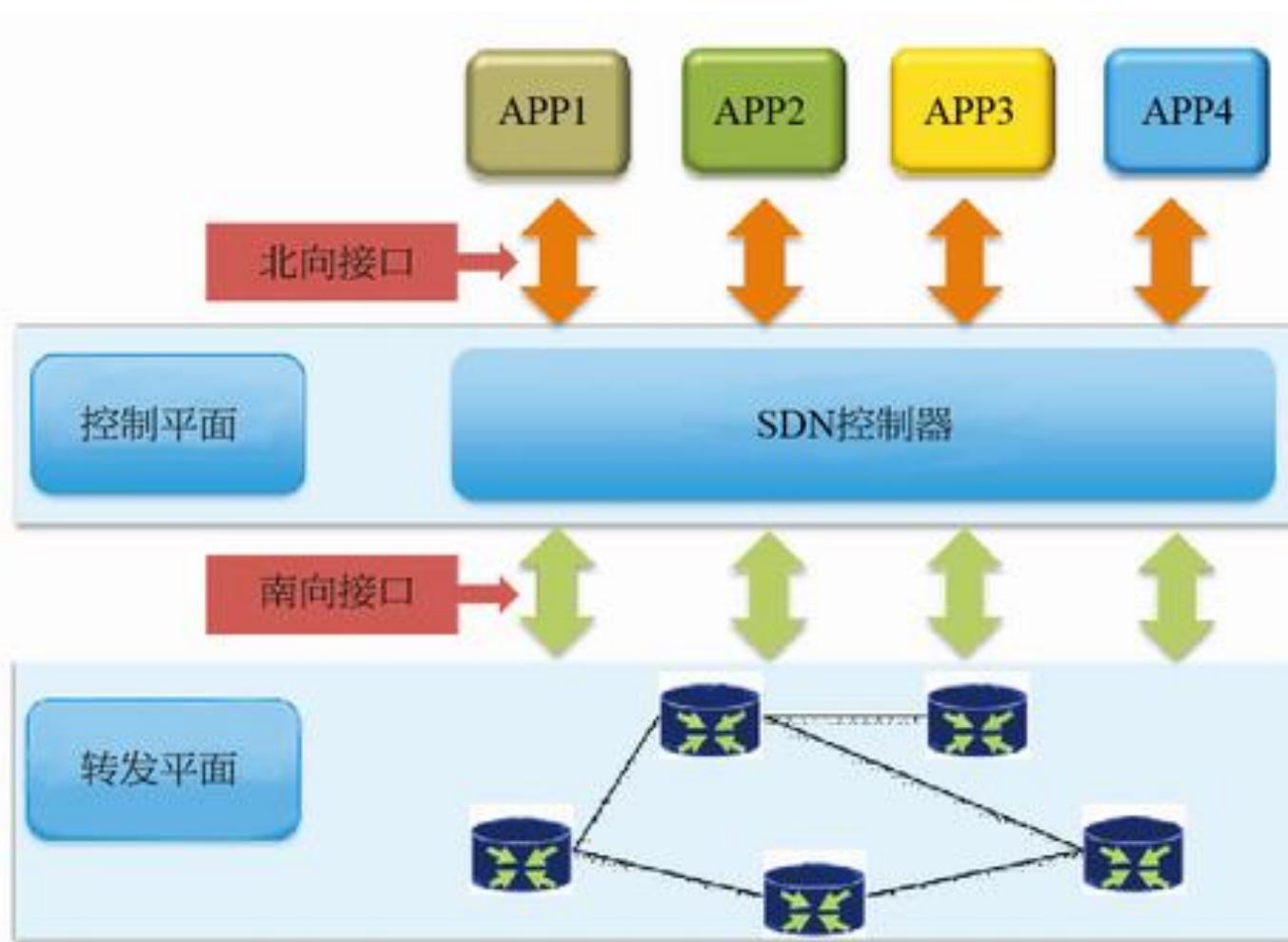
    {"element": "link", "from": "cloud", "to": "center1", "name": "包含关系"},  

    {"element": "link", "from": "gather2", "to": "firewall", "arrow": "11"}  

];
```



# 小结：交换数据格式





# 裸数据的组织

## ◆ 现有网络中数据组织的方式

数据如何组织是一个非常重要的问题

## ◆ 主流音视频流的数据组织方式

- 数字音乐 CD-DA
- 音乐文件 MP3
- 数字音频广播 DAB
- 数字电视 MPEG2
- MPEG-4 AVC / H.264



# 专题线索：多媒体数据的30年

## ◆ CD-DA

- 数字音乐的光存储， CD-DA
- 数据的光存储， CDROM
- 数字视频的光存储， VCD

## ◆ MP3

- 随着因特网发展而流行

## ◆ DAB

## ◆ MPEG2 PS / TS (HDTV)

- 随着DVD出现开始流行

## ◆ MPEG-4 AVC / H.264

- 随着下一代DVD的讨论而诞生

## ◆ HEVC / H.265

**1980's**

**1990's**

**2000**

**2010**



# 裸数据的组织

- ◆ 现有网络中数据组织的方式
- ◆ 主流音视频流的数据组织方式

- 数字音乐 CD-DA

- 各种光盘的通道编码
    - CD-DA的数据帧组织
    - VCD与DVD的数据组织

- 音乐文件 MP3

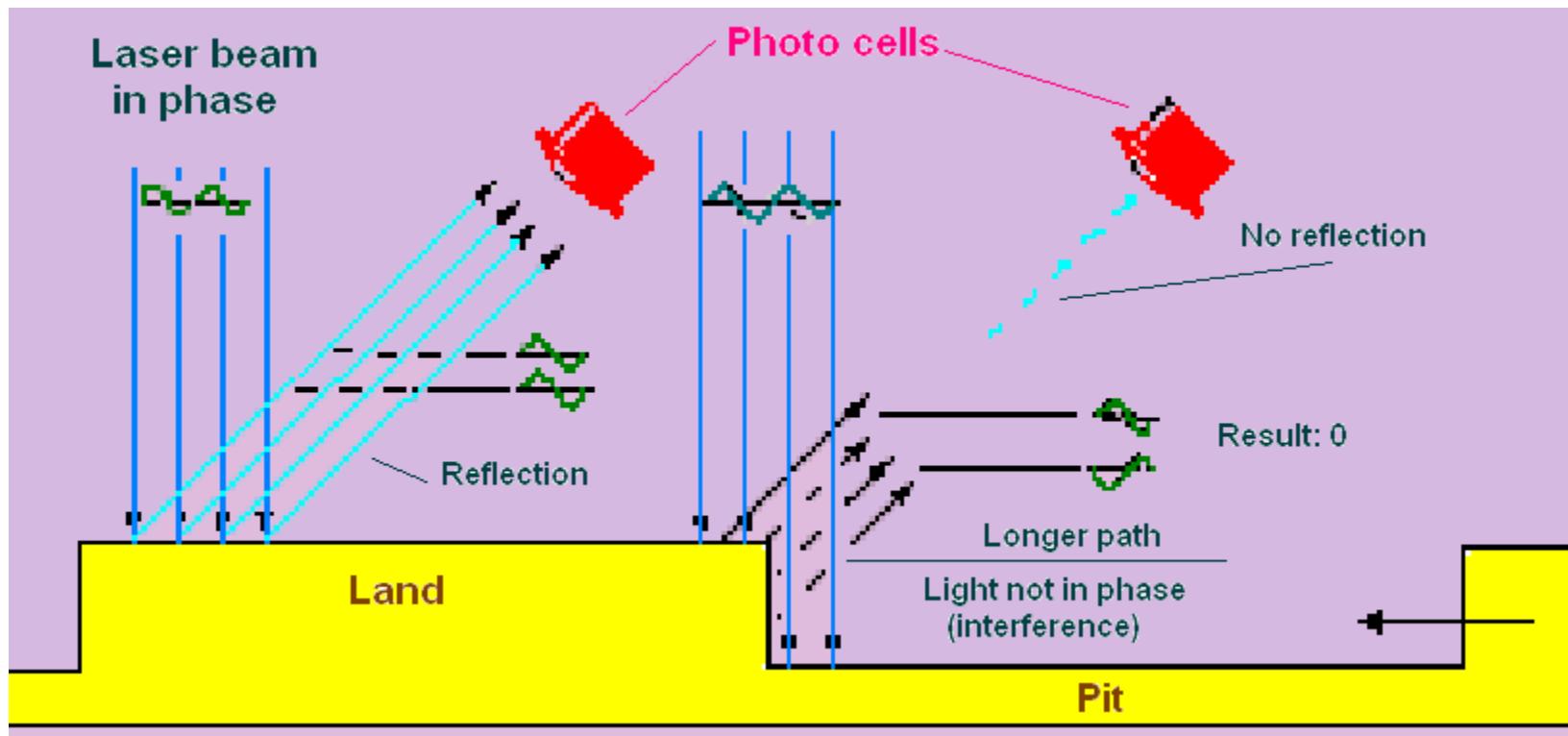
- 数字音频广播 DAB

- 数字电视 MPEG2

- MPEG-4 AVC / H.264

# Interaction of laser beam with pits and lands

- ◆ a binary **one** is stored on the disc as a **change from a land to a pit** or a pit to a land, while a binary **zero** is indicated by **no change**



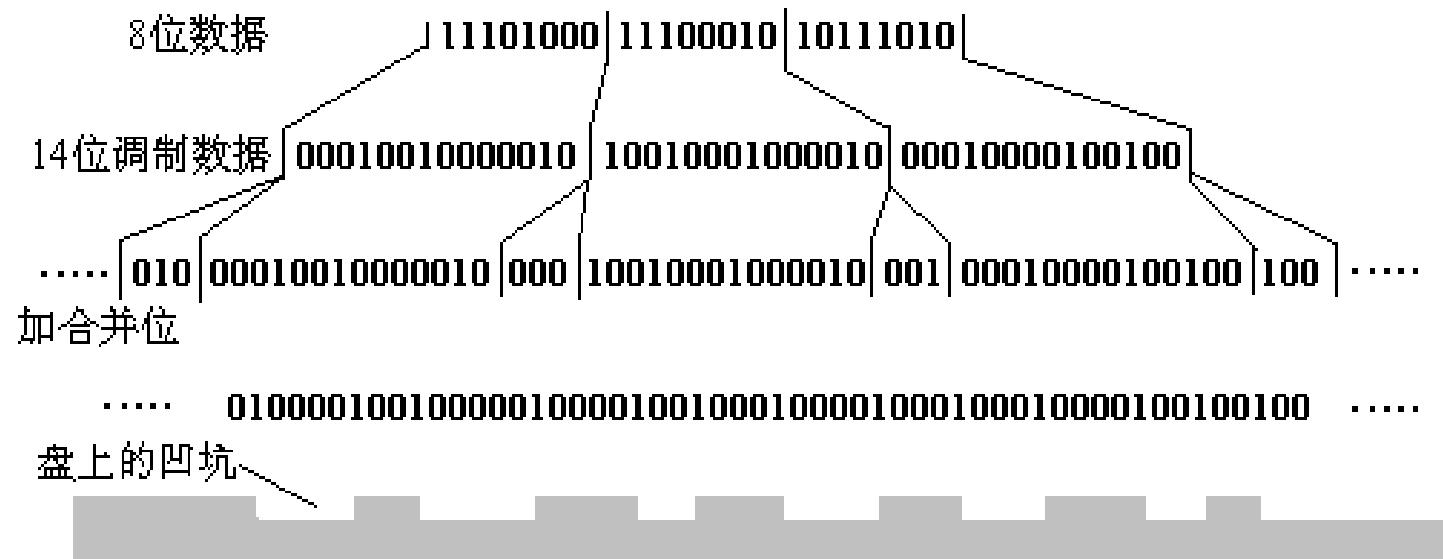


# 通道编码

- ◆ 激光唱盘使用的通道编码叫做8到14比特调制编码(eight to fourteen modulation, EFM)。这种编码的含义就是把一个8个比特(即1个字节)的数据用14比特来表示。
- ◆ 在数字记录中要做通道编码的主要原因有两个，一是为了改善读出信号的质量，二是为了在记录信号中提取同步信号。例如，有连续多个字节的全“0”信号或者全“1”信号要记录到盘上，如果不作通道编码就把它们记录到盘上，读出时的输出信号就是一条直线，电子线路就很难区分有多少个“0”或者多少个“1”信号。而对于没有规律的数字信号，读出时的信号幅度和频率的变化范围都很大，电子线路很难把“0”和“1”区分开，读出的信息就很不可靠。因此通俗说来，通道编码实际上就是要在连续的“0”插入若干个“1”，而在连续的“1”之间插入若干个“0”，并对“0”和“1”的连续长度数目即“行[游]程长度”加以限制。



◆根据70年代(?)的技术水平，把“0”的游程长度最短限制在2个，而最长限制在10，光盘上的信号就能够可靠读出。故将8bit数据映射到14bit的通道码。当通道码合并时为了满足游程长度的要求，在通道码之间再增加了3bit来确保读出信号的可靠性。





# “凹凸”的最大最小长度

- ◆ Because EFM ensures there are at least 2 zeroes between every 2 ones, it is guaranteed that **every pit and land is at least three bit clock cycles long**. This property is very useful since it reduces the demands on the optical pickup used in the playback mechanism.
- ◆ The ten consecutive-zero maximum ensures **worst-case clock recovery** in the player.



# 专利情况

## ◆ EFM

### United States Patent [19]

Immink et al.

[11] Patent Number: 4,501,000

[45] Date of Patent: Feb. 19, 1985

[75] Inventors: **Kornelis A. Immink; Jakob G. Nijboer**, both of Eindhoven, Netherlands; **Hiroshi Ogawa; Kentaro Odaka**, both of Tokyo, Japan

[73] Assignee: **Sony Corporation**, Tokyo, Japan



US005696505A

## ◆ EFMPlus

### United States Patent [19]

Schouhamer Immink

[11] Patent Number: 5,696,505

[45] Date of Patent: Dec. 9, 1997

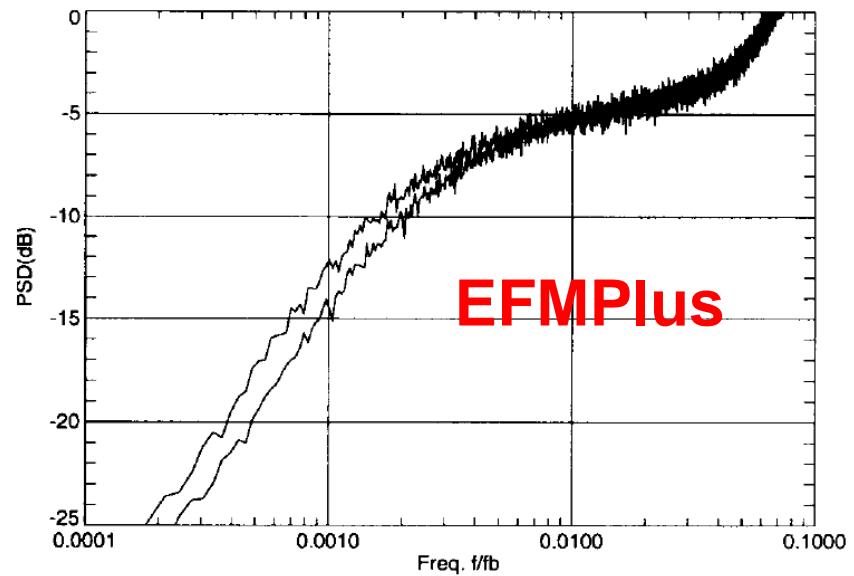
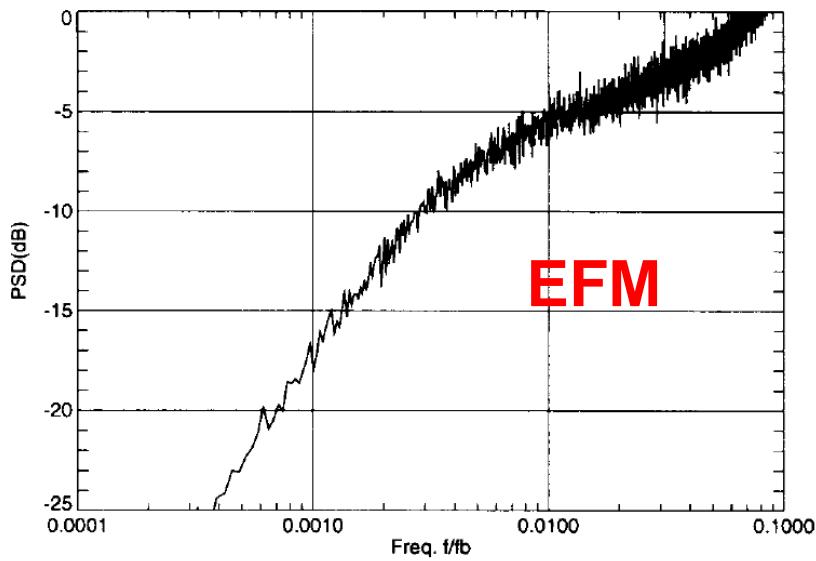
[75] Inventor: **Kornelis A. Schouhamer Immink**, Eindhoven, Netherlands

[73] Assignee: **U.S. Philips Corporation**, Tarrytown, N.Y.

{ynh,cxh}@ustc.edu.cn



# PSD

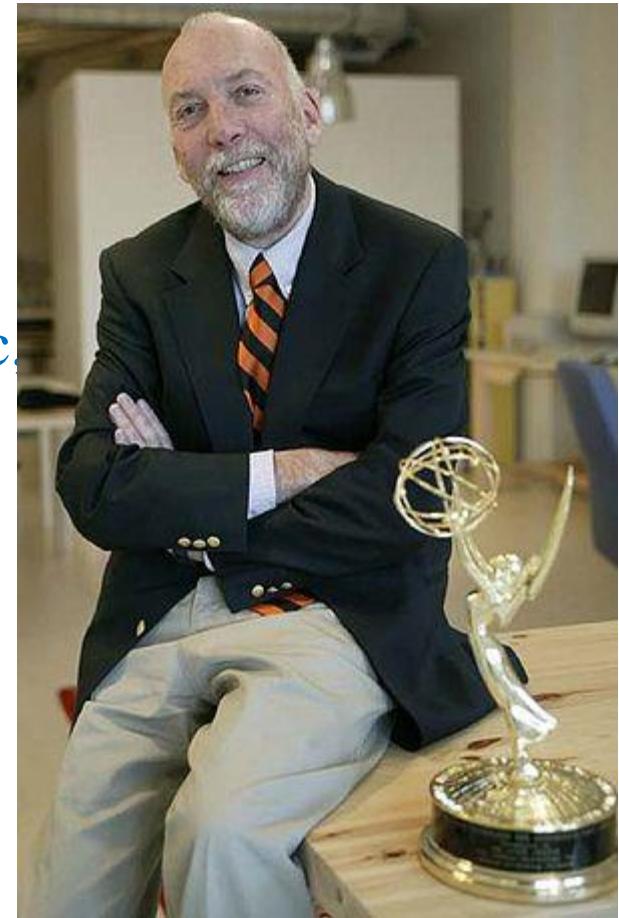


编码后比特序列在频率 $f=0$ 处无信号分量



# Kees Schouhamer Immink

Kees Schouhamer Immink (born 18 December 1946 in Rotterdam) is a Dutch scientist, inventor, and entrepreneur, who pioneered and advanced the era of digital audio, video, and data recording including popular digital media such as **Compact Disc**, **DVD** and **Blu-Ray Disc**. He has been a prolific and influential engineer, who holds more than 1100 US and international patents. The impact of his work on consumer electronics is so large **that it is virtually impossible to enjoy digital audio or video that does not reflect his work.**





# DC-balanced or DC-Free Code

- ◆ 数字基带信号包含直流分量或低频分量，那么对于一些具有电容耦合电路的设备或者传输频带低端受限的信道（广义信道），信号将可能传不过去。
- ◆ 数字基带信号传输码型的要求
  - 有利于提高系统的频带利用率
  - 应不含直流分量、低频分量要尽量少
  - 考虑到码型频谱中高频分量的影响
  - 基带信号应具有足够大的定时信号供提取
  - 基带信号的传输码型应具有误码检测能力
  - 码型变换设备简单，容易实现



# Run Length Limited (RLL) Coding

- ◆ **有限游程长度码**是由两个参数定义的二进制序列， $T_{min}=(d+1)$ 和 $T_{max}=(k+1)$ ，分别代表**最小游程长度**和**最大游程长度**。游程长度指码流中两个相邻的1之间连续0的个数。设计RLL码主要考虑两方面的限制：
- ◆ 确定d和k。d限制信号的最高跳变频率，控制相邻跳变之间的干扰；k控制最低的跳变频率，保证参考时钟的正确恢复。
- ◆ 直流平衡，即编码本身在频率 $f=0$ 处无信号分量。根据编码理论，限制编码后序列的游程数字和（RDS: Running Digital Sum）有界即可。



# 磁盘和光盘的调制编码

硬盘		MFM	Intel	1977
硬盘		RLL (2,7)	IBM	1979
CD	EFM	RLL(2,10)	SONY	1985
DVD	EFMPlus	RLL(2,10)	Philip	1997
Blu-ray	17PP	RLL(1,7)	SONY	2007

MFM encoding can be thought of as having data bits separated by clock bits. The basic encoding rule is that  $(x, y, z, \dots)$  encodes to  $(x, x \text{ NOR } y, y, y \text{ NOR } z, z, z \text{ NOR } \dots)$ . On average this means that each data bit is encoded as two bits on disk, but some delimiters are required at the beginning and end of a sequence, so this limit is never quite reached in practice.

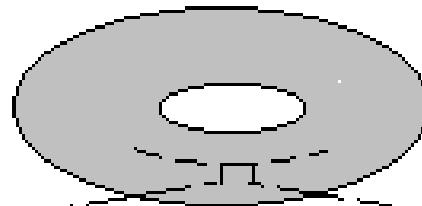


# CD-DA数据的基本格式

- ◆ CD上有一条螺旋型的物理光道，可以分成多个逻辑光道。
- ◆ 一条逻辑光道由许多扇区(sector)组成，一扇区由98帧(frame)组成。
- ◆ 帧是激光唱盘上存放声音数据的基本单元。
- ◆ 通常一首歌曲安排在一个逻辑光道上

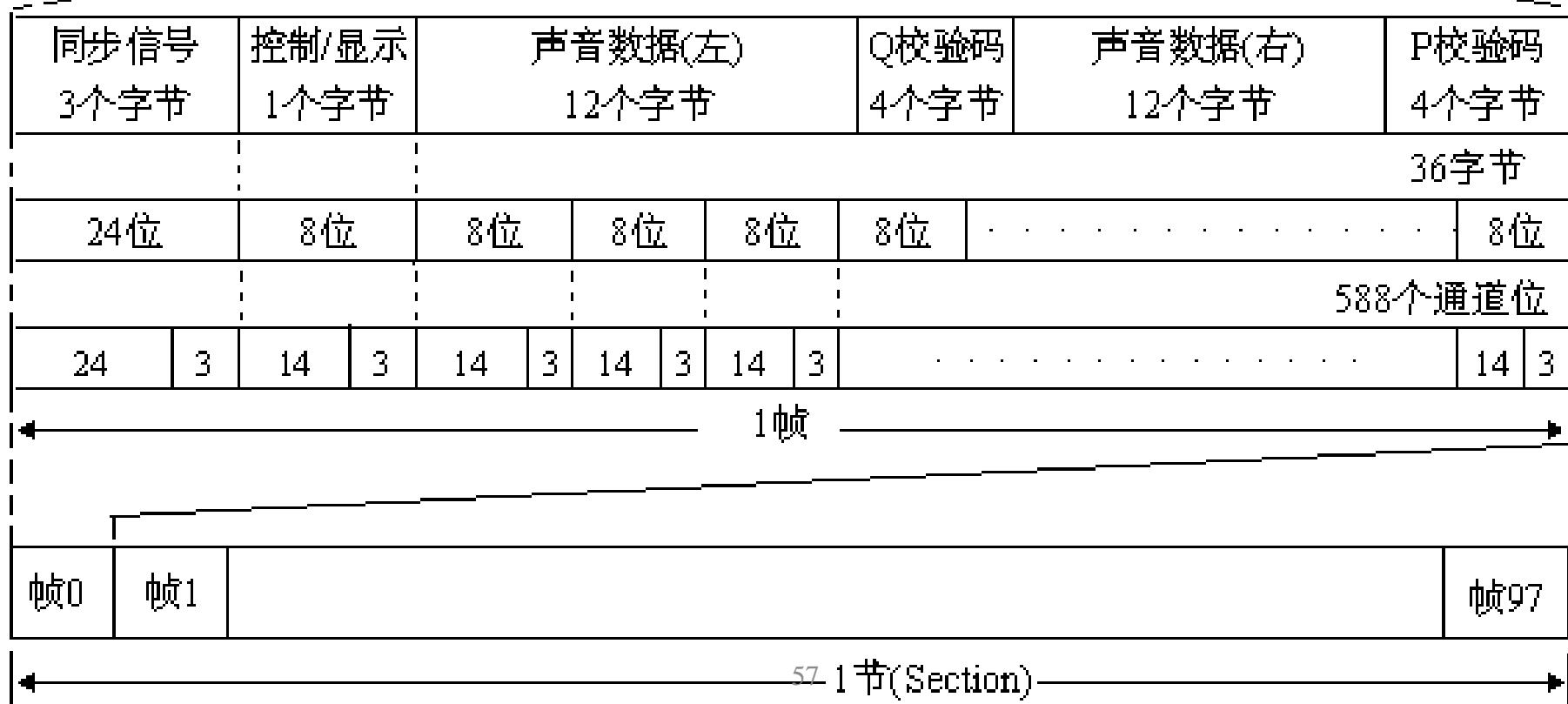


# 扇区中的一个帧



$$(12+12)*98=2352$$

$$2352/(3234/8*17)=34.2246\%$$





# 小结：CD-DA的数据组织

## ◆ 数字基带信号的编码

- EFM、EFMPlus、17PP
- Run Length Limited (RLL) Coding
- 硬盘、CD-DA、DVD、Blu-Ray

## ◆ CD-DA的数据组织

- 物理光道→逻辑光道→扇区→帧
- 单个帧：同步、控制、数据、校验
- CD-DA中比特效率约34%



# 从CD-DA → CDROM

## ◆ CD-DA 光道

- 一个扇区(sector)2352字节

## ◆ CD-ROM Mode 1 光道

4字节的扇区地址称为HEADER			
分(MIN) 1字节 0~74	秒(SEC) 1字节 0~59	分秒(FRAC) 1字节 0~74	方式.Mode) 1字节 01

2352字节

同步字节 12字节	扇区地址 4字节	用户数据 2048字节	EDC 4字节	未用 8字节	ECC 276字节
--------------	-------------	----------------	------------	-----------	--------------

## ◆ CD-ROM Mode 2 光道

2352字节

同步字节 12字节	扇区地址 4字节	用户数据 2336字节
--------------	-------------	----------------



# 物理格式与逻辑格式的分离是一次伟大的进步 ISO 9660

- ◆ 仅有物理格式标准化还不够，还需要有一个如何把文件和文件目录放到CD-ROM盘上的逻辑格式标准，也就是文件格式格式。
- ◆ ISO 9660, also referred to as CDFS (**Compact Disc File System**) by some hardware and software providers, is a file system standard published by the International Organization for Standardization (ISO) for optical disc media.
- ◆ 1985年CD-ROM出现之后，所有光盘存储系统不再关注存储内容与光道的关系



# VCD(1993)光盘目录结构

## ◆\CDI

- CDI\_ALL.RTF
- CDI\_BUM.DAT
- CDI\_FONT.FNT
- CDI\_VCD.APP

## ◆\EXT

- LOT\_X.VCD
- PSD\_X.VCD

## ◆\MPEGAV

- AVSEQ01.DAT
- AVSEQ02.DAT
- .....

## ◆\SEGMENT

- ITEM0001.DAT

## ◆\VCD

- ENTRIES.VCD
- INFO.VCD
- LOT.VCD
- PSD.VCD

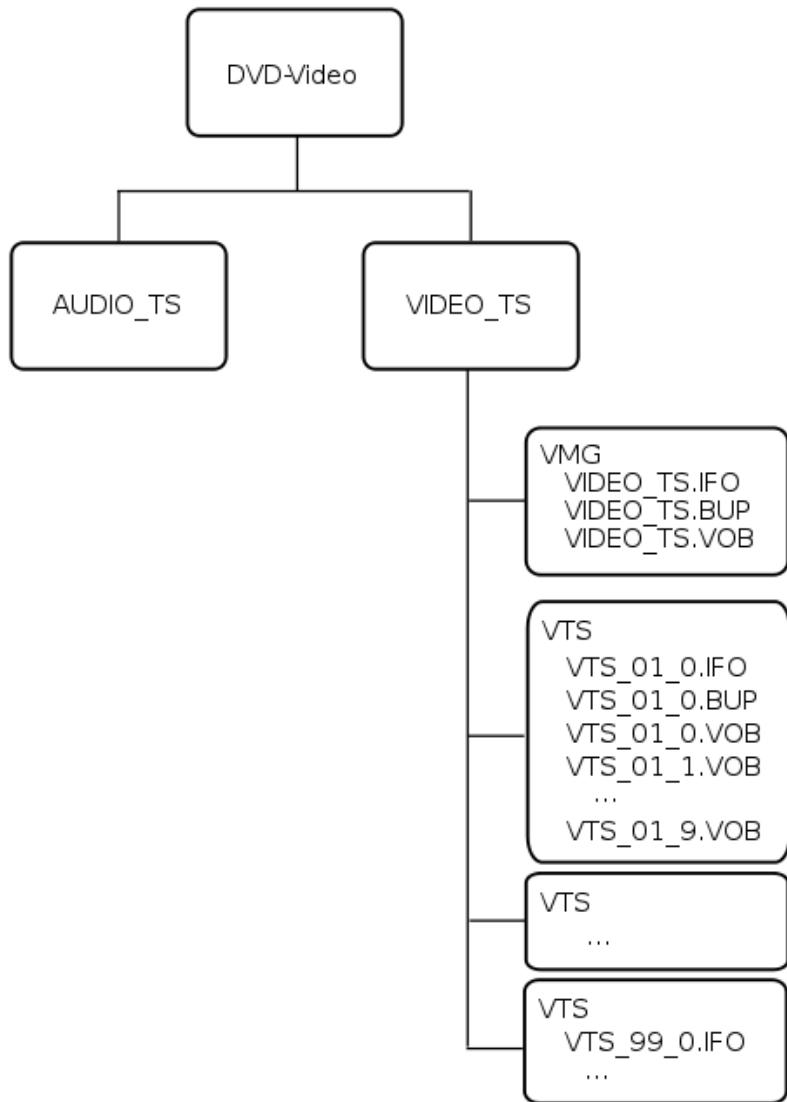


# DVD(1996)的UDF

◆ Universal Disk Format (UDF) is an implementation of the specification known as ISO/IEC 13346 and ECMA-167 and is an open vendor-neutral file system for computer data storage for a broad range of media. In practice, it has been most widely used for DVDs and newer optical disc formats, supplanting ISO 9660. Due to its design, it is very well suited for incremental updates on both recordable or (re)writable optical media. UDF is developed and maintained by the Optical Storage Technology Association (OSTA).



# DVD光盘内容



◆ 三种类型的文件：.VOB、.IFO 和.BUP文件

- .VOB文件用来保存所有MPEG-2或MPEG-1格式的音视频数据。这些数据不仅包含影片本身，而且还有供菜单和按钮用的画面以及多种字幕的子画面流。
- .IFO文件则是控制.VOB文件播放的，这个文件中可以找到有关怎么样以及何时播放.VOB文件中数据的控制信息。由于.IFO文件对于保证光盘的正常播放是至关重要的，因此.IFO文件的副本保存在.BUP文件中。



# 小结：从CD-DA看光存储音视频的数据组织

## ◆ CD-DA

- 只能存放音频；数字音频的采样率固定；样本精度固定；声道数目固定 ← 无任何扩展余地

同步信号 3个字节	控制/显示 1个字节	声音数据(左) 12个字节	Q校验码 4个字节	声音数据(右) 12个字节	P校验码 4个字节
--------------	---------------	------------------	--------------	------------------	--------------

## ◆ CD-ROM/VCD/DVD

- 光盘的物理格式和逻辑格式分离；音视频数据的组织表现为光盘内文件和目录的组织。 ← 无限扩展能力

## ◆ 网络中的分层？

- 光盘存储是点到点的链路，只需要L1+L2+应用层？



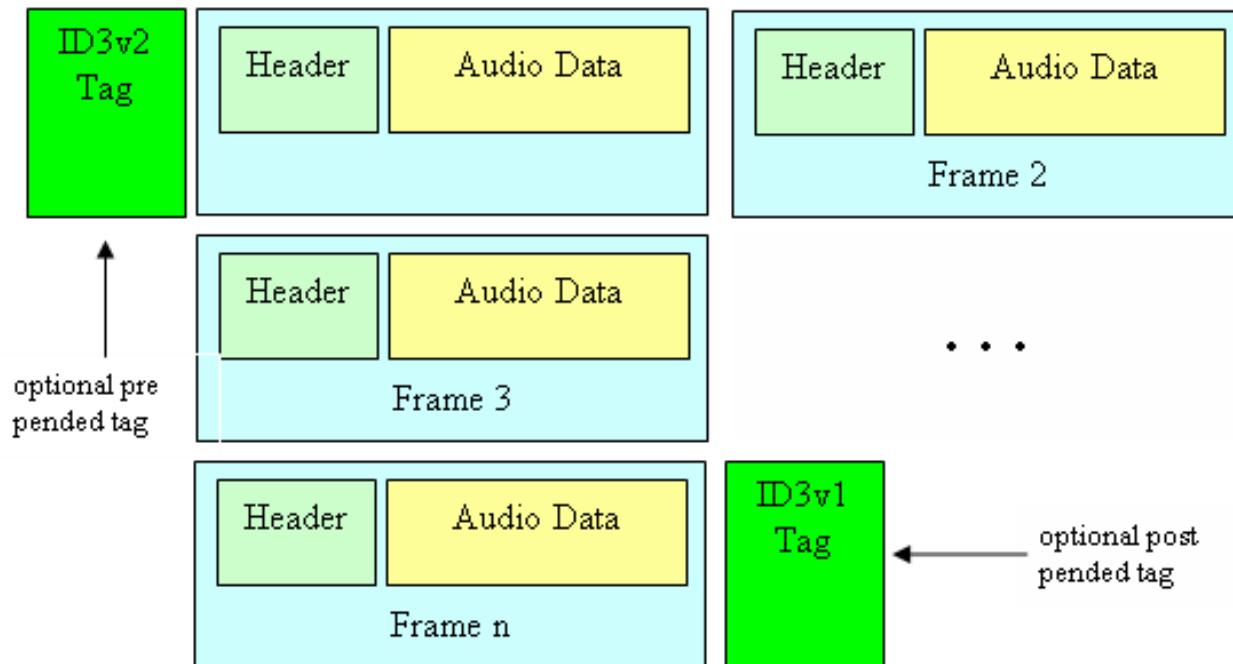
# 裸数据的组织

- ◆ 现有网络中数据组织的方式
- ◆ 主流音视频流的数据组织方式
  - 数字音乐 CD-DA
  - 音乐文件 MP3
    - Tag
    - Frame: Header + Audio Data
  - 数字音频广播 DAB
  - 数字电视 MPEG2
  - MPEG-4 AVC / H.264



# MP3文件结构： Tag+Frame

- ◆ Tag： MP3除了保存声音信息外，还可以同时保存一些和曲目相关的文本信息，比如歌名、演唱者、专辑名等等，这些信息就称为tag（标签）信息。
- ◆ Frame： 用于存放声音数据。





# 单个MP3文件可嵌入多种Tag

## MPEG Audio Header

Frame 1

|<|<|>|>|

MPEG 1 Layer III

128 kbps, Stereo

Sampling Rate: 48000 Hz

Copyright: No, CRC: Yes, Private: No,  
Original: Yes, Emphasis: None

Beginning of header: Byte 2104

Computed Frame Size: 384 Byte

Padding Size: 0 Byte

Samples Per Frame: 1152

## VBR Header

XING

Offset: Byte 2144

Number Of Frames: 210

File Size: 71496 Byte (69 KB)

Quality: 78

## Tags

ID3 V1.00: Position: Byte 73832

Size: 128 Byte

Lyrics3 V2.00: Position: Byte 73756

Size: 76 Byte

APE V2.00: Position: Byte 73600

Size: 156 Byte

ID3 V2.30: Position: Byte 0

Size: 2104 Byte



# 嵌入音频以外的数据：Lyrics

[00:05.50]歌曲:姑娘我爱你

[00:10.51]原唱:索朗扎西

[00:15.51]翻唱: 紫陌赤枫

[00:20.51]

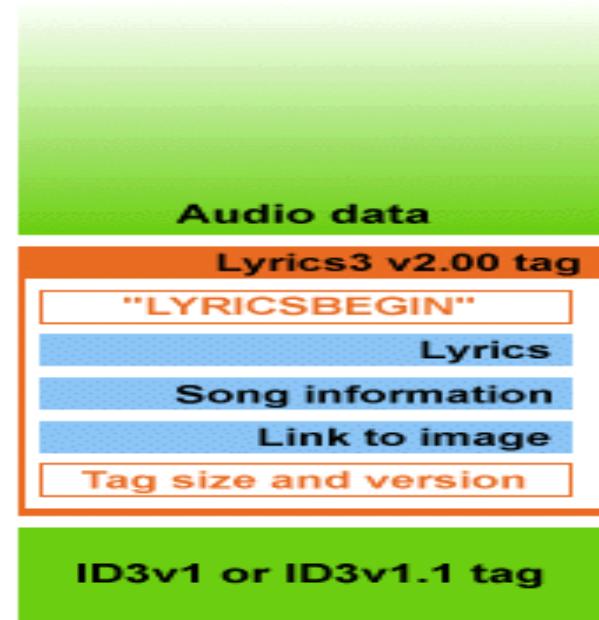
[00:42.31]长长的头发 黑黑的眼睛

[00:47.49]好象在什么地方见过你

[00:53.04]山上的格桑花开的好美丽

[00:58.27]我要摘一朵 亲手送给你

[01:04.67]





# MP3 Tag

## ◆ 比较流行的tag格式有**ID3**和**APE**

- Donald Ray Moore, Jr., in 1996 , 在MP3文件尾增加一块用于存放歌曲的说明信息，形成了ID3标准，至今已制定出ID3 V1.0, V1.1, V2.0, V2.3和V2.4标准
- APE也是一种tag格式， Frank Klemm 提出，在Musepack 音乐格式中应用

## ◆ 其他tag格式

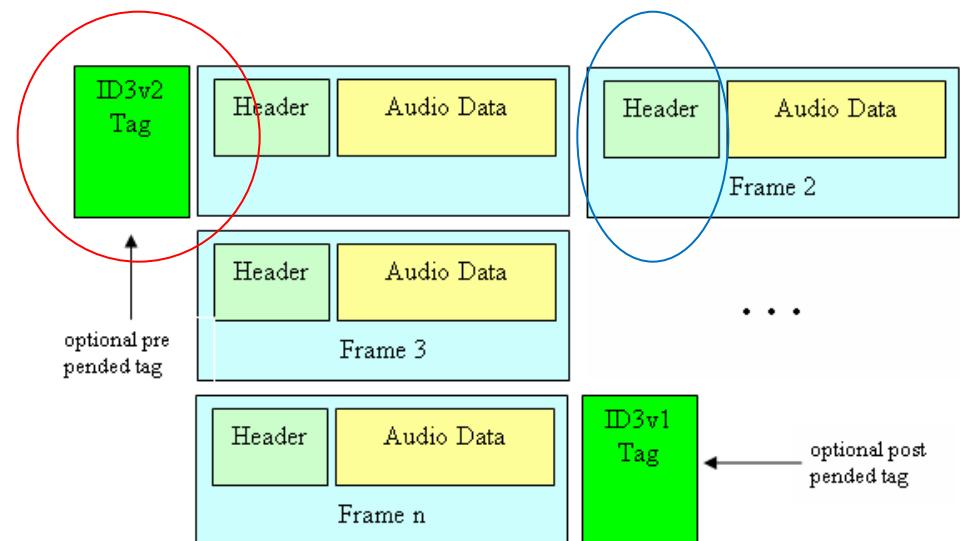
- Ogg有自定义的tag格式，叫Vorbis Comment
- WMA的tag系统是用微软ASF的
- MP4用的是Adobe 的Extensible Metadata Platform (XMP)



# MPEG Audio Tag ID3v1 (128bytes)

AAABBBBB BBBBBBBB BBBBBBBB BBBBBBBB  
BCCCCCCC CCCCCCCC CCCCCCCC CCCCCCCD  
DDDDDDDD DDDDDDDD DDDDDDDD DDDDEEEE  
FFFFFFF FFFFFFFF FFFFFFFF FFFFFFFG

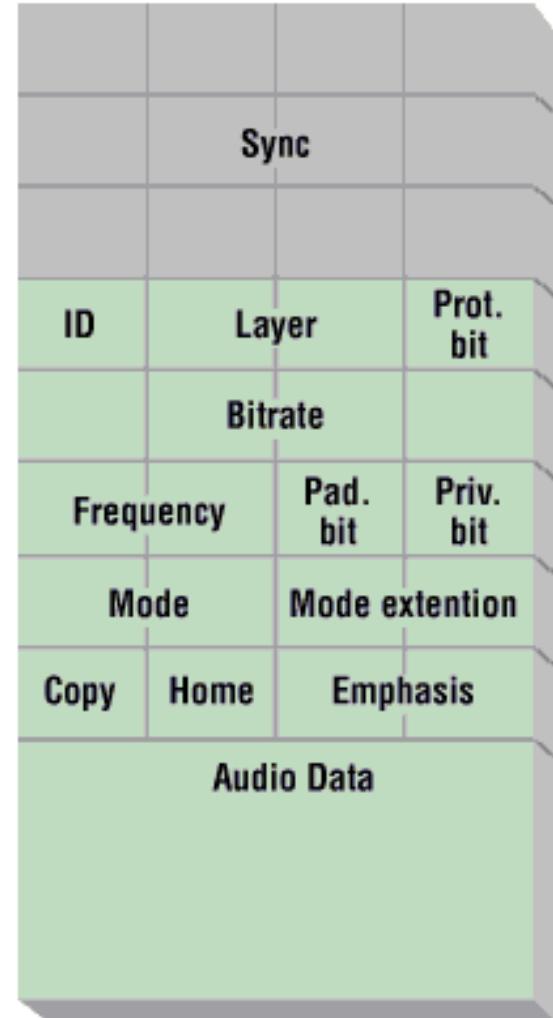
Sign	Length	Position (bytes)	Description (bytes)
A	3	(0-2)	Tag identification. Must contain 'TAG' if tag exists and is correct.
B	30	(3-32)	Title
C	30	(33-62)	Artist
D	30	(63-92)	Album
E	4	(93-96)	Year
F	30	(97-126)	Comment
G	1	(127)	Genre





# MP3 Header (32bits)

```
typedef FrameHeader {  
    unsigned int sync: 11;           //同步信息  
    unsigned int version: 2;         //版本  
    unsigned int layer: 2;          //层  
    unsigned int error protection: 1; // CRC校验  
    unsigned int bitrate_index: 4;   //位率  
    unsigned int sampling_frequency: 2; //采样频率  
    unsigned int padding: 1;          //帧长调节  
    unsigned int private: 1;          //保留字  
    unsigned int mode: 2;             //声道模式  
    unsigned int mode extension: 2;   //扩充模式  
    unsigned int copyright: 1;        // 版权  
    unsigned int original: 1;         //原版标志  
    unsigned int emphasis: 2;         //强调模式  
}HEADER, *LPHEADER;
```



缺少什么信息？



# MP3音乐的音量自动控制

- ◆ Since volume levels of different audio sources can vary greatly, due to the loudness war and other factors, it is sometimes desirable to adjust the playback volume of audio files such that a consistent average loudness is perceived. This normalization, while similar in purpose, is distinct from dynamic range compression.
- ◆ ReplayGain is one standard for measuring and storing the loudness of an MP3 file in its metadata tag, enabling a ReplayGain-compliant player to automatically adjust the overall playback volume for each file.
- ◆ David Robinson(PhD, University of Essex) in 2001.
- ◆ 1993: MP3 → 1996: TAG → 2001: ReplayGain



# 从David Robinson想到Email

- ◆ 1982: RFC 822, Standard for the Format of Arpa Internet Text Messages
- ◆ 1989: RFC 1113 - Privacy enhancement for Internet electronic mail
- ◆ Base64 is a group of similar encoding schemes that represent binary data in an ASCII string format by translating it into a radix-64 representation.
- ◆ RFC1113 (3/4 Base64编码法)
  - 使用64个ASCII字符（每个字符代表6bit的数据）
  - 被编码的数据以24bits为单位，编码成4个ASCII字符
  - 编码后数据量增加约33%
- ◆ Future?
  - [http://staff.ustc.edu.cn/~network/mmt/mm\\_attach\\_mail.pdf](http://staff.ustc.edu.cn/~network/mmt/mm_attach_mail.pdf)
  - <http://en.wikipedia.org/wiki/Base64>



# MP3帧大小

**Frame Size = ( (Samples Per Frame / 8 \* Bitrate) / Sampling Rate) + Padding Size**

长度是否变化决定于**FRAMEHEADER** 的**bitrate**是否变化

	MPEG 1	MPEG 2	MPEG 2.5
Layer I	384	384	384
Layer II	1152	1152	1152
Layer III	<b>1152</b>	576	576



# 小结：MP3文件比特效率？

## ◆ MP3文件构成

- 各种类型的Tag信息

- ID3、APE、Lyrics、...

- 音频数据帧

- 32bits的帧头

- 16bits的CRC（可选）

- Padding Bits: Layer I 32bits / Layer II 8bits

- 有效的音频数据: 每帧的样本数因Layer不同而异

## ◆ 思考：一个好的Container需要什么？

- Tag信息被有效地集成到MP3数据组织结构中



# 裸数据的组织

- ◆ 现有网络中数据组织的方式
- ◆ 主流音视频流的数据组织方式
  - 数字音乐 CD-DA
  - 音乐文件 MP3
  - 数字音频广播 DAB
  - 数字电视 MPEG2
  - MPEG-4 AVC / H.264



# 数字音频的应用

## ◆ 光盘存储

□ CD-DA → VCD Audio → DVD Audio → SACD

## ◆ 网络

□ MIDI/WAV → MP3...

## ◆ 无线广播

□ AM/FM → DAB



- ◆ 数字音频广播（DAB, **Digital Audio Broadcasting**）技术是由欧共体开发的重大产业工程项目，于1995年在欧洲正式投入运行。DAB的特点是在传送具有CD质量的声音节目的同时，还可以附加传送数据业务和多媒体广播（DMB）业务。
- ◆ DAB采用COFDM调制传输方法，在每小时200公里行车速度的汽车上，可以比较满意地实现移动接收。目前国际上已形成包括DAB系统、接收机、发射机和网络、附加数据传输在内的四大系列标准，标准体系已经非常完善。

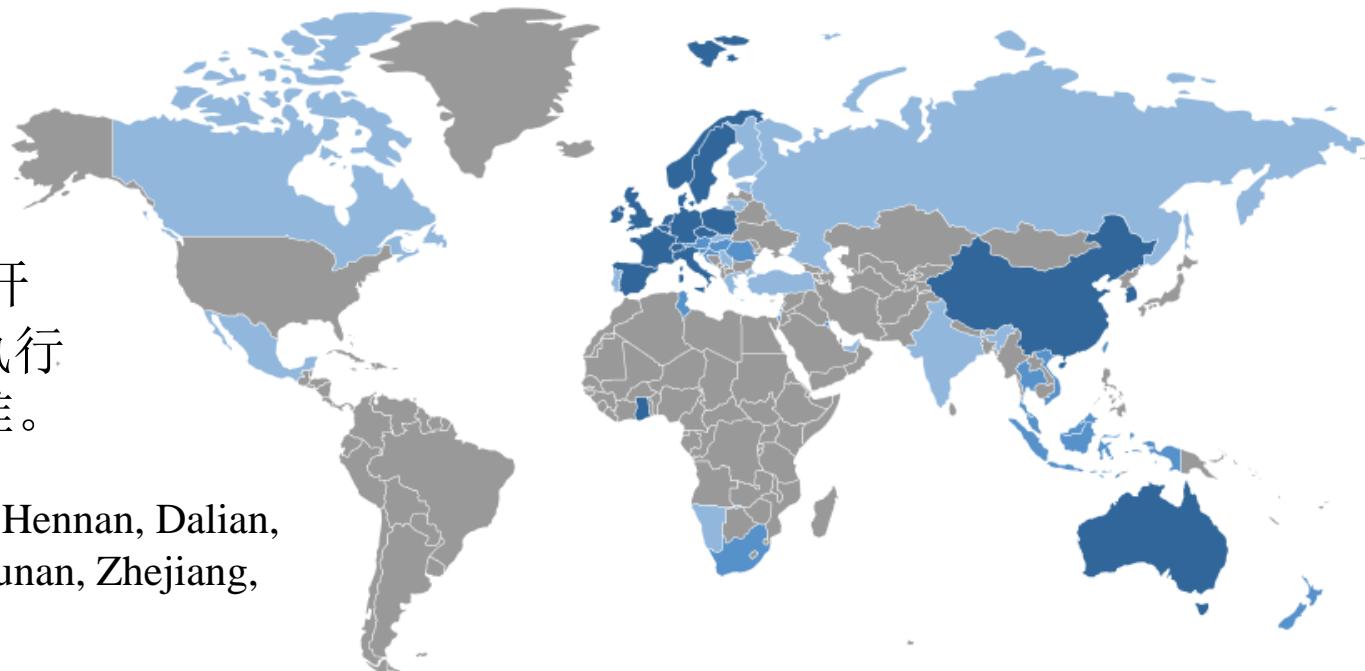


# DAB全球现状

数字音频广播（DAB, **Digital Audio Broadcasting**）技术是由欧共体开发的重大产业工程项目，于1995年在欧洲正式投入运行。DAB的特点是在传送具有CD质量的声音节目的同时，还可以**附加传送数据业务和多媒体广播（DMB）业务**。

2006年6月1日开始正式在中国执行欧洲的DAB标准。

Beijing, Guangdong, Hennan, Dalian, Yunnan, Liaoning, Hunan, Zhejiang, Anhui and Shenzhen.



Countries with regular services

Countries with trials and/or regulation

Countries with interest



# 中国DAB现状

- ◆ 中国广电部自1992年6月就通过了开展DAB重大科研的可行性报告，并与欧盟签订了DAB项目的合作规划；1995年DAB项目被列为国家“九五”重大科技产业项目，先后在北京和广东建立了DAB实验室和无线发射先导网；2006年5月18日，中国国家广电总局正式出台了《30 MHz ~ 3000 MHz地面数字音频广播系统技术规范》，欧洲的DAB标准正式成为中国数字广播行业标准，并于2006年6月1日开始正式在中国执行。
- ◆ DAB and DMB are on-air in cities and regions across the country including: Beijing, Guangdong, Hennan, Dalian, Yunnan, Liaoning, Hunan, Zhejiang, Anhui and Shenzhen.



# DAB→DAB+

- ◆ The DAB standard was initiated as a European research project in the 1980s.
  - The BBC launched the first DAB digital radio in 1995.
  - using a bit rate of 128 kbit/s or less, with the MP2 audio codec
  - The **MPEG-1 Audio Layer II** ("MP2") codec was created as part of the EU147 project.
- ◆ An upgraded version of the system was released in February 2007, which is called DAB+.
  - adopt the **AAC+** audio codec and stronger error correction coding



# MPEG1 Audio

## ◆ Layer1 1993年

- 频带相等的子带，使用频域掩蔽特性，每个子带用6bit量化。

## ◆ Layer2 1993年

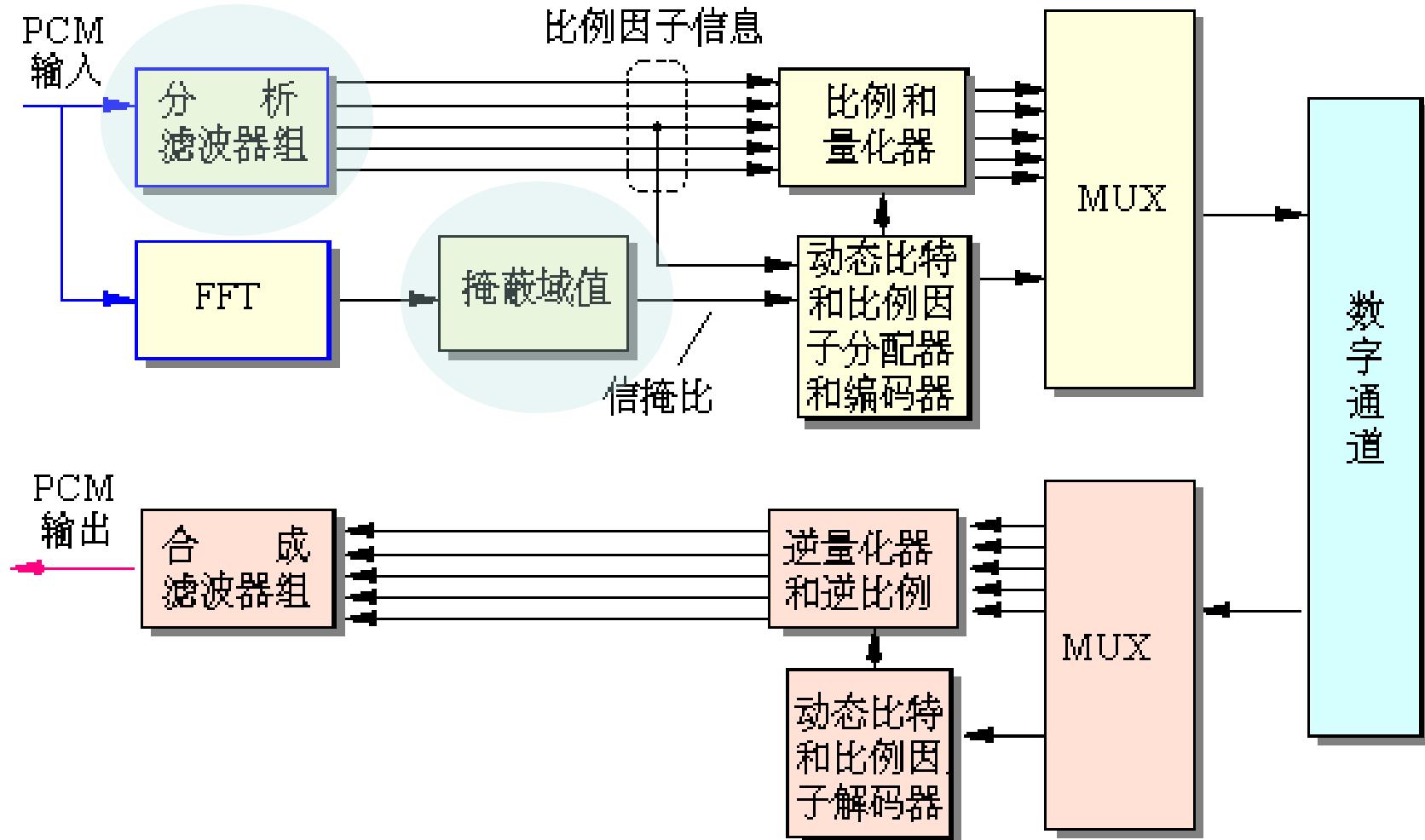
- 频带相等的子带，除了使用频域掩蔽特性之外还利用了时间掩蔽特性，低频段的子带用4比特，中频段的子带用3比特，高频段的子带用2比特。

## ◆ Layer3 1993年

- 层3使用比较好的临界频带滤波器，把音频带分成非等带宽的子带，除了使用频域掩蔽特性和时间掩蔽特性之外，还考虑了立体声数据的冗余，并且使用了霍夫曼(Huffman)编码器。还使用了MDCT(modified discrete cosine transform)把子带的输出在频域里进一步细分以达到更高的频域分辨率。



# MPEG1 audio层1和层2编解码





# 思考：MP2与MP3最本质的差异

◆ MP2 is a **sub-band** audio encoder, which means that compression takes place in the **time domain** with a low-delay filter bank producing 32 frequency domain components. By comparison, MP3 is a transform audio encoder with hybrid filter bank, which means that compression takes place in the **frequency domain** after a hybrid (double) transformation from the time domain.

◆ 1993年后的音频编码都是变换域做的

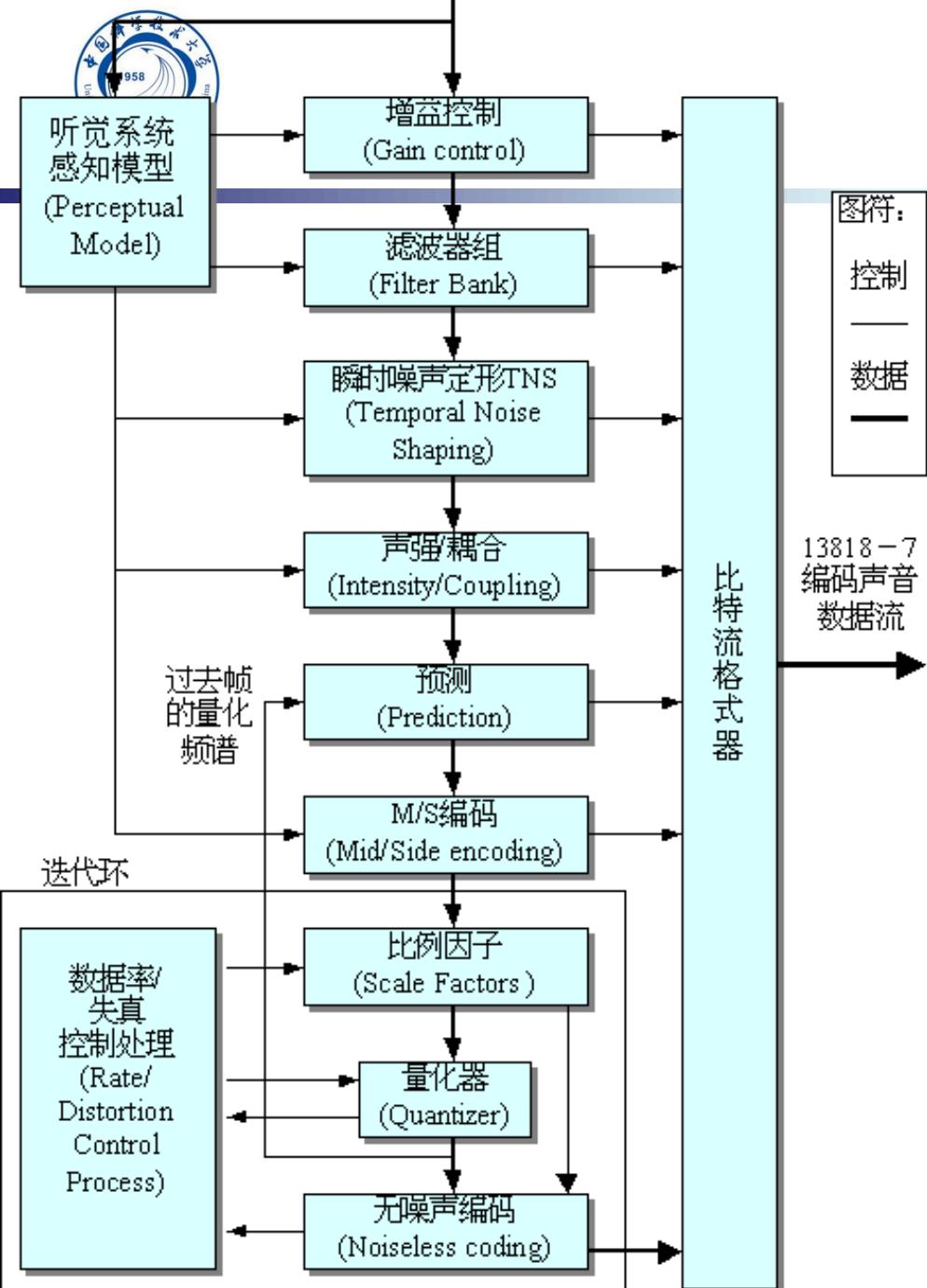


### ◆ MPEG-2 BC (Backward Compatible)

- 增加了16 kHz, 22.05 kHz和24 kHz采样频率
- 输出速率由32~384 kb/s扩展到8~640 kb/s
- 支持5.1声道和7.1声道的环绕声
- 支持Linear PCM(线性PCM)和Dolby AC-3(Audio Code Number 3)编码

### ◆ MPEG-2 AAC (Advanced Audio Coding) 1997年

- 利用掩蔽特性减少数据量，并把量化噪声分散到各个子带中，用全局信号把噪声掩蔽掉。
- 采用频率可从8 kHz到96 kHz，可支持声道数目极多



# MPEG-2 AAC编码器

## Review

- ◆ 增益控制
- ◆ 滤波器组
- MDCT/TDAC
- ◆ 瞬时噪声定形
- ◆ 声强/耦合
- 单声道信号加上位置信息
- ◆ 预测
- ◆ Mid/Side
- 左右声道转换为中央 M(middle)和边S(side)声道
- ◆ 比例因子
- ◆ 量化器
- ◆ 无噪声编码
- 霍夫曼编码



# MPEG2 AAC如何超越MP3？

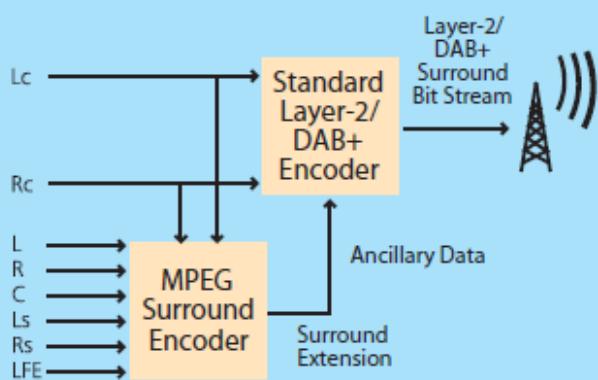
## 测试表明相同情形下AAC比MP3音质好

- ◆ More **sample frequencies** (from 8 to 96 kHz) than MP3 (16 to 48 kHz)
- ◆ Up to 48 channels (MP3 supports up to two channels in MPEG-1 mode and up to 5.1 channels in MPEG-2 mode)
- ◆ Higher efficiency and simpler filterbank (rather than MP3's hybrid coding, AAC uses a pure MDCT)
- ◆ Higher coding efficiency for stationary signals (AAC uses a **blocksize** of 1024 or 960 samples, allowing more efficient coding than MP3's 576 sample blocks)
- ◆ Higher coding accuracy for transient signals (AAC uses a blocksize of 128 or 120 samples, allowing more accurate coding than MP3's 192 sample blocks)
- ◆ Can use Kaiser-Bessel derived window function to eliminate spectral leakage at the expense of widening the main lobe
- ◆ Much better handling of audio frequencies above 16 kHz
- ◆ More flexible joint stereo
- ◆ Adds additional modules (tools) to increase compression efficiency: **TNS**, Backwards Prediction, PNS etc...

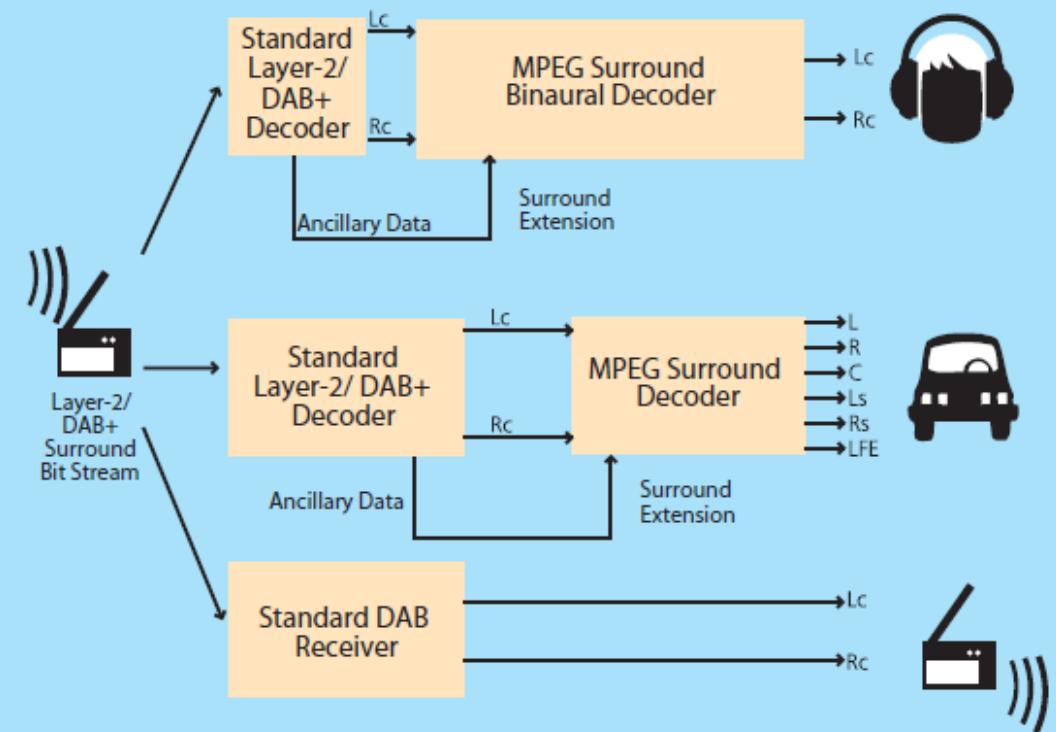


# DAB环绕声架构

DAB 5.1 Surround Broadcasting



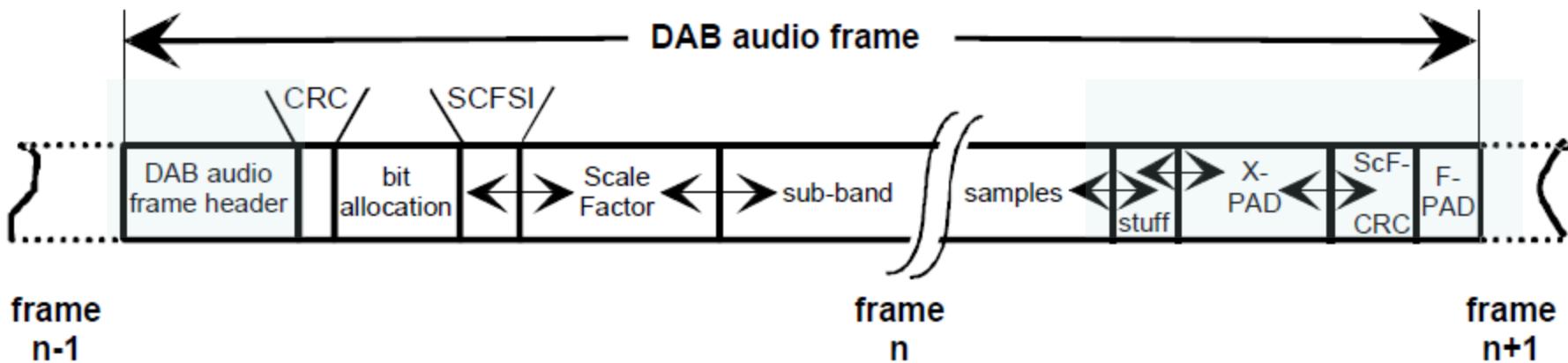
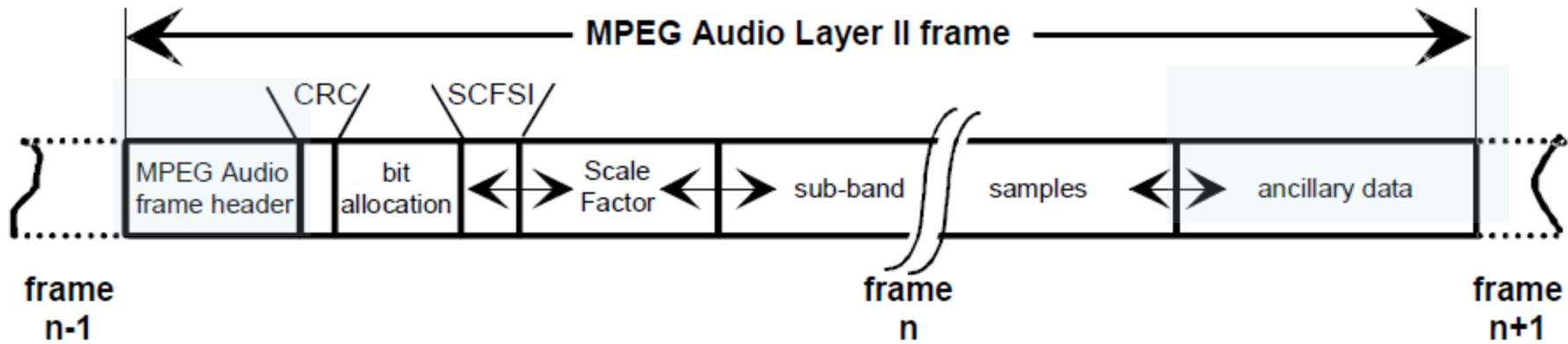
DAB 5.1 Surround Receiver





# DAB帧与MP2帧

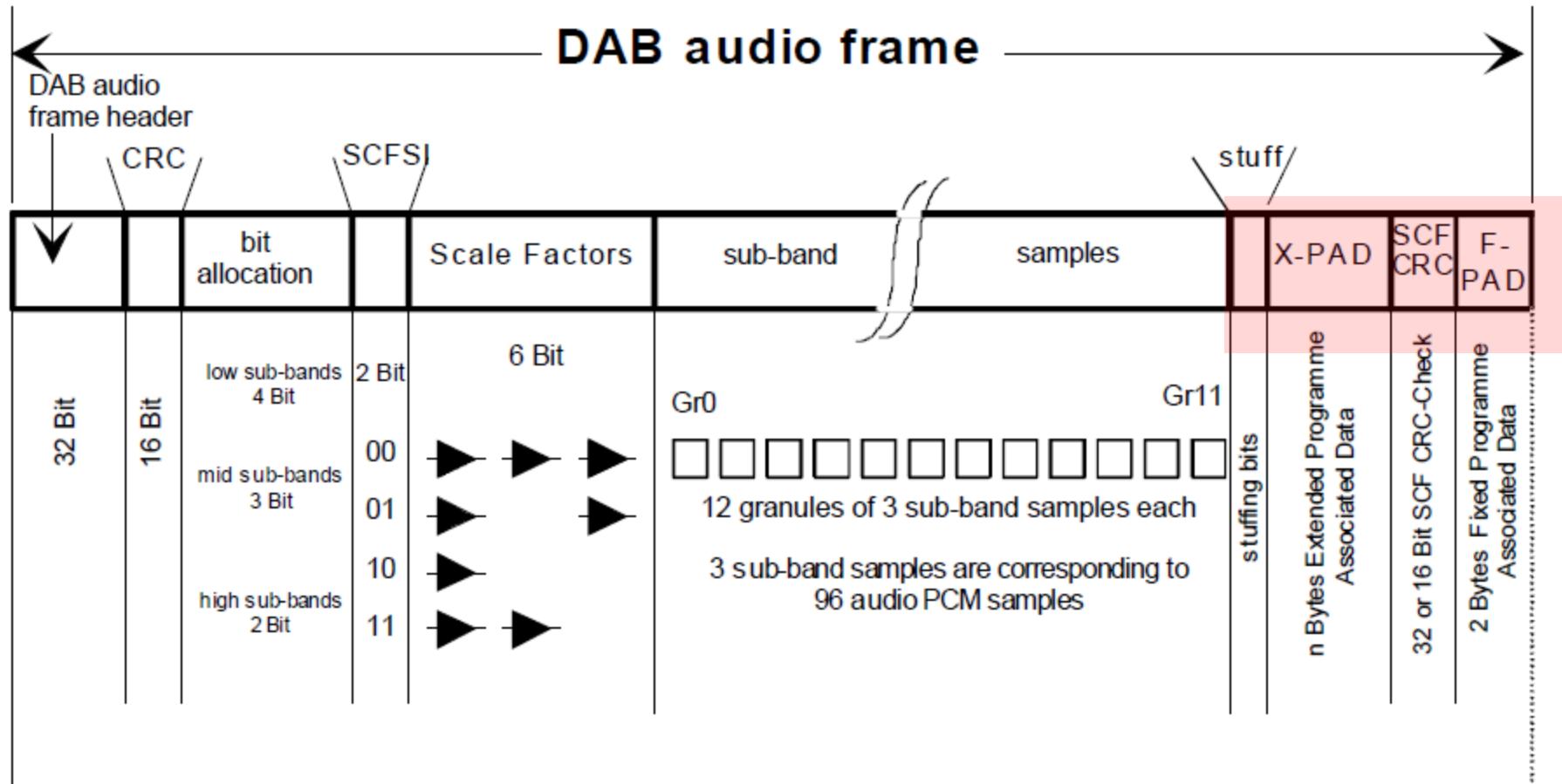
- ◆ 一个DAB Audio Frame与一个MP2帧对应





# DAB audio frame 结构

F-PAD: Fixed Program Associated Data  
X-PAD: Extended PAD





# 小结：DAB与MP3的区别？

## ◆ 编码方式

- DAB: MPEG-1 Audio Layer II , MP2
- MP3: MPEG-1 or MPEG-2 Audio Layer III

## ◆ 音频以外的附加信息

- MP3采用Tag形式，较强的扩展
- DAB局部修改了MP2的帧(PAD)

## ◆ 应用方式

- MP3面向存储 ← 音频存储
- DAB面向无线通信系统 ← 音频流媒体

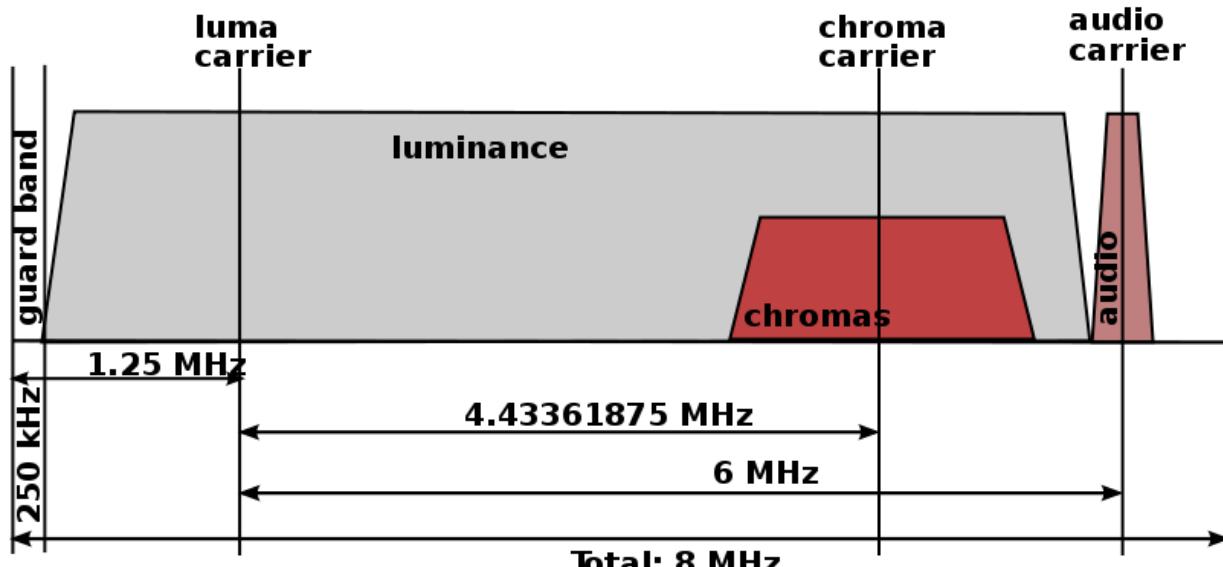
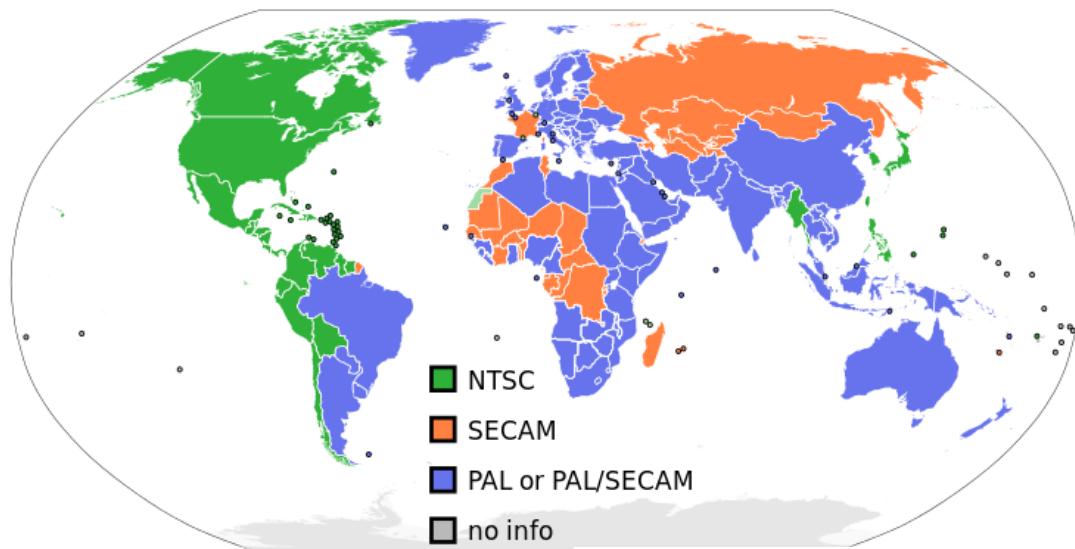


# 裸数据的组织

- ◆ 现有网络中数据组织的方式
- ◆ 主流音视频流的数据组织方式
  - 数字音乐 CD-DA
  - 音乐文件 MP3
  - 数字音频广播 DAB
  - 数字电视 MPEG2
    - 数字电视现状
    - HDTV数据流格式
  - MPEG-4 AVC / H.264



# 模拟电视



中国电视是PALD/K制式，图像信号带宽 $7.25\text{MHz}$ ，伴音载频比图像载频高 $6.5\text{MHz}$ ，带宽± $0.25\text{MHz}$ ，所以每个频道带宽为 $8\text{MHz}$ ，从图像载频 $-1.25\text{MHz}$ 至图像载频频率 $+6.5\text{MHz}$ 。



# 数字电视标准之争

- ◆ 由于高清晰度电视的标准肯定是全数字式，所以高清晰度电视的标准的竞争又扩大到数字电视系统的竞争。
- ◆ 目前竞争的两大标准是欧洲的数字电视广播联盟（**DVB**，Digital Video Broadcasting）和美国的先进电视系统联盟（**ATSC**，Advanced Television System Committee）。
- ◆ 继欧、美之后，日本在欧洲DVB技术的基础上，研制出世界上第三个拥有独立知识产权的数字电视地面广播标准（**ISDB**，Integrated Services Digital Broadcasting）。



# DVB系列标准

## ◆ 卫星广播 DVB-S

□ (Digital Video Broadcasting-Satellite)

## ◆ 有线电视 DVB-C

□ (Digital Video Broadcasting-Cable)

## ◆ 地面广播 DVB-T

□ (Digital Video Broadcasting-Terrestrial )

## ◆ 手持设备 DVB-H

□ (Digital Video Broadcasting Handheld)



# 国内地面广播标准之争

- ◆ 从2000年开始，上海交通大学、清华大学、西安电子科技大学、广播总局科学研究院、成都电子科技大学共5家教育科研单位开始了数字电视地面传输标准的研发。2003年，上海交通大学的ADTB-T标准、清华大学的DMB-T标准、广播总局科学研究院的CDTB-T标准入围。
  - DMB-T (Terrestrial Digital Multimedia/Television Broadcasting) 基于 TDS-OFDM (Time Domain Synchronous -Orthogonal Frequency Division Multiplexing) 调制技术
  - 上海交大的ADTB-T (Advanced Digital Television Broadcasting System - Terrestrial) 标准是单载波调制技术：4/16/64QAM
- ◆ 2006年，国内标准确定
  - DTMB (Digital Terrestrial Multimedia Broadcast)



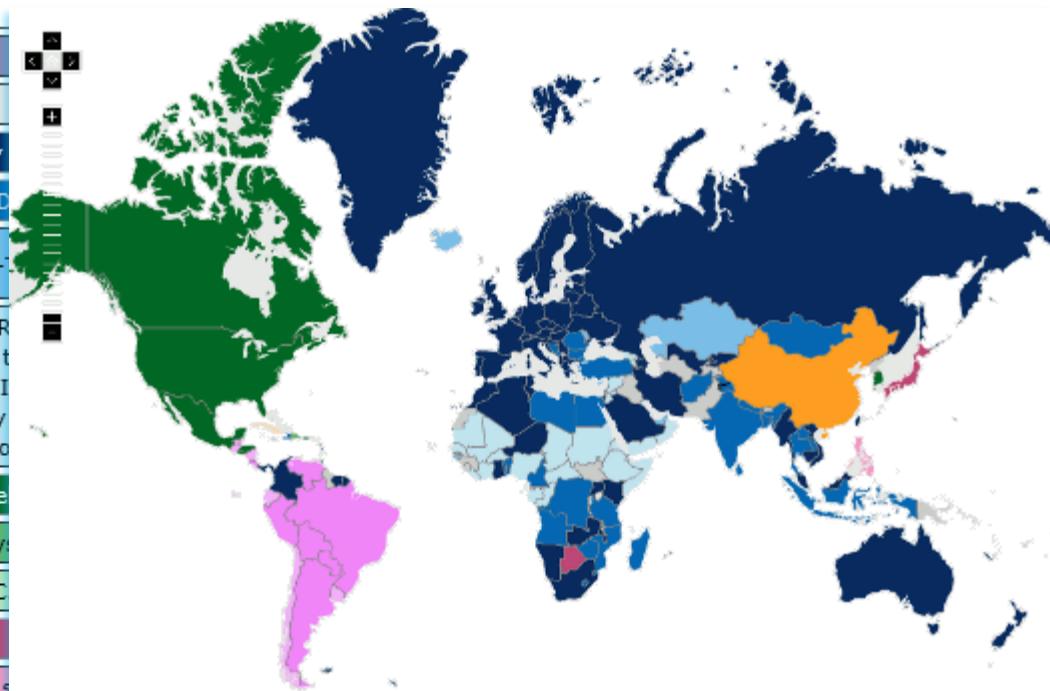
# 地面广播标准比较

	美ATSC	欧DVB-T	日ISDB	清华DMB-T
视频	<b>MPEG-2—视频部分</b>			
音频	Dolby AC-3	MPEG1 layerII Dolby AC-3	MPEG2- AAC	MPEG1 LayerI/II Dolby AC-3
外码	RS,参数和交织方式不同			
内码	2/3格形码	卷积码		卷积码,网格码 Turbo码
调制	8,16VSB	COFDM	BST-OFDM	TD6-OFDM
均匀星座图	8,16VSB	QPSK/16QAM/ 64QAM	QPSK/16QAM/ 64QAM	QPSK/16QAM/6 4QAM
分级调制		多分别率星座图(16QAM 和 54QAM)	在13个分段上有3种不同的调制	多分别率星座图 (64QAM和 256QAM)



# 数字电视地面广播标准部署现状

Brief overview	
System	Explanation
DVB-T/DVB-T2	Broadcasting via DVB-T/DVB-T2 is actively in use.
DVB-T/DVB-T2 adopted	Countries which have adopted the DVB-T/DVB-T2 system.
DVB-T/DVB-T2 trial broadcasts	Those countries undertake trials with DVB-T/DVB-T2.
RRC06	The according countries participate in the Radiocommunications Conference 2006 of the International Telecommunication Union. It is expected that all countries taking part will ultimately adopt the DVB-T/DVB-T2 system when they move from RRC06.
ATSC	Broadcasting via the ATSC system is actively in use.
ATSC adopted	Countries which have adopted the ATSC system.
ATSC trial broadcasts	Those countries undertake trials with ATSC.
ISDB-T	Broadcasting via ISDB-T is actively in use.
ISDB-T adopted	Countries which have adopted the ISDB-T system.
ISDB-T trial broadcasts	Those countries undertake trials with ISDB-T.
SBTVD-T	Broadcasting via SBTVD-T is actively in use.
SBTVD-T adopted	Countries which have adopted the SBTVD-T system.
DTMB	Broadcasting via DTMB is actively in use.
DTMB adopted	Countries which have adopted the DTMB system.
DTMB trial broadcasts	Those countries undertake trials with DTMB.
Commercial DVB-T services	No formal adoption of a DTT standard.
	Undecided countries.



**SBTVD**, short for Sistema Brasileiro de Televisão Digital (English: Brazilian Digital Television System) is a technical standard **based on ISDB-T** standard, launched in commercial operation on December 2, 2007.



## 地面数字电视广播覆盖网发展规划

到2011年底，全国拥有电视转播发射台15397座，电视发射机30082部，总发射功率13960千瓦，无线电视综合覆盖率达94.52%，电视机社会拥有量超过5亿台。

在此基础上，分三步继续推动有关工作，计划分为以下三个步骤：

第一步：2013-2015年，争取在全国县级（含）以上城镇以高、标清方式播出地面数字电视，并逐步开始优化省会城市以及地市和县的覆盖网络。

从2015年开始，当全国地级（含）以上城市的中央电视台第一套、第七套和本省第一套、本地第一套地面数字电视同播节目的人口覆盖率达到模拟电视的水平且地面数字电视接收机基本普及后，逐步停播地面数字电视服务区内的模拟电视发射（本地还没有同播的模拟电视可直接转换成数字电视）。

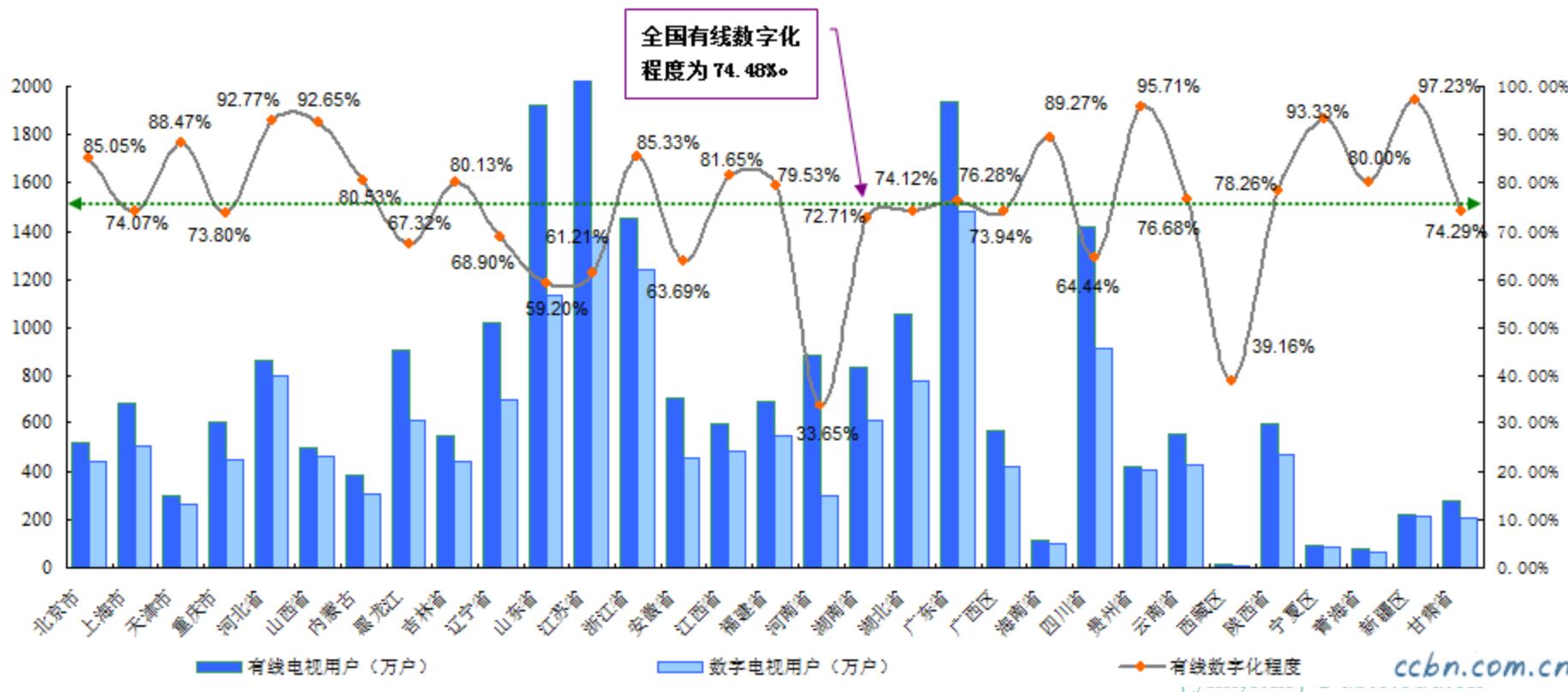
第二步：到2018年底，全国地级（含）以上城市地面电视完成向数字化过渡。全国县级（含）以上城镇的中央电视台第一套、第七套和本省第一套、本地第一套地面数字电视同播节目人口覆盖率达到模拟电视的水平且地面数字电视接收机基本普及后，开始逐步停播模拟电视。

第三步：到2020年底，全面完善地面数字电视广播覆盖网，当全国中央电视台第一套、第七套和本省第一套、本地第一套地面数字电视同播节目人口覆盖率达到模拟电视的水平且地面数字电视接收机基本普及后，全面关闭地面模拟电视信号，完成地面电视向数字化过渡。



# 中国有线电视数字化程度进展

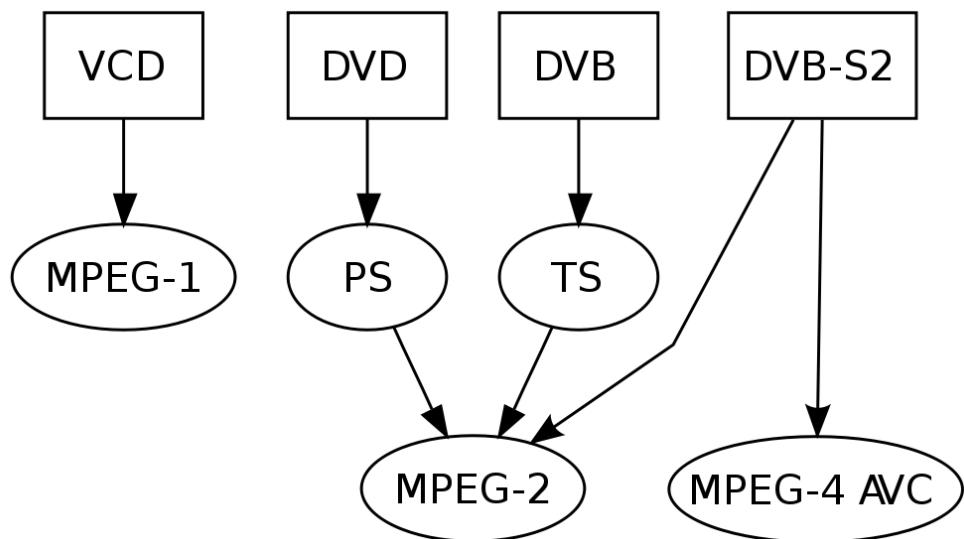
截至2014年4月底，我国有线数字电视用户达到16682.8万户，有线数字化程度约为74.48%（有线电视用户基数为2.24亿户，数据来源于国家广电总局），我国有线数字化整体转换已步入中后期。





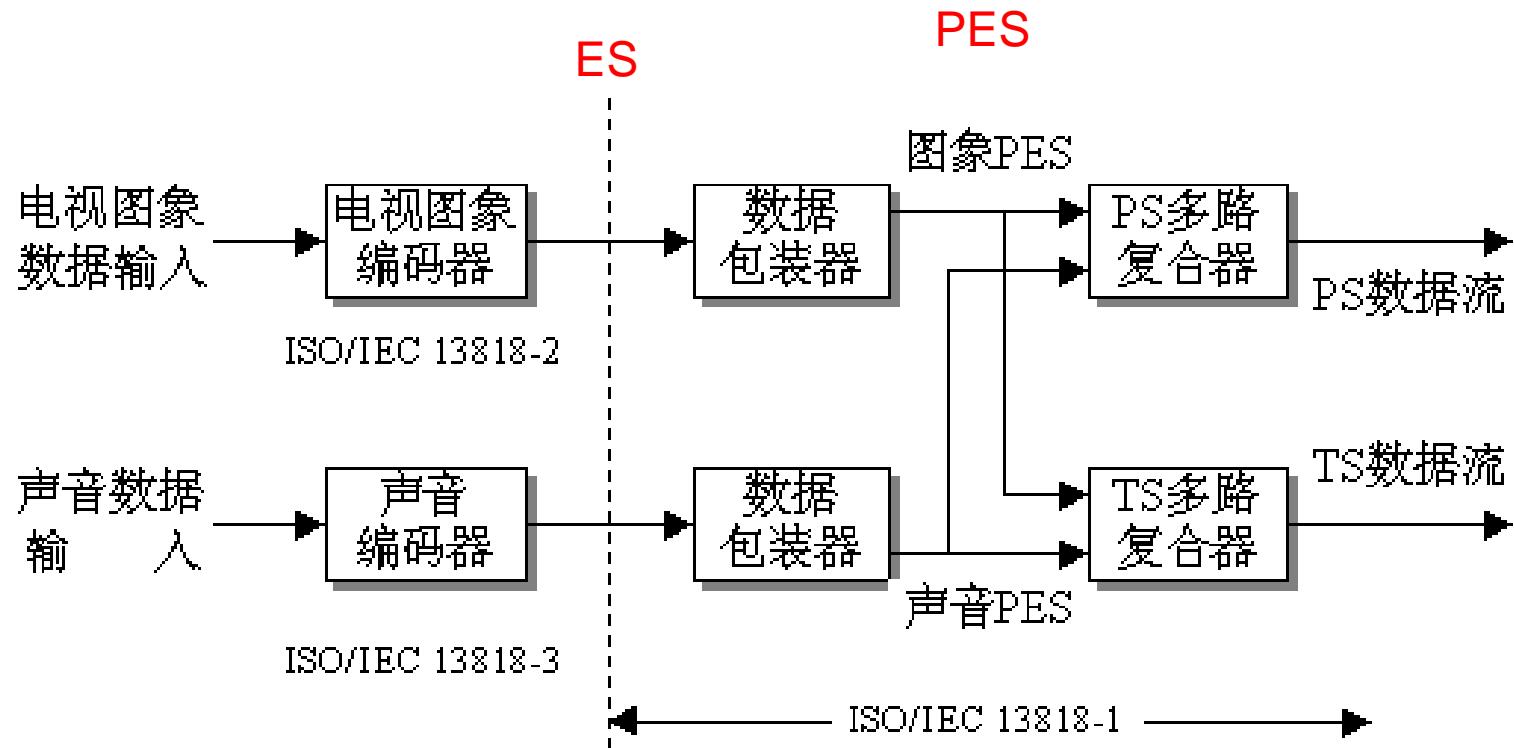
# 数据电视的音视频编码

◆ MPEG-2 is widely used as the format of **digital television signals** that are broadcast by terrestrial (over-the-air), cable, and direct broadcast satellite TV systems. It also specifies the format of movies and other programs that are distributed on DVD and similar discs. As such, TV stations, TV receivers, DVD players.





# MPEG-2 系统传输层结构





# MPEG数据流：标准

## ◆ MPEG-2标准

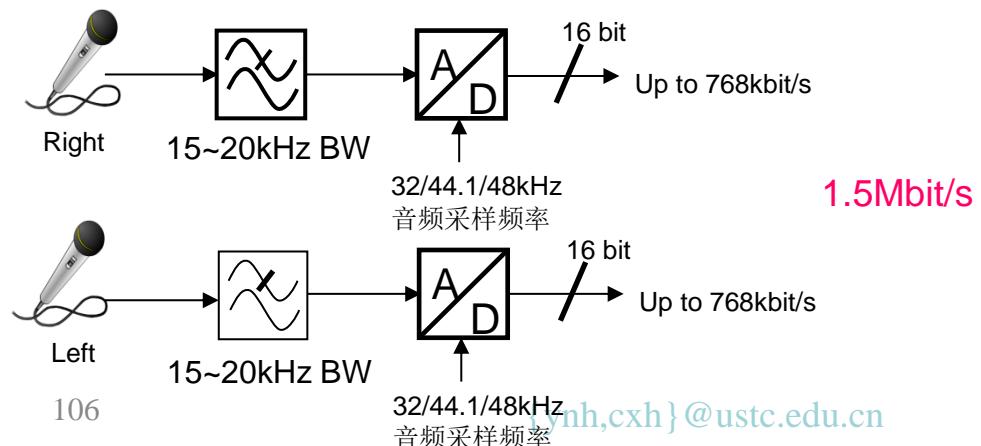
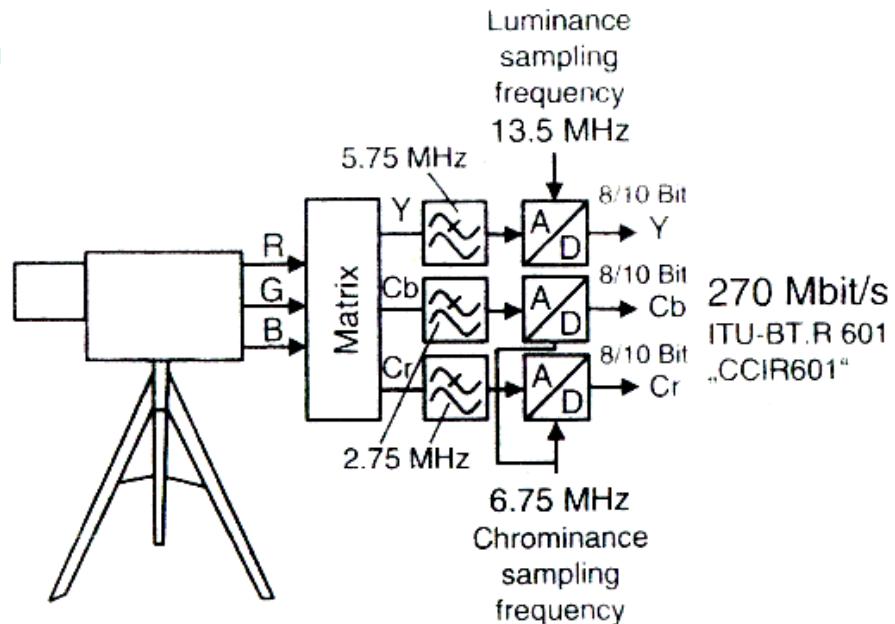
- ISO/IEC 13818-1 系统层
- ISO/IEC 13818-2 视频编码层
- ISO/IEC 13818-3 音频编码层

◆ 系统层也可用于传送音视频以外的数据，如Internet数据。

◆ 系统层描述MPEG数据流的整体结构，实际中具有重要意义。

◆ 原始SDTV信号（ITU601）码率**270Mbit/s**；

◆ CD质量的原始数字立体声音频信号码率为**1.5Mbit/s**。



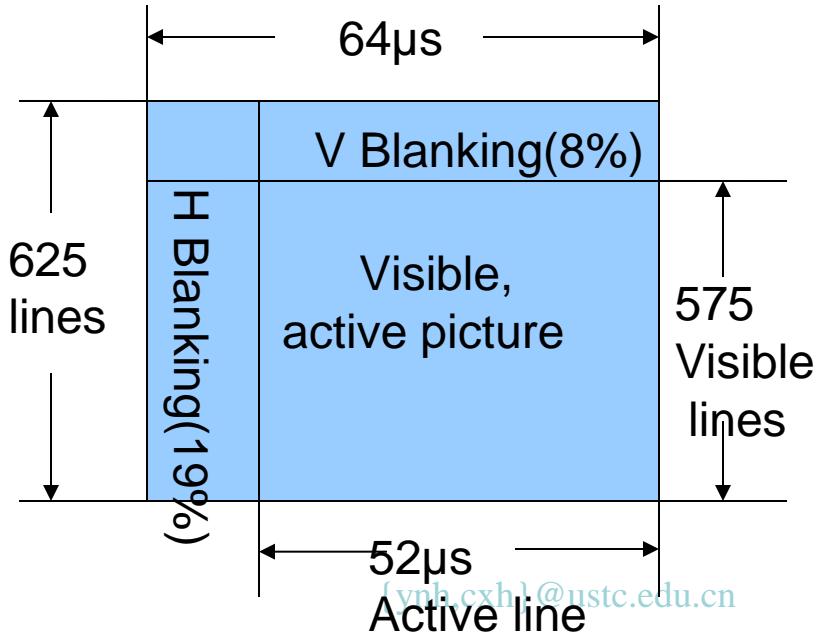
# 270Mbps → 2-6Mbps ?

## ◆ CCIR 601: 270Mbps

- Y: 13.5 MHz Cr, Cb: 6.75 MHz / 10bits

## ◆ MPEG-2: 2-6Mbps

- YUV 4:4:4 → YUV 4:2:2 20%
- YUV 4:2:2 → YUV 4:2:0 20%
- 去除空白 25%
- DPCM/DCT/Huffman
- 去帧间冗余





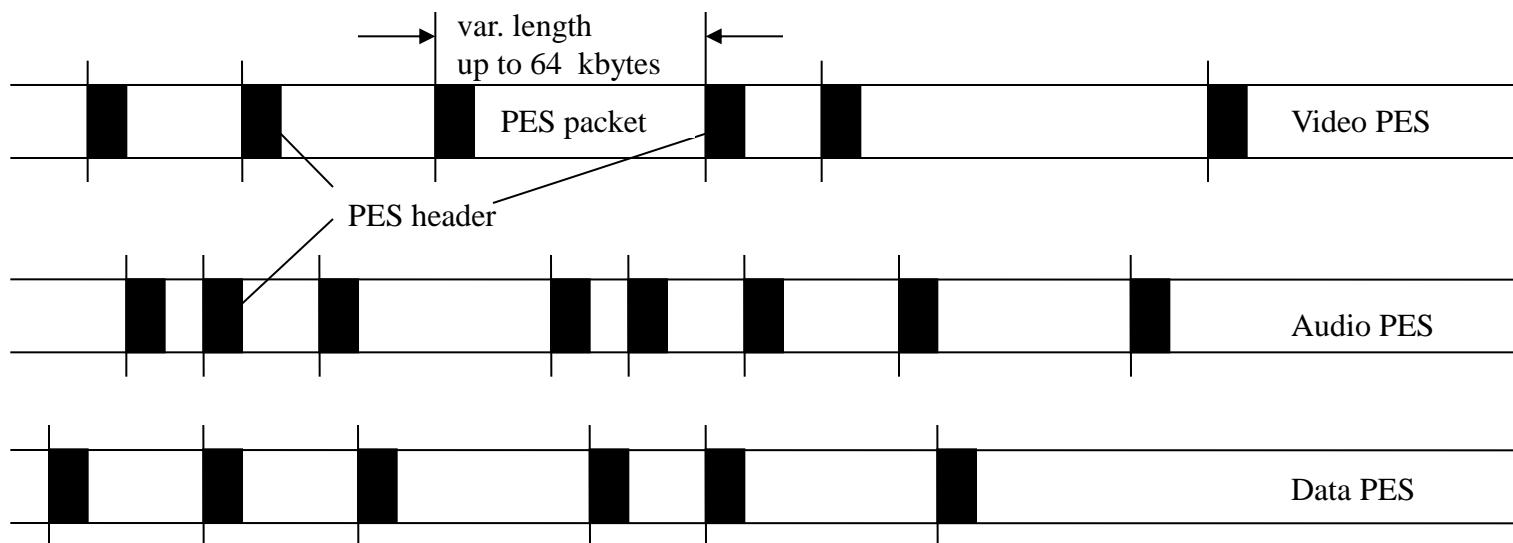
# 视频ES流静荷内容

- ◆ 块：视频流的最小单元是8x8像素块，是DCT的单位。
- ◆ 宏块：是量化控制的单位，不同宏块可采用不同量化因子。
  - 量化因子是实际视频PES流控制码率的调节器。
  - 量化表不能在宏块层改变。
  - 宏块可以是帧编码或场编码。
- ◆ Slice：一行中若干宏块构成一个slice
  - 每个slice头在误码时用于重同步
  - 误码掩盖主要在slice层，发生误码时，MPEG解码器将前一帧的slice复制到当前帧，解码器可以在一个新slice开头重新同步，slice越短，误码造成的干扰越小。
- ◆ 帧：若干slice构成一帧，有I/P/B帧
  - 每一帧都有帧头
  - 由于B帧，编码顺序和显示顺序不同，帧头和PES头带有时间戳，可以恢复原始帧顺序。
- ◆ GOP：若干I/P/B帧构成一个GOP
  - 每一组GOP都有GOP头
  - 广播采用短GOP，半秒12帧
  - MPEG解码器只能在GOP的第一个I帧重同步。
  - 大容量设备如DVD可以采用长GOP。
- ◆ 序列：一个GOP或多个GOP构成一个序列
  - 序列有序列头，包括重要视频参数如量化表。



# MPEG数据流：ES

- ◆ 视频信号压缩到1Mbit/s(MPEG-1)和2~6Mbit/s(MPEG-2)
- ◆ 音频信号压缩到100~400kbit/s
- ◆ 压缩后的视音频信号称作ES(elementary stream)流，包括：
  - 视频流
  - 音频流
  - 数据流——任何类型的压缩或未压缩数据





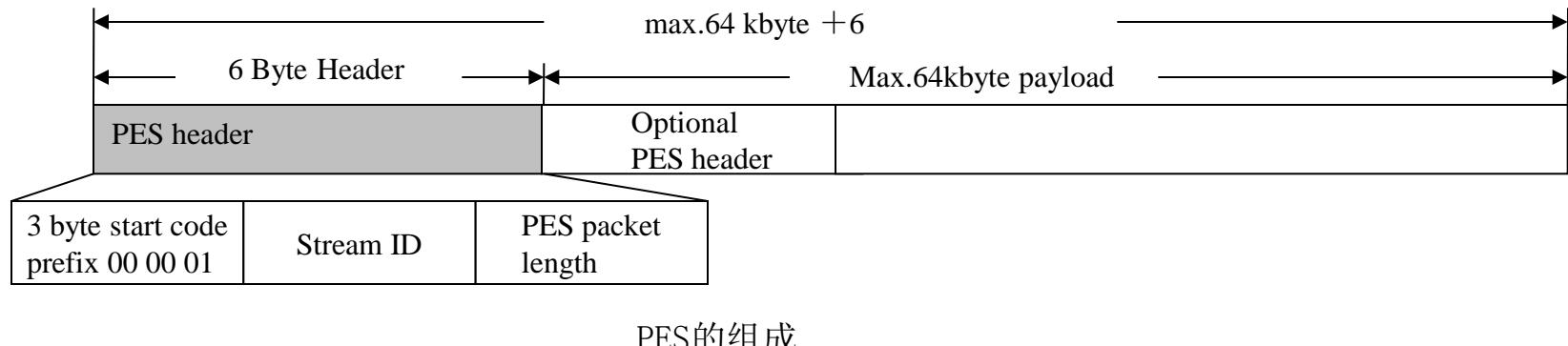
# MPEG数据流：PES

- ◆ 压缩后所有ES流被打成不同长度的包，叫做 **PES(packetized elementary streams)**。
- ◆ 根据不同时刻视音频内容的不同，压缩比也时刻变化，就需要有不同长度的数据包。
  - 每个视频包有一个或几个压缩视频帧
  - 每个音频包有一个或多个压缩音频信号段
- ◆ 每个PES包最大64kbytes，由头Header和净负荷数据 payload组成：
  - header包含了16位bit指示包的长度
  - payload包含压缩视频/音频流或纯数据流
  - 视频包的长度某些情况下可大于64kbytes，此时包头(header)中包长度指示为0，解码器要利用其他机制找出包的终止处。



# PES头标

- ◆ 所有ES流首先被打包成不同长度的PES包，通常为64kbytes。
- ◆ 开头为6byte的PES头：
  - 前3个byte是起始码前缀00 00 01，用于表明一个PES包的开始。
  - 第4个byte是起始码标志，说明起始码种类，表明payload中是视频、音频还是数据流。
  - 后两个byte是包长度，说明后面还有多少字节。如果长度为0，表示PES包大于64kbytes。



PES的组成

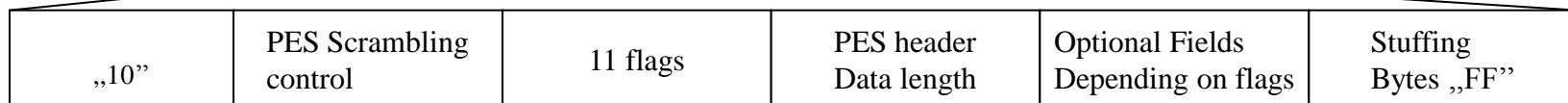
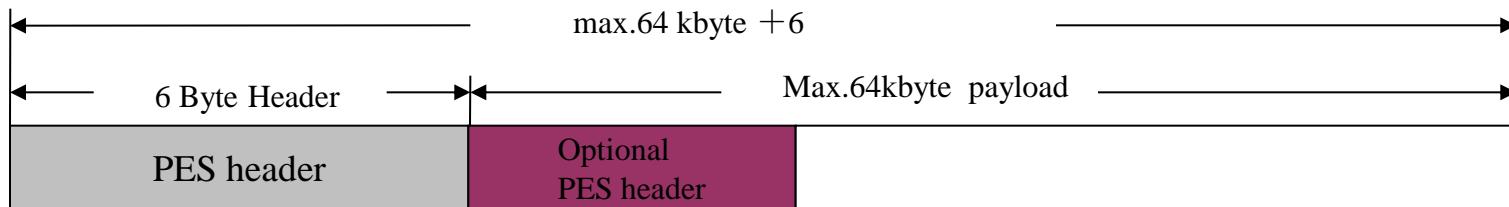
- ◆ 然后是可选PES头
- ◆ 最后是实际传送ES流的净负荷数据(payload)



# PES扩展头标

## ◆ 可选PES头：

- PES头的可选扩展，根据当前传送ES流的要求设置。
- 由12个比特的11个标志来控制可选PES头中包含哪些字段，其中有PTS(presentation time stamps)和DTS(decoding time stamps)，这对视/音频同步非常重要。
- 最后可能有填充字节。

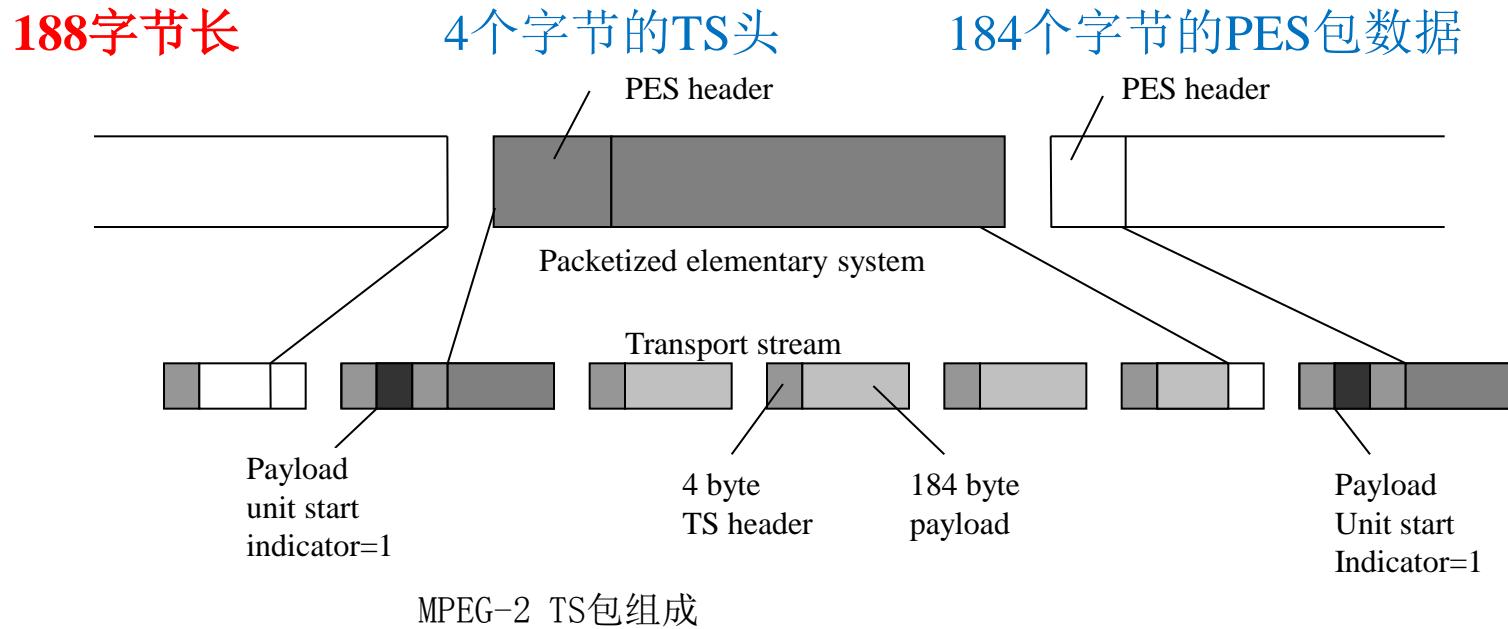


PTS	DTS	ESCR	ES rate	DSM Trick mode	Additional Copy Info	Previous PES CRC	PES extension
33	33	42	22	112	8	8	16

PES的组成

# PES → TS

- ◆ PES结构不适合传输，尤其不适合一个流中有多个节目的广播应用。
- ◆ 而MPEG-2目标是将6,10甚至20个电视节目或广播节目形成一个复用MPEG-2数据流，然后通过卫星、电缆或地面传输。
- ◆ 因此，PES包再分成固定长度的更小的包，即TS包(transport stream packets)：





# 包的复用

## ◆ 对TS包再进行复用：

- 先复用同一个节目的TS包，一个节目可包含一个或多个视频和音频信号（如不同角度摄像机、不同语言等）。
- 所有节目的所有复用数据流再进行复用形成最终的TS流。



Multiplexed video and audio PES packets

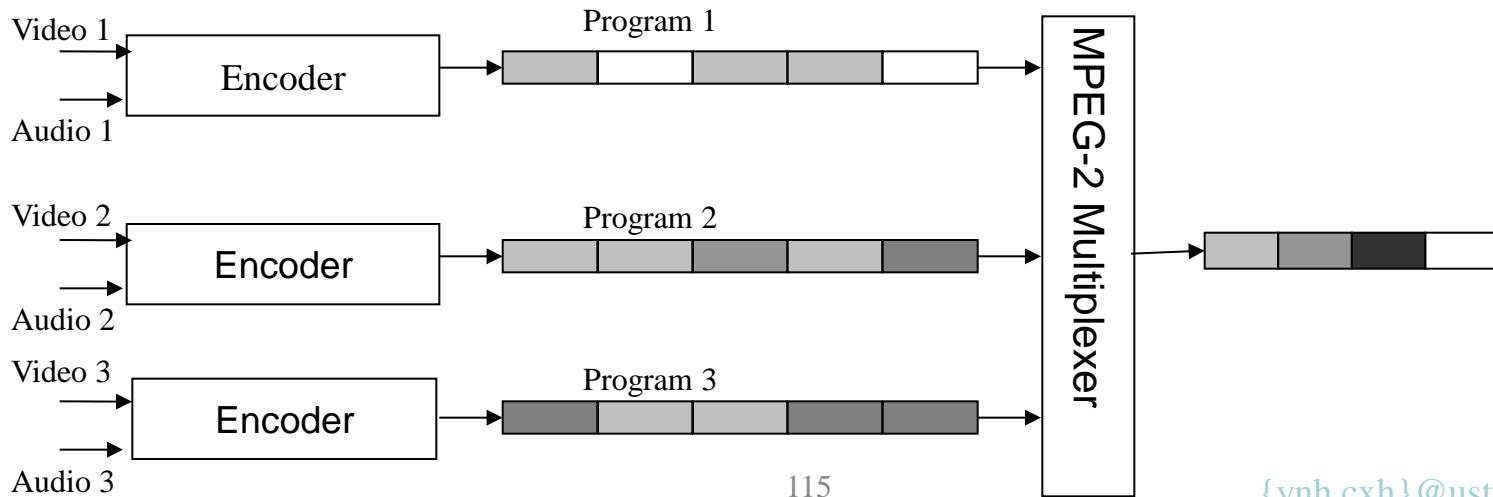
Application:  
MPEG-1 Video CD  
MPEG-2 SVCD  
MPEG-2 Video DVD

- MPEG-1，视频PES包与音频PES包复用，最大码率为1.5Mbit/s，用于VCD。



# MPEG-2 TS流的复用

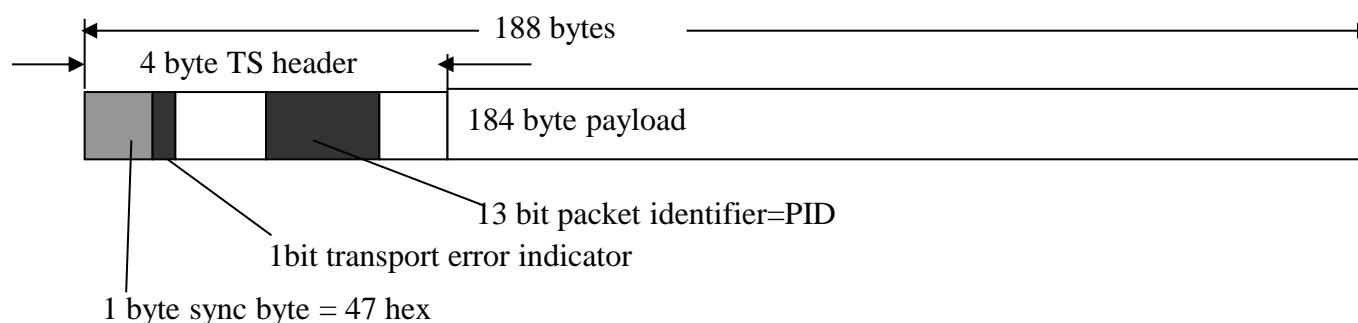
- ◆ MPEG-2的TS包长188字节，包含所有节目的所有数据。
- ◆ 由于码率不同，MPEG-2 TS流中不同ES流的包出现频率不一样。
- ◆ 每个节目有一个编码器对所有ES流编码，产生PES，并将PES包打包成TS包。
- ◆ 每个节目的码率通常约2~8Mbit/s，但由于节目内容随时间变化，视/音频和数据总码率可以是固定或变化的，称为统计复用。
- ◆ 所有节目的TS流再复用成一个总的TS流，最大约40Mbit/s。





# MPEG-2 TS包:头标

- ◆ 固定长度188字节，4字节header和184字节payload



- ◆ header包含了对包传输过程非常重要的信息：

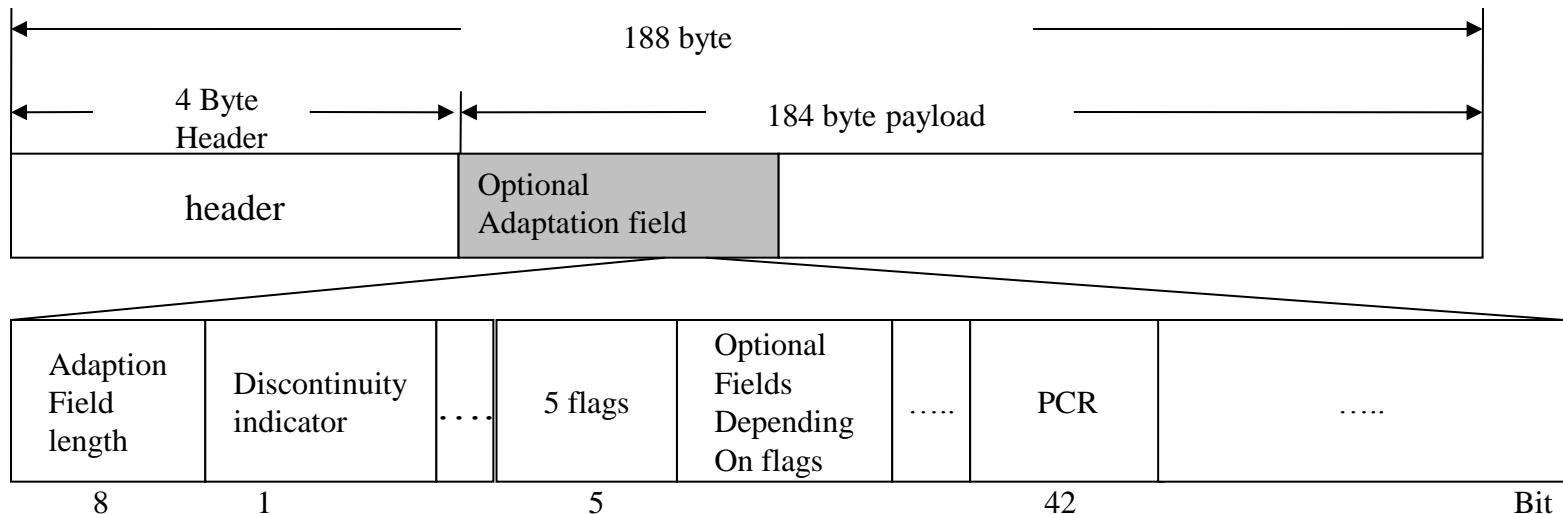
- 第一个字节是**同步**字节
  - 固定值47hex，在TS流中的间隔也固定。
  - 码流中其他位置也可能出现47hex，因此同步字节利用固定数值和固定间隔两方面联合实现同步。
  - 解码器在接收到5个TS包后开始同步。
- 同步字节后的一个比特是传输**差错标志**
  - 由解调器在传输信道末端设置
  - 例如错误太多无法利用误码纠正机制进行恢复的情况。
- 13bit的**PID(packet identifier)**
  - 描述该包中payload的内容以及该包属于哪个ES流。



# MPEG-2 TS包: 扩展头标

◆ 有时TS包的头长度必须大于4个byte:

- 包头扩展到payload中， payload长度相应缩短，但总包长度仍是188bytes。
- 扩展的包头称作“adaptation field”。
- 在4bytes的包头中由Adaptation control bits标志是否有adaptation field。



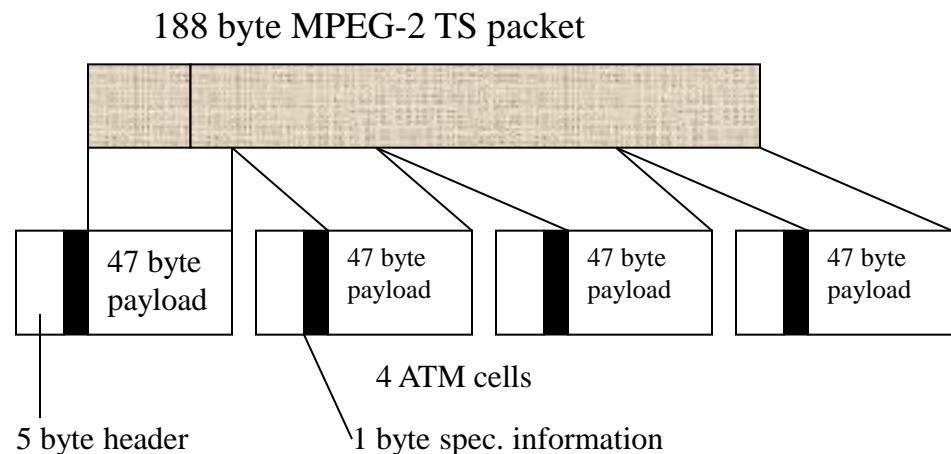
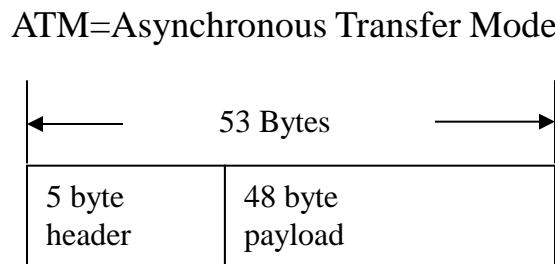
带adaptation field的TS包



# MPEG-2 TS包: ATM

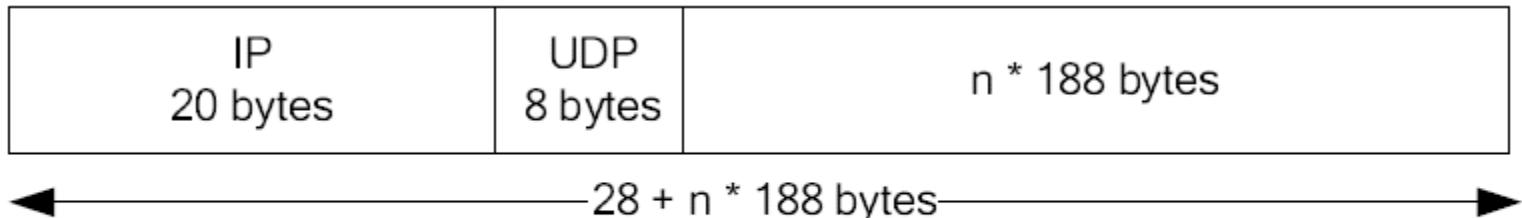
◆ TS包的结构和长度与电话和LAN技术采用的异步转移模式ATM类似：

- ATM用于电话的远程网络和局域网的计算机网络中。
- ATM也采用包结构，每个包53bytes，由5bytes的头和48bytes的payload组成。
- MPEG-2初期考虑利用ATM传输，ATM包的payload中有一个特殊byte，实际payload只有47bytes，因此TS包的188bytes正好可以由4个ATM包传送。
- 实际也存在MPEG-2通过ATM传输。

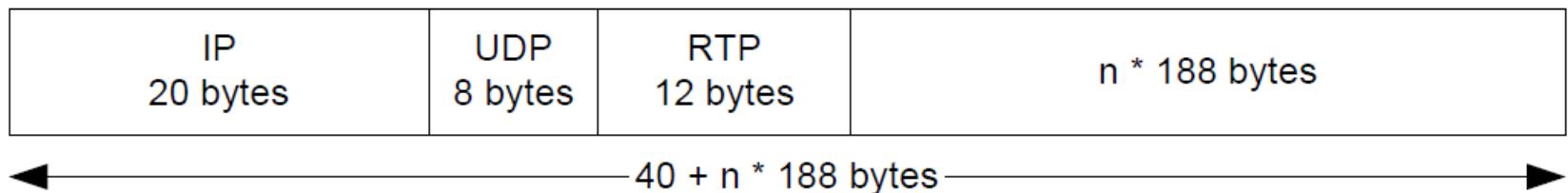




# DVB中TS流的IP承载



**Minimal packet format (IPv4) for UDP encapsulation**



**Minimal packet format (IPv4) for RTP encapsulation**



# 下一代的数字电视标准? DVB系列的演进

- ◆ Satellite: DVB-S → DVB-S2
- ◆ Cable: DVB-C → DVB-C2
- ◆ Terrestrial television: DVB-T → DVB-T2
- ◆ Microwave: using DTT (DVB-MT), the MMDS (DVB-MC), and/or MVDS standards (DVB-MS)

	DVB-S2	DVB-T2	DVB-C2
Input Interface	Multiple <a href="#">Transport Stream</a> and <a href="#">Generic Stream Encapsulation</a> (GSE)	Multiple Transport Stream and Generic Stream Encapsulation (GSE)	Multiple Transport Stream and Generic Stream Encapsulation (GSE)
Modes	<a href="#">Variable Coding &amp; Modulation</a> and <a href="#">Adaptive Coding &amp; Modulation</a>	<a href="#">Variable Coding &amp; Modulation</a> <sup>[4]</sup>	<a href="#">Variable Coding &amp; Modulation</a> and <a href="#">Adaptive Coding &amp; Modulation</a>
FEC	LDPC + BCH 1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 9/10	LDPC + BCH 1/2, 3/5, 2/3, 3/4, 4/5, 5/6	LDPC + BCH 1/2, 2/3, 3/4, 4/5, 5/6, 8/9, 9/10 <sup>[5]</sup>
Modulation	Single Carrier <a href="#">QPSK</a> with Multiple Streams	<a href="#">OFDM</a>	absolute OFDM <sup>[6]</sup>
Modulation Schemes	QPSK, 8PSK, 16APSK, 32APSK	QPSK, <a href="#">16QAM</a> , 64QAM, 256QAM	16- to 4096-QAM
Guard Interval	Not Applicable	1/4, 19/256, 1/8, 19/128, 1/16, 1/32, 1/128	1/64 or 1/128
Fourier transform size	Not Applicable	1k, 2k, 4k, 8k, 16k, 32k <a href="#">DFT</a>	4k <a href="#">Inverse FFT</a> <sup>[7]</sup>
Interleaving	Bit-Interleaving	Bit- Time- and Frequency-Interleaving	Bit- Time- and Frequency-Interleaving
Pilots	Pilot symbols	Scattered and Continual Pilots	Scattered and Continual Pilots



# Generic Stream Encapsulation (GSE) Protocol

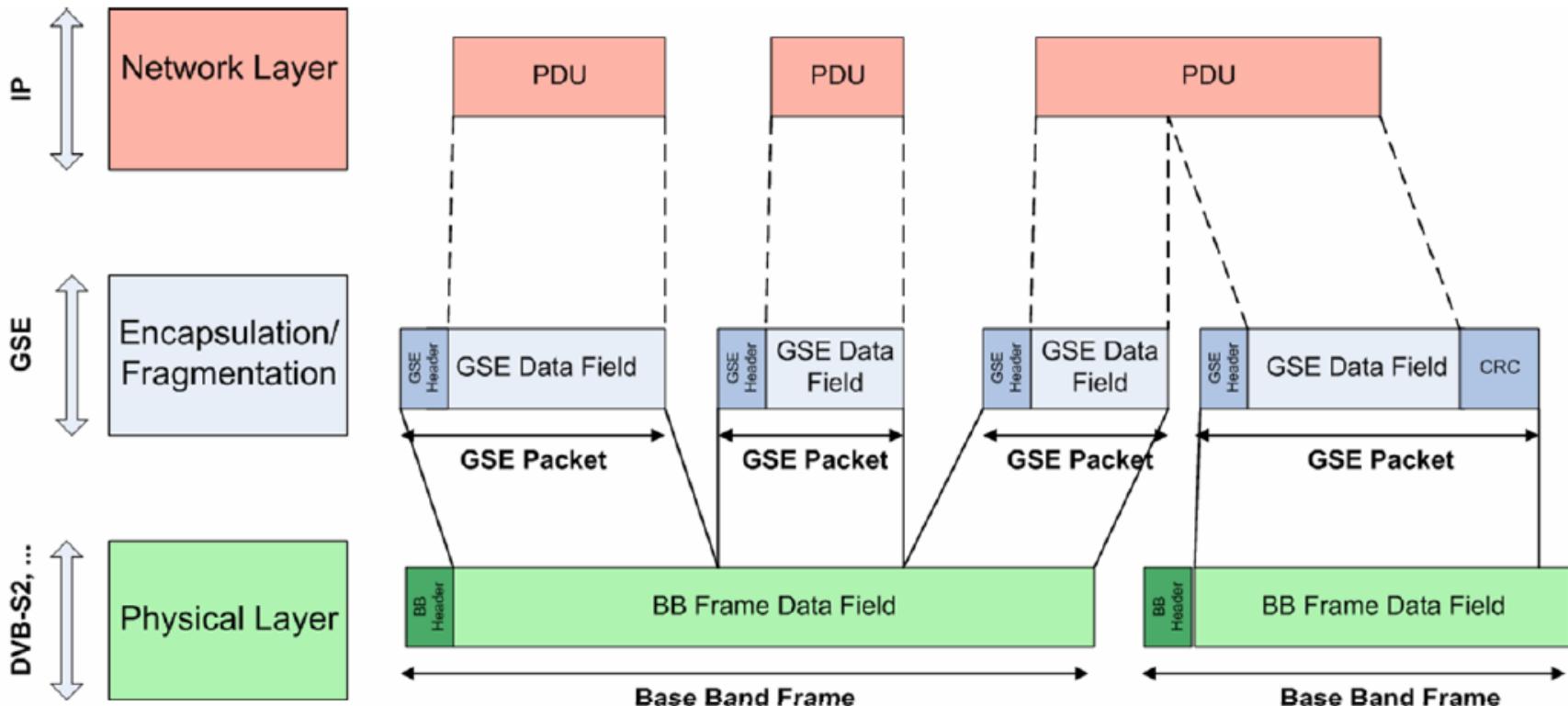
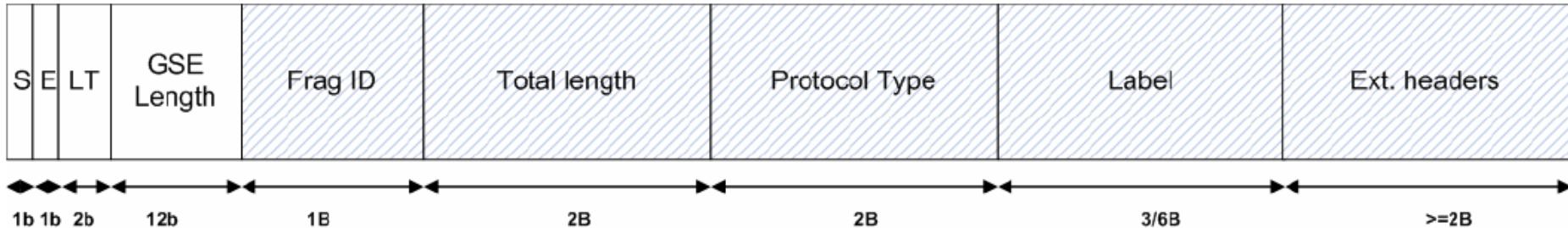


Figure 1: GSE encapsulation within DVB protocol stack



# GSE Header Format

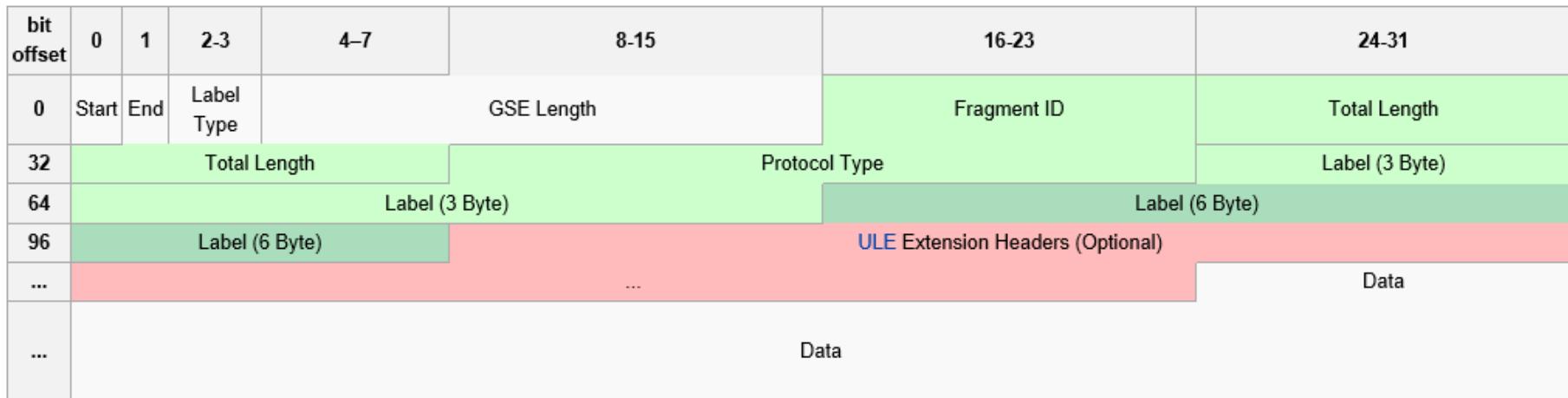


- **Start\_Indicator**: A value of "1" indicates that this GSE Packet contains the start of the encapsulated PDU.
- **End\_Indicator**: A value of "1" indicates that this GSE Packet contains the end of the encapsulated PDU.
- **Frag\_ID**: This is present when a PDU fragment is included in the GSE Packet
- **Protocol\_Type**: This 16-bit field indicates the type of payload carried in the PDU
- The **GSE Label Field** is optional. This field shall be carried for IP unicast packets destined to routers using shared links (i.e., where the same link connects multiple Receivers). A sender may omit this field for an IP unicast packet and/or multicast packets delivered to Receivers that are able to utilize a discriminator field (e.g. the IPv4/IPv6 destination address, or a bridged MAC destination address) of the encapsulated protocol, which could be interpreted as a Layer 2 address.



# GSE Header Format Details

- ◆ The GSE Packet header is highly dynamic and provides for many options. The minimum header is two bytes, comprising three flags fields, and a 12-bit payload length field. The diagram below shows all possible fields.





# 小结2-1：HDTV标准和流格式

## ◆ 数字电视标准

- DVB/ATSC/ISDB

## ◆ 标准系列

- DVB-S/DVB-C/DVB-T

## ◆ 移动电视

- DVB-T/DMB-T/ADTB-T

## ◆ 视频序列 → GOP → 帧 → Slice → 宏块 → 块

## ◆ ES → PES Packet → TS Packet



## 小结2-2：数字电视系统中的数据封装

### ◆ DVB

- ES→PES→TS
- IP→TS

### ◆ DVB-C2

- Every thing →GSE
- GSE Label Field 可指示MAC地址或IP地址

### ◆ 电视还是网络？



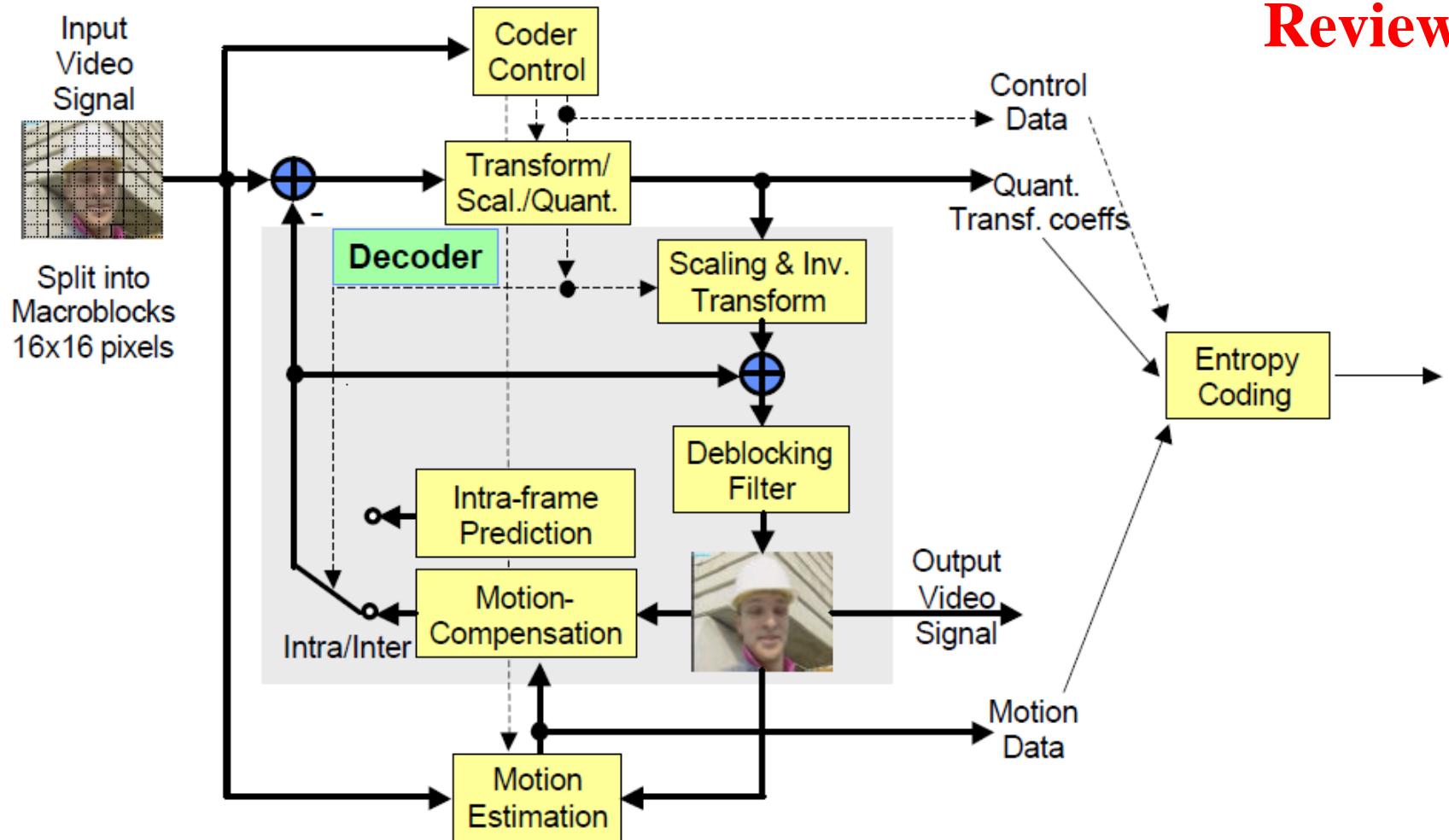
# 裸数据的组织

- ◆ 现有网络中数据组织的方式
- ◆ 主流音视频流的数据组织方式
  - 数字音乐 CD-DA
  - 音乐文件 MP3
  - 数字音频广播 DAB
  - 数字电视 MPEG2
  - **MPEG-4 AVC / H.264**
    - H.264的“Network Friendly”特性
    - H.264码流结构



# H.264性能是如何提高的？

Review





# H.264的主要创新点

Review

## ◆ Motion Compensation

- 1/4象素精度的运动估计、7种大小不同的块进行匹配、前向与后向多参考帧

## ◆ Intra-frame Prediction

- 包括9种 $4 \times 4$ 亮度块的预测、4种 $16 \times 16$ 亮度块的预测和4种色度块的预测

## ◆ De-blocking Filter

## ◆ Transform/Scal./Quant.

- DCT-> $4 \times 4$ 整数变换
- 52级步长的量化器，量化步长以12.5%递增

## ◆ Entropy Coding

- CA VLC:Context-Adaptive Variable Length Coding
- CABAC:Context-Adaptive Binary Arithmetic Coding



# H.264的概念层: VCL + NAL

## ◆ 视频编码层(VCL)和网络抽象层(NAL)

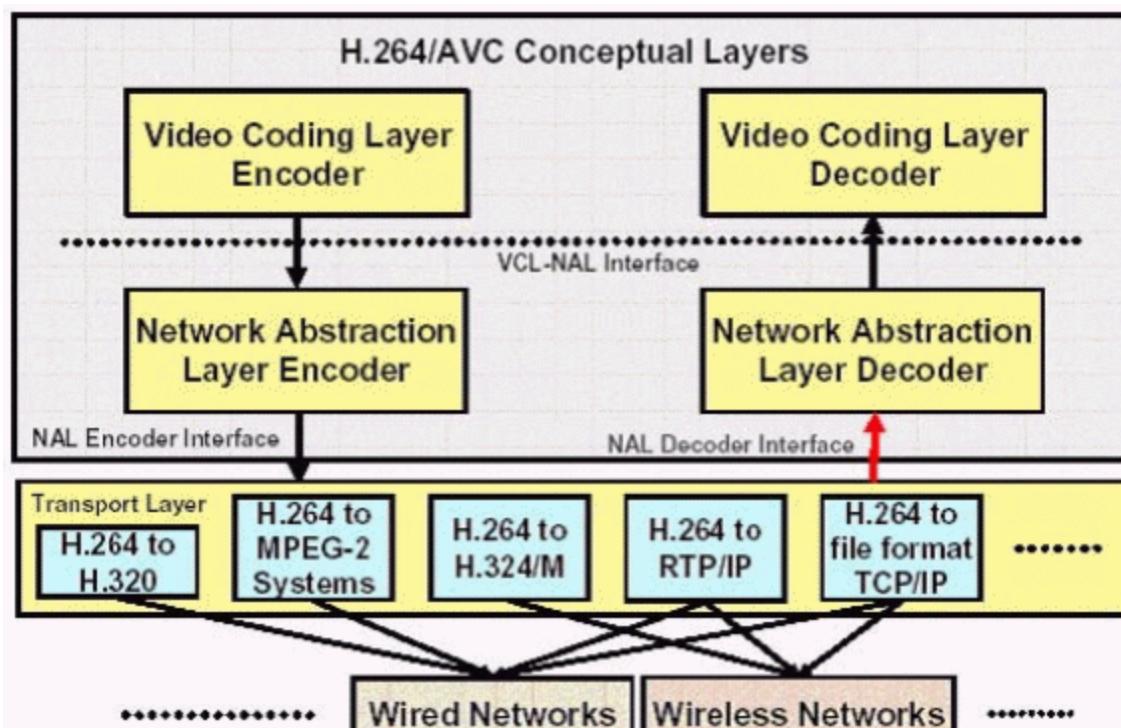
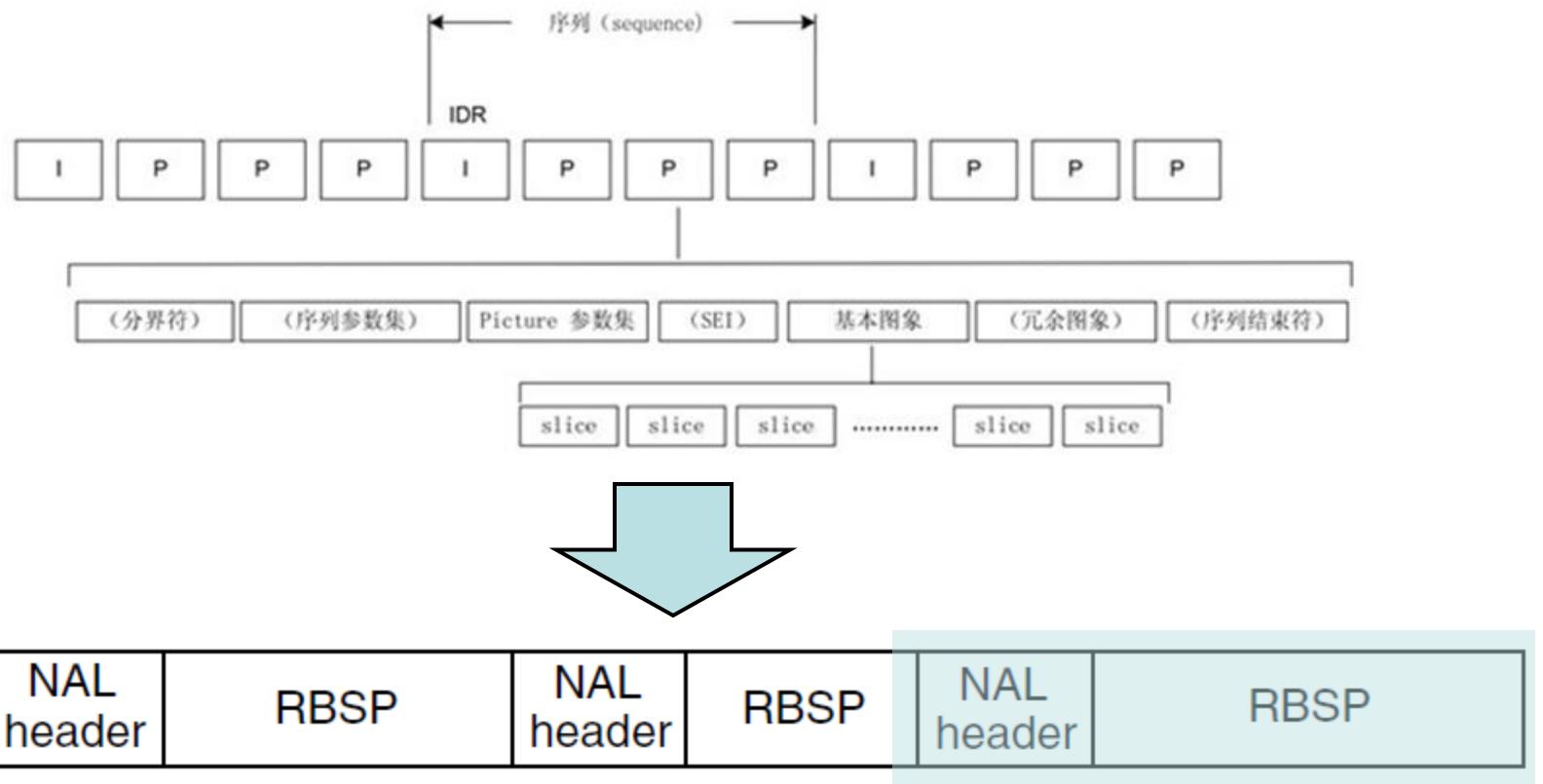


Figure 2 : The H.264/AVC Standard in Transport Environment



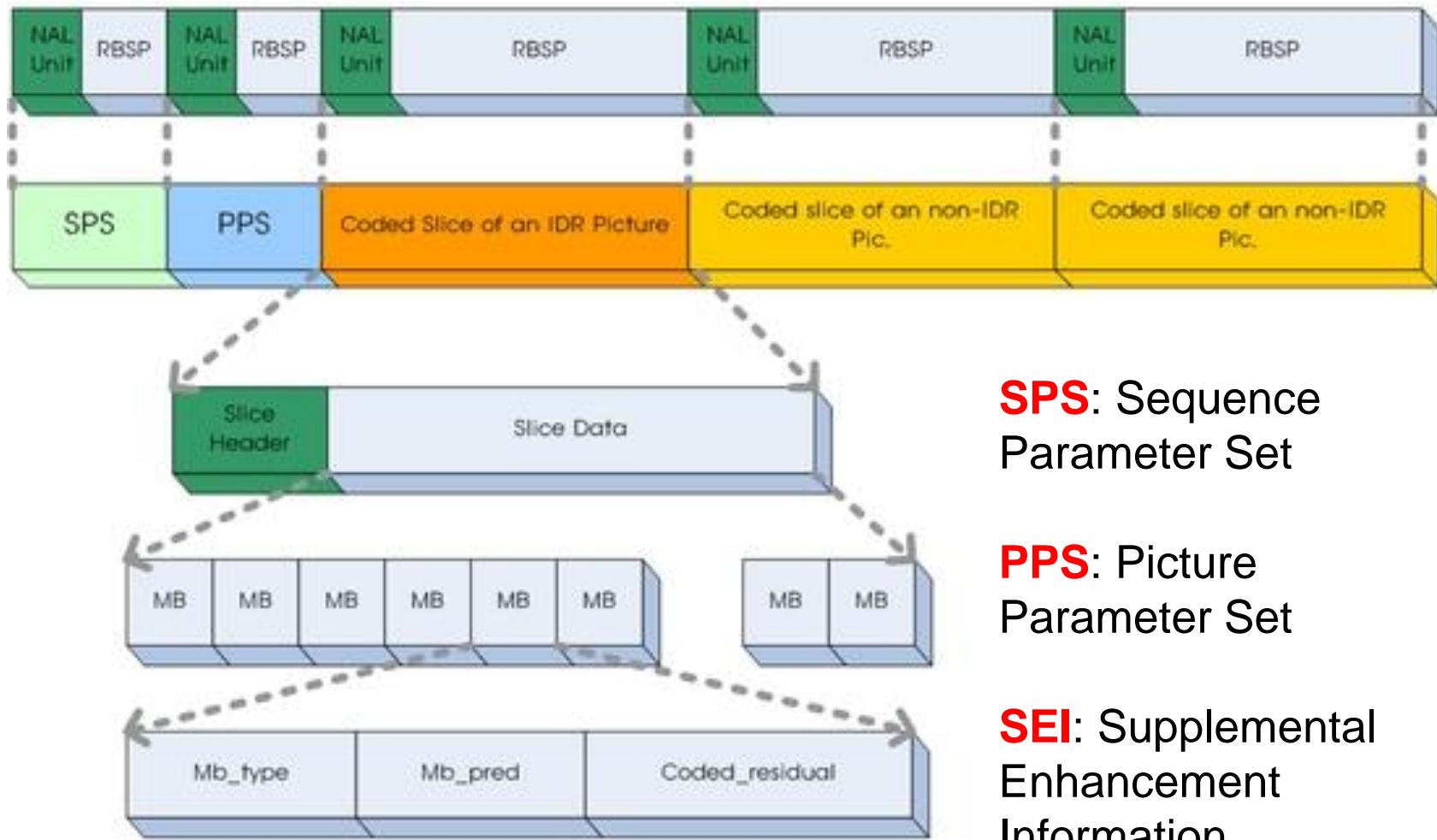
# H.264码流内容

◆ IDR: 一个序列的第一个图像叫做 IDR 图像（立即刷新图像）





# 含参数、图像数据的流格式示意





# NAL Unit 头标 (NALU类型)

◆ NALU 头由一个字节组成

<b>F(1bit)</b>	<b>NRI(2bits)</b>	<b>Type(5bits)</b>
----------------	-------------------	--------------------

◆ F: forbidden\_zero\_bit

□ H.264中规定必须为 0

◆ NRI: nal\_ref\_idc

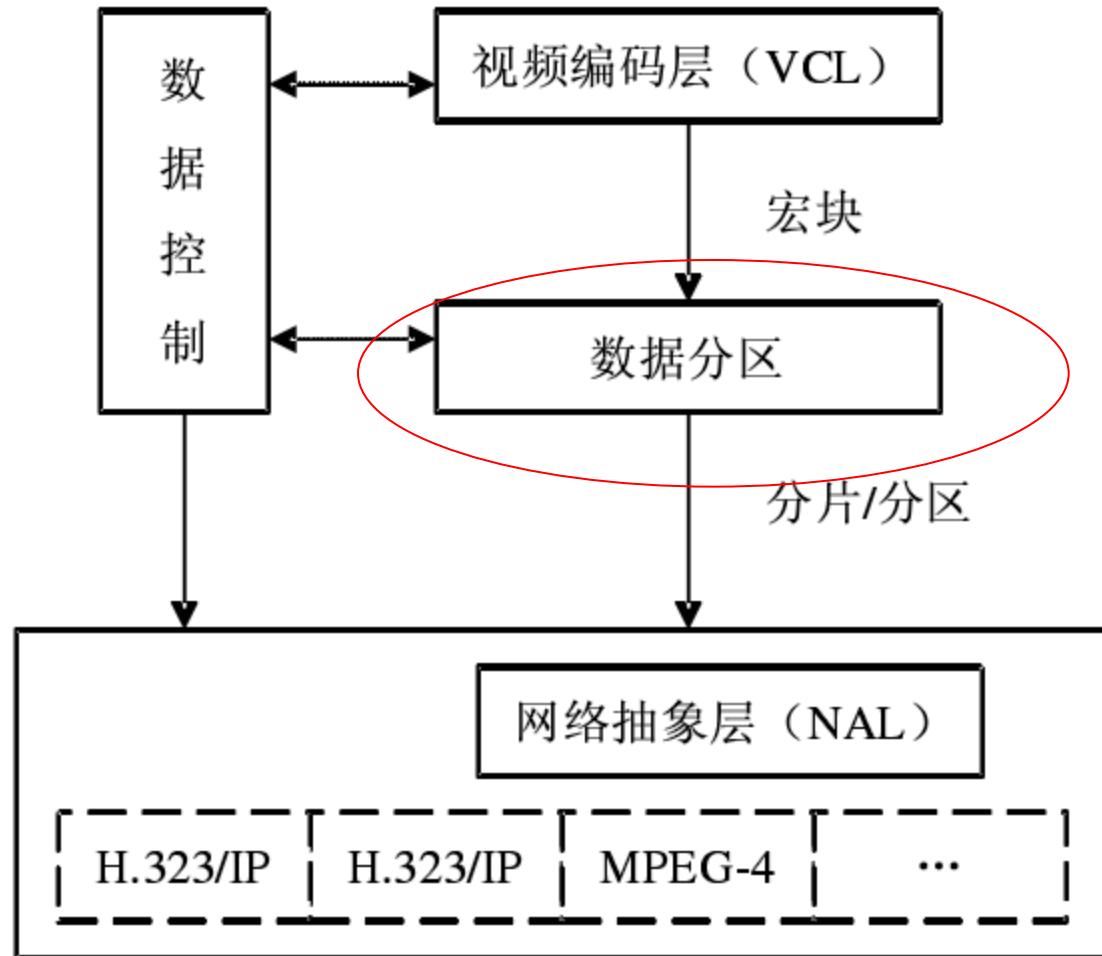
□ 指示这个 NALU 的重要性

## Type 定义

- 0: 未规定
- 1: 非IDR图像中不采用数据分区的片段
- 2: 非IDR图像中A类**数据分区**片段
- 3: 非IDR图像中B类数据分区片段
- 4: 非IDR图像中C类数据分区片段
- 5: IDR图像的片段
- 6: 补充增强信息 (SEI: Supplemental Enhancement Information)
- 7: 序列参数集(SPS: Sequence Parameter Set)
- 8: 图像参数集(PPS: Picture Parameter Set)
- 9: 分割符(PD, Picture Delimiter)
- 10: 序列结束符(End of sequence)
- 11: 流结束符(End of stream)
- 12: 填充数据
- 13 – 23: 保留
- 24 – 31: 未规定



# 数据分区 Data Partition



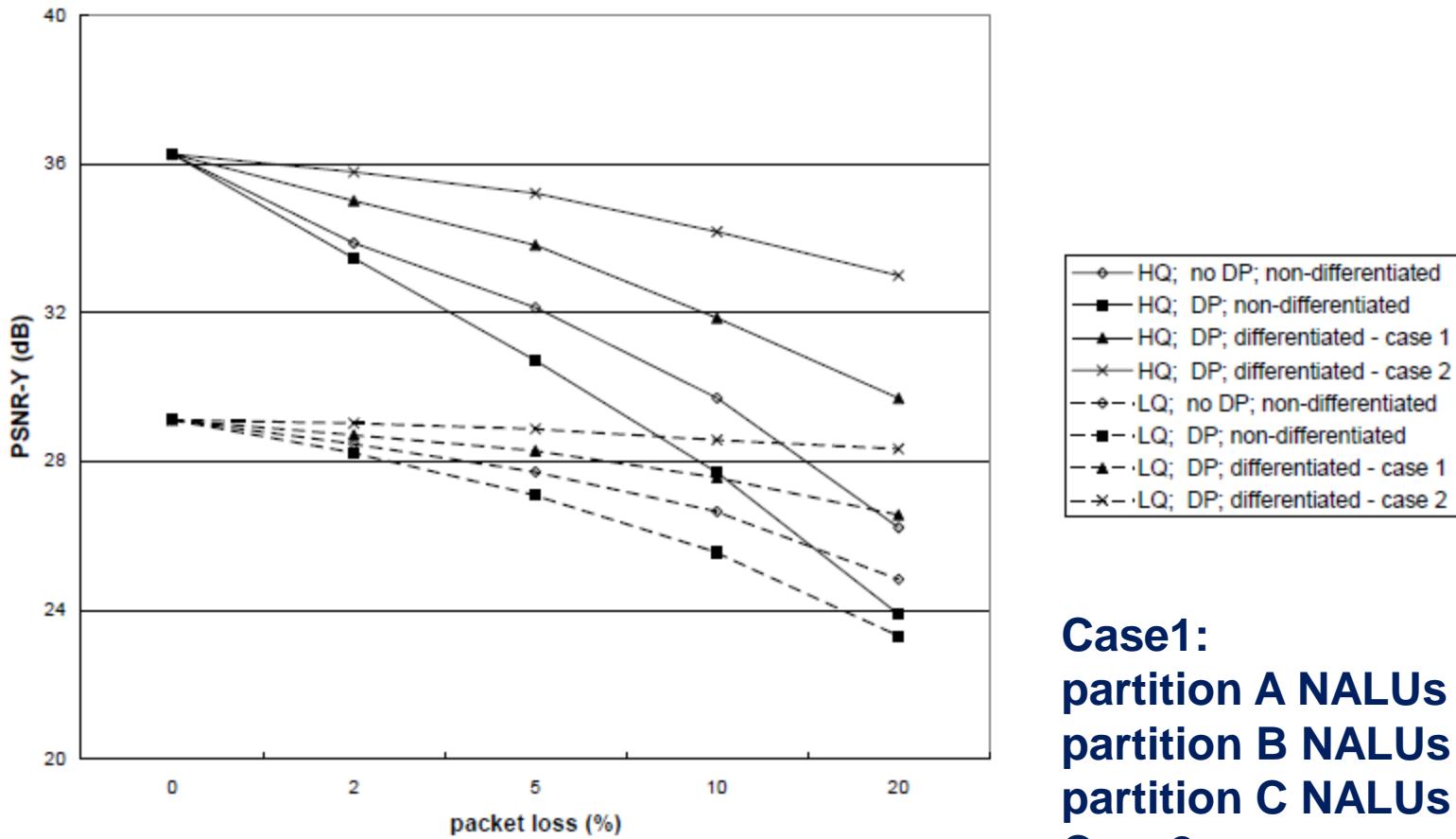


# 三类数据分区重要性不同

**Loss of partition B's or C's -> video degradation**  
**Loss of partition A's or IDR -> the frame dropped**

数据分类	句法单元	句法单元含义
A 类数据	SE_MBTYPE	宏块类型
	SE_INTRAPREDMODE	帧内预测模式
	SE_REFFRAME	参考帧
	SE_MVD	运动矢量信息
	SE_BFRAME	属于 B 帧的所有宏块信息
	SE_SPFRAME	属于 SP 帧的所有宏块信息
B 类数据	SE_CBPI_NTRA	帧内编码数据块编码方式
	SE_LUM_DC_INTRA	帧内编码宏块亮度直流变换系数
	SE_CHR_DC_INTRA	帧内编码宏块色度直流变换系数
	SE_LUM_AC_INTRA	帧内编码宏块亮度交流变换系数
	SE_CHR_AC_INTRA	帧内编码宏块色度交流变换系数
C 类数据	SE_CBPI_INTER	帧间编码数据块编码方式
	SE_LUM_DC_INTER	帧间编码宏块亮度直流变换系数
	SE_CHR_DC_INTRA	帧间编码宏块色度直流变换系数
	SE_LUM_AC_INTER	帧间编码宏块亮度交流变换系数
	SE_CHR_AC_INTER	帧间编码宏块色度交流变换系数

# Data Partitioning 性能



## Case1:

partition A NALUs lost 5%  
partition B NALUs lost 25%  
partition C NALUs lost 70%

## Case2:

partition A NALUs lost 1%  
partition B NALUs lost 5%  
partition C NALUs lost 94%



# HEVC对H.264流封装的改进

- ◆ The high-level syntax architecture used in the H.264/MPEG-4 AVC standard has generally been retained, including the following features.
  - Parameter set structure
  - NAL unit syntax structure
  - Slices
  - Supplemental enhancement information (SEI) and video usability information (VUI) metadata
- ◆ Few new features are introduced in the HEVC standard to enhance the **parallel processing** capability or modify the structuring of slice data for **packetization** purposes.
  - Tiles
  - Wavefront parallel processing
  - Dependent slice segments



# 小结：Network Friendly H.264

## ◆ H.264特性

- Motion Compensation、Intra-frame Prediction, De-blocking Filter, Transform/Scal./Quant, Entropy Coding
- Data partition

## ◆ NALU的不同类型

- 数据分区(**Data Partition**)：A类含有数据片和宏块的头部信息、运动矢量信息以及宏块类型信息等；B类含有帧内编码数据块的编码方式信息和帧内变换系数等；C类含有帧间编码数据块的编码方式信息和帧间变换系数等



# 专题小结 (2-1)

- ◆ CD-DA: EFM8-14、每扇区1/75秒
- ◆ MP3: Tag、 ID3/ APE/Lyrics 、 Frame Header
- ◆ DAB: MP2
- ◆ HDTV: ES、 PES、 PS、 TS
- ◆ MPEG-4 AVC / H.264
  - VCL、 NAL、 RBSP、 SPS、 PPS、 SEI
  - SODB→RBSP→NALU



# 专题小结 (2-2)

- ◆ CD-DA (单一音频流) → CD-ROM
  - 光盘物理格式与逻辑格式分离
- ◆ MP3 (音频+附加信息, 存储)
  - 良好的组织结构提供无限的扩展能力
- ◆ DAB (音频+附加信息, 流)
  - 面向存储与面向流在数据组织结构方面有显著差异
- ◆ HDTV (音频+视频+附加信息, 复用)
  - 音视频流的封装 → 所有类型数据的封装
- ◆ MPEG-4 AVC / H.264 (什么是流? )
  - 网络友好的数据封装

谢谢大家



《多媒体通信》课程专题  
主流音视频流结构：裸数据的组织