Atitit 获取mp3音乐文件的音乐名与歌手结构化元数据 nlp java

目录

[1.1. 一、MP3文件的元数据 1](#_Toc14711)

[1.2. MP3文件的数据结构以及为mp3内嵌歌词的代码 3](#_Toc5037)

[1.3. MP3文件是由帧构成，帧是 MP3文件的最小组成单位。 3](#_Toc29500)

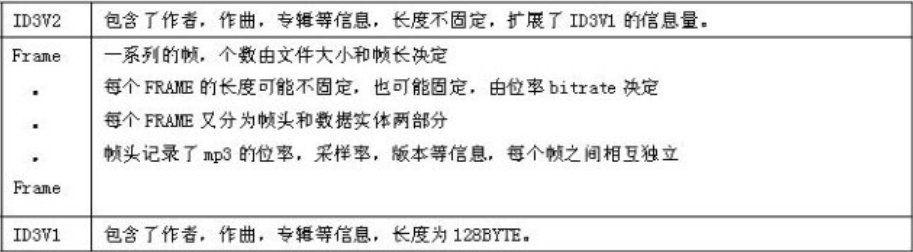
[1.4. 标签帧，而只有数据帧　　APEV2 是最新出现的一种标签， 3](#_Toc30317)

[1.5. 值得一提的是 Lyrics3v2，它是千千静听播放器发明的一种独立帧， 3](#_Toc493)

## 一、MP3文件的元数据

一个规则的MP3文件大致含有3个部分:

* TAG\_V2(ID3V2)
* Frame
* TAG\_V1(ID3V1)



* TAG\_V1的长度是固定的,128byte.期中包含MP3文件的基本信息.
* TAG\_V2扩展了TAG\_V1，TAG\_V2的长度不是固定的，包含了众多关于MP3文件信息

TAG\_V1中包含的信息  
TAG\_V1部分是MP3文件的最后128byte的内容.期中包括的信息有:

* 标签头"TAG" 3字节
* 标题 30字节
* 作者 30字节
* 专辑 30字节
* 出品年份 4字节
* 备注信息 28字节
* 保留 1字节
* 音轨 1字节
* 类型 1字节

常见到有人说获取不到MP3信息,或者获取到的信息是一堆乱码.

解决办法将TAG\_V1中的字节信息转换成字符串时没有指定字符编码(一般用"utf-8")而出现的;

如果想获取关于MP3文件的更多信息,可以通过解析TAG\_V2中的字节来获取.

表1：ID3V1结构  
--------------------------------------------------------------------  
名称　字节 　　说明  
--------------------------------------------------------------------  
Tag　　　3 　　ID3V1标识符“TAG”的Ascii码  
Title　　30　　歌曲名  
Artist　 30　　歌手名  
Album　　30　　专辑名  
Year　　 4 　　日期信息  
Comment　28　　注释信息，有时为30字节  
Reserved 1 　　＝0说明有音轨，下一字节就是音轨；≠0表示注释是30个字节  
Track　　1 　　音轨（字节型数值），歌曲在专辑里的序号   
Genre　　1 　　歌曲风格（字节型数值）

说明：  
　　①如果MP3的注释＝30字节，那么就要占用 Reserved 和 Track 两个字节，这要看 Reserved 是否＝0，如果＝0，那么注释有 28 个字节。如果不是，那么注释有 30 个字节。当注释＝30 个字节的时候，那就没有 Track 了。  
　　②如果 MP3 文件后面虽然有“TAG”三个字母，但字母后面全是0，那就不是一个合法的 ID3V1 信息，应该认为没有 ID3V1 信息。  
　　③ID3V1 的各项信息都是顺序存放，没有任何标识将其分开，一般用 0补足规定的长度。比如歌曲名有 20 个字节，则在歌曲名后要补足 10 个 0，否则将造成信息错误。  
　　④歌曲风格共 148 种，用编号表示，表2列出了前 30 种的风格与编号对照，详情可上网查询。

## MP3文件的数据结构以及为mp3内嵌歌词的代码

## MP3文件是由帧构成，帧是 MP3文件的最小组成单位。

根据帧性质的不同，文件大体分为四个部分：ID3v2标签帧、数据帧、APEV2标签帧、ID3v1标签帧，而只有数据帧才是必需的。  
　　数据帧包含了歌曲的压缩数据。标签帧提供了歌曲的演唱者、歌名、专辑、年份等信息。  
　　ID3v1 在文件结尾，以字符串“TAG”为标识，其长度是固定的 128 个字节。  
　　ID3v2 在文件头，以字符串“ID3”为标识，长度不固定，扩展了 ID3V1 的信息量。

## 标签帧，而只有数据帧　　APEV2 是最新出现的一种标签，

以字符串“APETAGEX”为标识，长度不固定，位置也不固定，可能在文件末尾也可能在文件头，比较常见的是位于文件尾部, 但在 ID3v1 之前。

## 值得一提的是 Lyrics3v2，它是千千静听播放器发明的一种独立帧，

位于 ID3v1 之前，APEV2 之后（如果有 APEV2 的话），它专用于内嵌 Lrc 类型的歌词文件。

(9+条消息)操作MP3文件的元数据 - weixin\_34235105的博客 - CSDN博客.html