**一、TS 格式标准介绍**

TS是一种音视频封装格式，全称为MPEG2-TS。其中TS即"Transport Stream"的缩写。

先简要介绍一下什么是MPEG2-TS：

DVD的音视频格式为MPEG2-PS，全称是Program Stream。而TS的全称则是Transport Stream。MPEG2-PS主要应用于存储的具有固定时长的节目，如DVD电影，而MPEG-TS则主要应用于实时传送的节目，比如实时广播的电视节目。这两种格式的主要区别是什么呢？简单地打个比喻说，你将DVD上的VOB文件的前面一截cut掉（或者干脆就是数据损坏），那么就会导致整个文件无法解码了，而电视节目是你任何时候打开电视机都能解码（收看）的。

**所以，MPEG2-TS格式的特点就是要求从视频流的任一片段开始都是可以独立解码的。**

我们可以看出，TS格式是主要用于直播的码流结构，具有很好的容错能力。通常TS流的后缀是.ts、.mpg或者.mpeg，多数播放器直接支持这种格式的播放。TS流中不包含快速seek的机制，只能通过协议层实现seek。HLS协议基于TS流实现的。

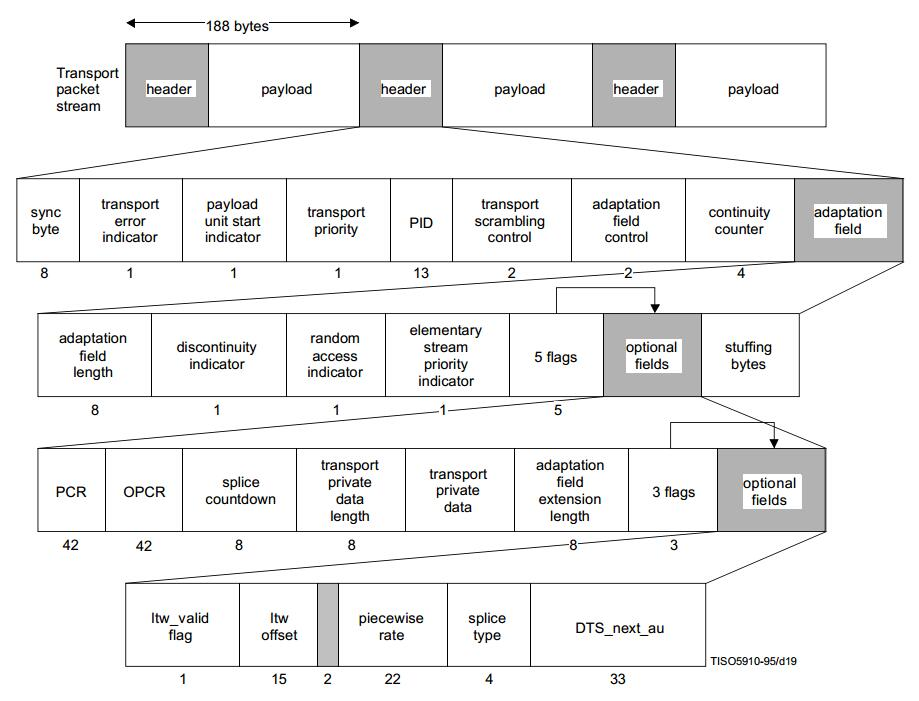
TS格式分析工具：链接: <https://pan.baidu.com/s/1mXPIyTt6dzuDUaTRqRMgCw> 提取码: je5m

**二、TS 格式详解**

TS文件（流）可以分为三层：TS层（Transport Stream）、PES层（Packet Elemental Stream）、ES层（Elementary Stream）。

ES层就是音视频数据，PES层是在音视频数据上加了时间戳等对数据帧的说明信息，TS层是在PES层上加入了数据流识别和传输的必要信息。TS文件（码流）由多个TS Packet组成的。

下图是TS文件（码流）的分层结构图：



原图可以在：[https://github.com/renhui/Thinking-in-AV/blob/master/多媒体格式/TS/1.TS分层结构.jpg](https://github.com/renhui/Thinking-in-AV/blob/master/%E5%A4%9A%E5%AA%92%E4%BD%93%E6%A0%BC%E5%BC%8F/TS/1.TS%E5%88%86%E5%B1%82%E7%BB%93%E6%9E%84.jpg" \t "_blank) 查看。

**三、TS层**

TS包大小固定为188字节，TS层分为三个部分：TS Header、Adaptation Field、Payload。

TS Header固定4个字节；Adaptation Field可能存在也可能不存在，主要作用是给不足188字节的数据做填充；Payload是PES数据。

**1. TS Header**

TS包的包头提供关于传输方面的信息。

TS包的包头长度不固定，前4个字节是固定的，后面可能跟有自适应字段（适配域）。4个字节是最小包头。

包头的结构体字段如下：

* sync\_byte（同步字节）：固定为0x47;该字节由解码器识别，使包头和有效负载可相互分离。
* transport\_error\_indicator（传输错误标志）：‘1’表示在相关的传输包中至少有一个不可纠正的错误位。当被置1后，在错误被纠正之前不能重置为0。
* payload\_unit\_start\_indicator（负载起始标志）：为1时，表示当前TS包的有效载荷中包含PES或者PSI的起始位置；在前4个字节之后会有一个调整字节，其的数值为后面调整字段的长度length。因此有效载荷开始的位置应再偏移1+[length]个字节。
* transport\_priority（传输优先级标志）：‘1’表明当前TS包的优先级比其他具有相同PID， 但此位没有被置‘1’的TS包高。
* PID：指示存储与分组有效负载中数据的类型。
* transport\_scrambling\_control（加扰控制标志）：表示TS流分组有效负载的加密模式。空包为‘00’，如果传输包包头中包括调整字段，不应被加密。其他取值含义是用户自定义的。
* adaptation\_field\_control（适配域控制标志）：表示包头是否有调整字段或有效负载。‘00’为ISO/IEC未来使用保留；‘01’仅含有效载荷，无调整字段；‘10’ 无有效载荷，仅含调整字段；‘11’ 调整字段后为有效载荷，调整字段中的前一个字节表示调整字段的长度length，有效载荷开始的位置应再偏移[length]个字节。空包应为‘10’。
* continuity\_counter（连续性计数器）：随着每一个具有相同PID的TS流分组而增加，当它达到最大值后又回复到0。范围为0~15。

**2. TS Adaptation Field**

Adaptation Field的长度要包含传输错误指示符标识的一个字节。

PCR是节目时钟参考，PCR、DTS、PTS都是对同一个系统时钟的采样值，PCR是递增的，因此可以将其设置为DTS值，音频数据不需要PCR。

打包TS流时PAT和PMT表是没有Adaptation Field的，不够的长度直接补0xff即可。

视频流和音频流都需要加adaptation field，通常加在一个帧的第一个ts包和最后一个ts包里，中间的ts包不加。

**3. TS Payload**

TS包中Payload所传输的信息包括两种类型：视频、音频的PES包以及辅助数据；节目专用信息PSI。

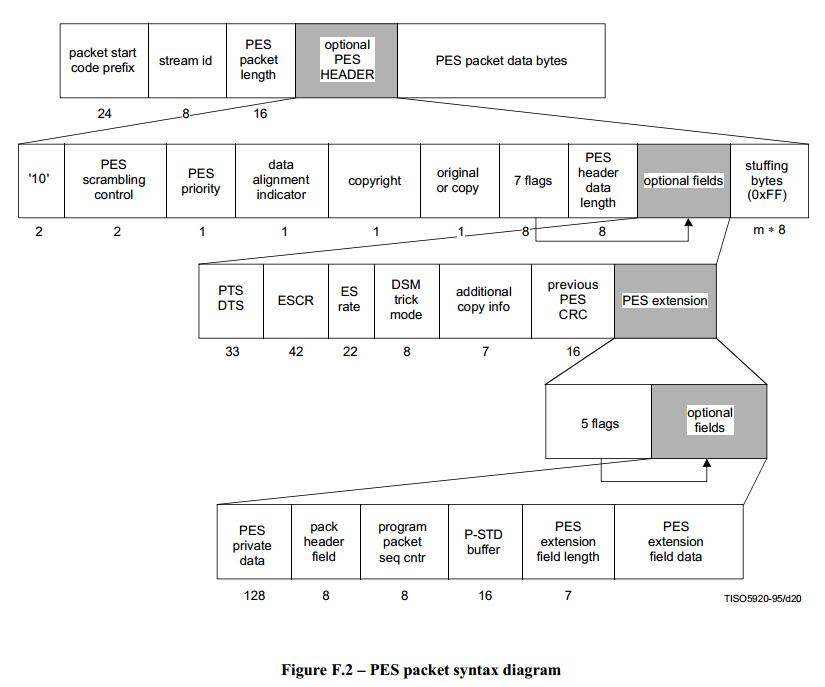
TS包也可以是空包。空包用来填充TS流，可能在重新进行多路复用时被插入或删除。

视频、音频的ES流需进行打包形成视频、音频的 PES流。辅助数据（如图文电视信息）不需要打成PES包。

**四、PES层 & ES 层**

**1. PES层**

PES结构如图：



从上面的结构图可以看出，PES层是在每一个视频/音频帧上加入了时间戳等信息，PES包内容很多，下面我们说明一下最常用的字段：

* pes start code：开始码，固定为0x000001。
* stream id：音频取值（0xc0-0xdf），通常为0xc0；视频取值（0xe0-0xef），通常为0xe0。
* pes packet length：后面pes数据的长度，0表示长度不限制，只有视频数据长度会超过0xffff。
* pes data length：后面数据的长度，取值5或10。
* pts：33bit值
* dts：33bit值

关于时间戳PTS和DTS的说明：

1. PTS是显示时间戳、DTS是解码时间戳。
2. 视频数据两种时间戳都需要，音频数据的PTS和DTS相同，所以只需要PTS。

有PTS和DTS两种时间戳是B帧引起的，I帧和P帧的PTS等于DTS。如果一个视频没有B帧，则PTS永远和DTS相同。

从文件中顺序读取视频帧，取出的帧顺序和DTS顺序相同。DTS算法比较简单，初始值 + 增量即可，PTS计算比较复杂，需要在DTS的基础上加偏移量。

音频的PES中只有PTS（同DTS），视频的I、P帧两种时间戳都要有，视频B帧只要PTS（同DTS）。

**2. ES 层**

ES层指的就是音视频数据。

一般的，视频为H.264视频，音频为AAC音频。

**五、TS流生成及解析流程**

**1. TS 流生成流程**

* 将原始音视频数据压缩之后，压缩结果组成一个基本码流（ES）。
* 对ES（基本码流）进行打包形成PES。
* 在PES包中加入时间戳信息(PTS/DTS)。
* 将PES包内容分配到一系列固定长度的传输包（TS Packet）中。
* 在传输包中加入定时信息(PCR)。
* 在传输包中加入节目专用信息(PSI) 。
* 连续输出传输包形成具有恒定比特率的MPEG-TS流。

**2. TS 流解析流程**

* 复用的MPEG-TS流中解析出TS包；
* 从TS包中获取PAT及对应的PMT；
* 从而获取特定节目的音视频PID；
* 通过PID筛选出特定音视频相关的TS包，并解析出PES；
* 从PES中读取到PTS/DTS，并从PES中解析出基本码流ES；
* 将ES交给解码器，获得压缩前的原始音视频数据。