

MPEG 简介 + 如何计算 CBR/VBR MP3 的播放时间

版本：**1.4**

作者：crifan

邮箱：green-waste (at) 163.com

版本历史

版本	日期	内容说明
1.0	2009-09-19	简介 MPEG 相关知识 详细介绍如何计算 CBR 和 VBR 的 MP3 的播放时间
1.4	2011-04-24	修正了 VBR 播放时间的计算公式（原中文说明部分有误） 添加了一些 MP3 相关的知识点的解释 调整了排版格式

目录

1.	正文之前	4
1.1.	声明	4
1.2.	此文目的	4
2.	MPEG 的相关知识.....	5
2.1.	MPEG 是啥.....	5
2.2.	为啥没有了 MPEG-3.....	5
2.3.	MPEG2.5 又是啥	5
2.4.	MPEG 中不同帧之间的关系.....	6
2.5.	MPEG-1 和 MPEG-2 音频特性	6
2.5.1.	MPEG-1 音频 (ISO/IEC 11172-3)	6
2.5.2.	MPEG-2 音频 (ISO/IEC 13813-3)	6
2.6.	什么是 ISO/IEC 11172-3 和 ISO/IEC 13818-3	6
3.	MP3 中常见的术语	8
3.1.	啥叫 MP3.....	8
3.2.	什么是 LSF.....	8
3.3.	什么是帧 (frame)	8
3.4.	什么是帧头 (Frame Header)	8
3.5.	啥是 CBR 和 VBR.....	8
3.6.	比特率 (Bitrate)	9
3.7.	边信息 (Side Information)	9
3.8.	MP3 的 TAG	9
4.	MPEG 的帧 Frame	10
4.1.	MPEG 帧头 (Frame Header)	10
4.1.1.	举例说明 MPEG 帧头的含义.....	12
4.2.	MPEG 音频数据	12
4.3.	MPEG 帧的大小.....	13
4.3.1.	常见 MPEG 帧的音频数据大小是 418 字节.....	14
5.	MP3 的播放时间的计算公式及 XING 和 VBRI 头介绍.....	16
5.1.	CBR 的 MP3 的播放时间(duration)计算方法	16
5.2.	VBR 的 MP3 的播放时间(duration)计算方法.....	16
5.2.1.	平均比特率法.....	16
5.2.2.	总帧数法.....	17
5.2.3.	VBR 的两种 Header: XING 和 VBRI.....	18
5.2.4.	Xing TAG / Xing 头 (header)	18
5.2.5.	VBRI 头 (header)	19
6.	计算 CBR 和 VBR 的 MP3 文件的播放时间的步骤.....	21

6.1.	定位到 MPEG 的帧头的位置.....	21
6.2.	解析 MPEG 帧头，获取必要信息.....	21
6.3.	判断是 VBR 还是 CBR，根据公式计算播放时间.....	21
6.3.1.	定位出 Xing 头的位置.....	21
6.3.2.	如果有 Xing 头，则是 VBR，解析 XING 头，计算播放时间.....	22
6.3.3.	如果没 XING 头，定位出 VBRI 头位置，找 VBRI 头.....	22
6.3.4.	如果也没找到 VBRI 头，则是 CBR，计算 CBR 的播放时间.....	22
7.	MP3 相关的一些知识点的解释.....	24
7.1.	MP3 的文件的内容组织结构.....	24
7.2.	MP3 帧的时长是 26ms 的来历.....	25
7.3.	怎么算出来 MP3 的压缩比大概是 1:11 的.....	25
7.4.	问：每一个帧都包含帧头，会不会很浪费空间？.....	26
7.4.1.	原因之一.....	26
7.4.2.	原因之二.....	26
8.	后记.....	27
9.	文章引用.....	28

表格

表 1	MPEG 音频的帧头的格式.....	10
表 2	MPEG 比特率索引表（单位：Kbit/s）.....	11
表 3	MPEG 帧的采样率索引表（单位：Hz）.....	11
表 4	MPEG 帧的采样数索引表（单位：个/帧）.....	12
表 5	XING 头的格式及含义.....	19
表 6	VBRI 头的格式及含义.....	20
表 7	MPEG Layer III 的边信息（side information）（单位：字节）.....	21
表 8	MP3 文件的内容组织结构.....	24

图片：

图表 1	MPEG 帧头含义举例.....	12
------	------------------	----

1. 正文之前

1.1. 声明

- 本文所写内容，多数整理自互联网，版权归原作者所有
- 笔者知识有限，文中难免有误，欢迎批评指正，[green-waste \(at\) 163.com](mailto:green-waste@163.com)
- 欢迎盗版，盗版不究，但请转载时注明原作者

1.2. 此文目的

- 了解 MPEG 相关知识
- 了解 MP3 的常见术语的含义
- 详解 VBR MP3 的帧头格式及含义
- 搞懂如何去计算 CBR 和 VBR 的 MP3 文件的播放时间（duration）

2. MPEG 的相关知识

想要了解如何计算 VBR 的 MP3 的播放时间之前,要了解一些和 MP3 相关的一些基本概念,其中主要是 MPEG 的相关知识和编解码的一些基础知识。

2.1. MPEG 是啥

MPEG 全名 Moving Pictures Experts Group, 动态图像专家组, 简单说就是一个专家组, 专门研究一些音视频规范的, 所以才叫专家, 不是我们国家的“砖家”哦。这个专家组是在 ISO/IEC(International Standards Organization/International Electrotechnical Commission, 国际标准化组织/国际电工委员会) 联合指导下成立的。这个组, 专门去研究出一个数字音视频的压缩相关的规范, 所以最后研究出适用于不同应用环境的 N 多规范。

和事物发展的过程类似, 研究出这么多的规范也是, 不同时期, 不同的版本, 针对不同的应用。也是由简到繁。并且, 命名规则都是, 按照阿拉伯数字从小到大的: MPEG 1, MPEG2, MPEG4, MPEG-7, 最新版本, 好像都有 MPEG-21 了。

2.2. 为啥没有了 MPEG-3

估计有人纳闷了, 中间的 MPEG3 咋没了呢?

是没 MPEG3, 当然, 不是被刘谦变魔术变没了, 而是由于当时设计者没有规划好, 导致已经设计好的 MPEG2, 性能太好了, 都能干本来打算让 MPEG3 干的活了, 所以后来干脆就不去再设计 MPEG3 了, 原定计划就取消了, 也就没了 MPEG3。看来这个 MPEG3, 待遇貌似不比胎死腹中好多少。对于很多人误解的, 以为 MP3 就是 MPEG-3, 也就错的离谱了。关于 MP3 的名称来历, 下面会再解释。

2.3. MPEG2.5 又是啥

MPEG2.5, 简单说就是出身不正, 不是官方推出的规范。

MPEG 2.5 是针对 MPEG2 的一个非官方的扩展版本, 支持更低的采样率。

关于其更多解释, 网上找到这些:

MPEG 声音标准提供三个独立的压缩层次: Layer I、Layer II 和 Layer III。

用户具体选哪个 Layer, 可以根据自己的要求, 在权衡复杂性和声音质量之后, 做出自己的选择。

- **Layer I** 的编码器最为简单, 编码器的输出数据率为 **384 kb/s**, 主要用于小型数字盒式磁带(digital compact cassette, DCC)。
- **Layer II** 的编码器的复杂程度属中等, 编码器的输出数据率为 **256 kb/s~192 kb/s**, 其应用包括数字广播声音(digital broadcast audio, DBA)、数字音乐、CD-I(compact disc-interactive)和 VCD(video compact disc)等。
- **Layer III** 的编码器最为复杂, 编码器的输出数据率**为 64 kb/s**, 主要应用于 ISDN 上的声音传输。

对于 Layer III:

- MPEG-1 Layer III 支持的采样率为 32,44.1,48khz,比特率支持 32---320kbps
- MPEG-2 Layer III 支持的采样率为 16,22.05,24khz,比特率支持 8---160kbps

Fraunhofer 对此又进行扩展，将原来 MPEG-2 所支持的低采样率再除以 2，得到：8, 11.025, 和 12 kHz，比特率跟 MPEG-2 相同，称为 "MPEG 2.5"。

2.4. MPEG 中不同帧之间的关系

对于 Layer I 和 Layer II，不同的帧之间，是互相独立的。也就意味着，你可以任意截取 MPEG 的音频文件，然后找到第一个正确的帧头，后开始解码，对于余下的帧，都是同样的处理，解码后，进行播放，这样都可以正确的播放。

而对于 Layer III，所有帧不保证都是互相独立的。此处不保证完全独立，指的是当前的帧和前后临近的那些帧，不一定完全独立，有可能有一定关系。这是由于可能用到“字节蓄水池（byte reservoir）”的技术，即内部的一个数据缓存，当前帧和其后的一些帧，都是相关的，最差情况下，要连续缓冲保存 9 个帧，才能对第一帧解码。

2.5. MPEG-1 和 MPEG-2 音频特性

2.5.1. MPEG-1 音频 (ISO/IEC 11172-3)

MPEG-1 音频 (ISO/IEC 11172-3) 描述了三层音频编码，具有如下特性：

- 一个或两个音频声道
- 采样率 32KHz、44.1KHz 或 48KHz
- 比特率从 32Kbps 到 448Kbps
- 每一层都有其自己的其他特点。

2.5.2. MPEG-2 音频 (ISO/IEC 13813-3)

MPEG-2 音频 (ISO/IEC 13813-3) 包含了对 MPEG-1 的两种扩展。通常称为 MPEG-2/低采样率 (LSF) 和 MPEG-2/多声道 (Multichannel)。

2.5.2.1. MPEG-2/LSF 的特性

1. 一个或两个音频声道
2. 采样率只有 MPEG-1 的一半
3. 比特率从 8Kbps 到 256Kbps

2.5.2.2. MPEG-2/多声道 的特性

- 多达 5 个全范围的音频声道和一个 LFE (Low Frequency Enhancement, 也叫做 重低音) 声道
- 采样率和 MPEG-1 相同
- 对于 5.1 声道，最高的比特率可达 1Mbps

2.6. 什么是 ISO/IEC 11172-3 和 ISO/IEC 13818-3

由于 MPEG 只是 ISO/IEC 下面的一个组织，所以，关于 MPEG 音频部分的规范，也都是出自 ISO/IEC 之手。

因此，ISO/IEC 11172 和 ISO/IEC 13818，其实就是 MPEG-1 和 MPEG-2 的别名。

另外, 由于 MPEG-1 和 MPEG-2, 每个都分好几个部分, 其中, 第 3 部分是关于音频(Audio)的。

所以, **ISO/IEC 11172-3** 和 **ISO/IEC 13818-3**, 就分别对应着 **MPEG-1** 的音频, **MPEG-2** 的音频, 也就是我们常常提到的 MPEG 的音频文件所对应的规范。

3. MP3 中常见的术语

知道了 MPEG 的来龙去脉后，在了解 MP3 的播放时间如何计算之前，还要知道其他一些 MP3 相关的知识及常见的术语：

3.1. 啥叫 MP3

注意，这个 MP3，不是 MPEG-3，但是为何叫 MP3，是因为：

MPEG 规范中规定了，每一个版本的 MPEG，比如 MPEG1，MPEG2 等，都有三种不同的 Layer，不同 Layer 的序号命令是以罗马数字的，所以叫做 Layer I，Layer II，Layer III。其中，MPEG-1 或 MPEG-2 的 Layer III，被称为 MP3，而其中最常见的是 MPEG 1 的 Layer III，所以，被大家所熟知的 MP3，一般都指的是 MPEG-1 的 Layer III。

即 MPEG-1 的 Layer III 被简称为 MP3。

根据事物发展由简到繁的规律，我们知道，Layer III，相对 Layer I 和 Layer II，有着更复杂的压缩算法。正是其相对复杂，用了很多算法，比如声学上的掩蔽效应（masking effect），Huffman 压缩等，才使得可以实现，在尽可能保持音质的前提下，极大地减少了音频文件大小，也就是说，尽量让你听上去音频声音和音质都没啥变化，但是 MP3 的文件大小，相对于原先没处理的音频数据或者其他格式的，比如 WAV 格式等，要小很多。

音质满足大家的要求，文件又小，在互联网时代，就非常方便大家互相交流传播。

这也是 MP3 如此流行的主要原因之一。

3.2. 什么是 LSF

MPEG2/2.5 也常被简称为 LSF（Low Sampling Frequencies），低采样率。

3.3. 什么是帧（frame）

帧，即数据帧，通俗点说，就是一段数据，一块数据，数据块。

对于 MPEG 音频文件本身，并没有什么文件头，而是由很多数据块所组成，这样的单个的数据块，就叫做一个（数据/音频）帧（frame）。

而 MP3 文件，就是由很多个帧所组成。

帧，也是其他很多音视频技术中的基本单位。

3.4. 什么是帧头（Frame Header）

每一个帧里面，包含了帧头和音频数据。帧头，就是在帧的头部，有一定长度的数据，用于描述该帧音频的一些参数，用于解码器识别相关音频参数，用于对音频帧解码。

对于 MPEG 的帧头，是固定的 32 比特，即 4 字节。

3.5. 啥是 CBR 和 VBR

CBR（Constant BitRate）固定（/不变）比特率，VBR（Variable BitRate），不定（/可变）比特率。此处的固定与可变，指的是 MP3 的比特率，而不是指采样率。

音频文件可以被编码器编码成 CBR 或 VBR。

VBR 意思就是，每一帧的比特率都不一定相同（当然，很有可能临近的一些帧的比特率相同）。由于 **VBR** 在编码的时候，根据当前采样的声音数据的复杂度，去判断是采用何种比特率。比如对于很复杂的声音，那么就用更多的比特位去编码，如果是很简单的声音甚至是无声（**silence**）的数据，那么就可以用很少的比特位去编码，这样不同情况下，所产生的 **MP3** 音频数据的大小，是不同的，但是却可以一直保持同样的音质。

因此，总体来说，对于同样大小的 **MP3** 文件，**VBR** 的音质一般要比 **CBR** 的好。

很多种音视频文件，都可以按照 **VBR** 来压缩，比如 **MP3**、**WMA**、**OGG Vorbis**、**AAC**，**MPEG-2** 的视频等。

VBR 的优点是，和 **CBR** 相比，用更小的空间，即文件更小，实现更高的音质。缺点是，编码复杂度增加，编码和解码都需要更长时间，而且很早之前，有些硬件编码器可能和 **VBR** 不兼容。不过现在一般硬件的音频解码器，都可以很好的兼容 **VBR** 了。

3.6. 比特率 (Bitrate)

即每秒包含了多少个比特的数据。比特率常用 **Kbps**（kilo bits per second，千比特每秒）表示。注意，此处的千比特=1000 bits，而不是 1024。

3.7. 边信息 (Side Information)

在 **MPEG** 音频的帧头的后面，有一些解码器会用到的一些信息，用于解码器控制音频流的播放，它就叫做 边信息。

3.8. MP3 的 TAG

MP3 文件中，会在一些位置（文件最开始或者是最后，或者其他某个固定位置），存放一些 **TAG**，即标签，用于描述文件的相关信息，其中最常见的一些，比如，**MP3** 音乐的歌手名，专辑名，专辑年份，曲风等等。

关于 **MP3** 的 **TAG**，现在常见的有 **ID3v1**（**ID3** 的第一版），**ID3v2**（**ID3** 的第二版），**APEv2** 等。

同一文件中，一般只存在某一种 **TAG**，即，是 **ID3v2** 或 **APEv2** 或 **ID2v1** 或其他。

与此相关的应用方面，我们用千千静听去播放 **MP3** 的时候，播放器中点击文件属性，还可以看到有 **MP3** 标签的读取优先级方面的设置，默认设置为 **APEv2 > ID3v2 > ID3v1**。

如果 **MP3** 文件中存在 **ID3** 的 **TAG** 话：

ID3v1 会放在音频文件的最后，大小 128 个字节，其中前三个字节是字符“**TAG**”。

ID3v2 一般放在音频文件的开头处，前三个字节是字符“**ID3**”。

关于 **ID3** 的具体格式，请参考附录中的引用文献。

MP3 的 **TAG**，只是用于存储歌曲等方面的辅助描述信息，与 **MP3** 的解码，没啥关系，故此处不在过多介绍。

4. MPEG 的帧 Frame

MPEG 音频文件，由一个个的帧（Frame）组成。
每一 MPEG 帧都有个帧头(Frame Header)，位于帧的最开始处，接下来的是音频数据(Audio Sample/Audio Data)，即：

MPEG 帧 = MPEG 帧头 + MPEG 音频数据

下面分别详细介绍 MPEG 帧头和音频数据的细节内容：

4.1. MPEG 帧头 (Frame Header)

MPEG 音频的三种 Layer 的，尽管他们的压缩方法各不相同，但是帧头格式都一样。

先说一下大小，MPEG 帧头，共 32bit=4 字节。

然后再看具体的格式及含义，如下所示：

表 1 MPEG 音频的帧头的格式

位置 (bit)	长度 (bit)	含义	示例								
0	11	用于同步帧，找到此帧头（所有位均置 1）	1111 1111 111								
11	2	MPEG 音频的版本 ID <table><tr><td>00</td><td>MPEG 2.5 (MPED-2 的非官方扩展版本)</td></tr><tr><td>01</td><td>保留</td></tr><tr><td>10</td><td>MPEG 2 (ISO/IEC 13818-3)</td></tr><tr><td>11</td><td>MPEG 1 (ISO/IEC 11172-3)</td></tr></table>	00	MPEG 2.5 (MPED-2 的非官方扩展版本)	01	保留	10	MPEG 2 (ISO/IEC 13818-3)	11	MPEG 1 (ISO/IEC 11172-3)	11
00	MPEG 2.5 (MPED-2 的非官方扩展版本)										
01	保留										
10	MPEG 2 (ISO/IEC 13818-3)										
11	MPEG 1 (ISO/IEC 11172-3)										
13	2	Layer 的索引 <table><tr><td>00</td><td>保留</td></tr><tr><td>01</td><td>Layer III</td></tr><tr><td>10</td><td>Layer II</td></tr><tr><td>11</td><td>Layer I</td></tr></table>	00	保留	01	Layer III	10	Layer II	11	Layer I	01
00	保留										
01	Layer III										
10	Layer II										
11	Layer I										
15	1	保护位 0 – 用 16 位的 CRC 保护下面的帧头 1 – 无 CRC	1								
16	4	比特率索引 (见下表)	1001								
20	2	采样率索引 (见下表)	11								
22	1	填充位 如果设置了此位，就会对每帧数据填充一个 slot (对于帧大小的计算很重要)	0								
23	1	私有位 (仅用于标示性的)	1								

24	2	声道的模式		01
		00	立体声	
		01	混合立体声	
		10	双声道 (两个单声道)	
		11	一个声道 (单声道)	
		注：双声道文件由两个独立的单声道所组成。大多数解码器把双声道输出成立体声，但是实际上，不是所有的双声道都是立体声的。		
26	2	模式扩展 (仅用于联合立体声)（注：此处不是本文重点，故忽略相关的索引表）		00

表 2 MPEG 比特率索引表 (单位: Kbit/s)

比特率索引	MPEG 1			MPEG 2, 2.5 (LSF)	
	Layer I	Layer II	Layer III	Layer I	Layer II & III
0000	空闲				
0001	32	32	32	32	8
0010	64	48	40	48	16
0011	96	56	48	56	24
0100	128	64	56	64	32
0101	160	80	64	80	40
0110	192	96	80	96	48
0111	224	112	96	112	56
1000	256	128	112	128	64
1001	288	160	128	144	80
1010	320	192	160	160	96
1011	352	224	192	176	112
1100	384	256	224	192	128
1101	416	320	256	224	144
1110	448	384	320	256	160
1111	保留				

注:

(1) 上表中:

蓝色部分: 就是我们此处所关心的 MP3 (MPEG-1, Layer III) 的比特率。

红色部分: 即 128kbps, 192kbps, 320kbps 等, 就是我们常见的 MP3 的比特率。

表 3 MPEG 帧的采样率索引表 (单位: Hz)

采样率索引	MPEG 1	MPEG 2 (LSF)	MPEG 2.5 (LSF)
00	44100 Hz	22050 Hz	11025 Hz
01	48000 Hz	24000 Hz	12000 Hz
10	32000 Hz	16000 Hz	8000 Hz
11	保留		

注：

(1) 上表中红色部分，就是我们最常见的 MP3（MPEG-1）的采样率 44100Hz=44.1K Hz

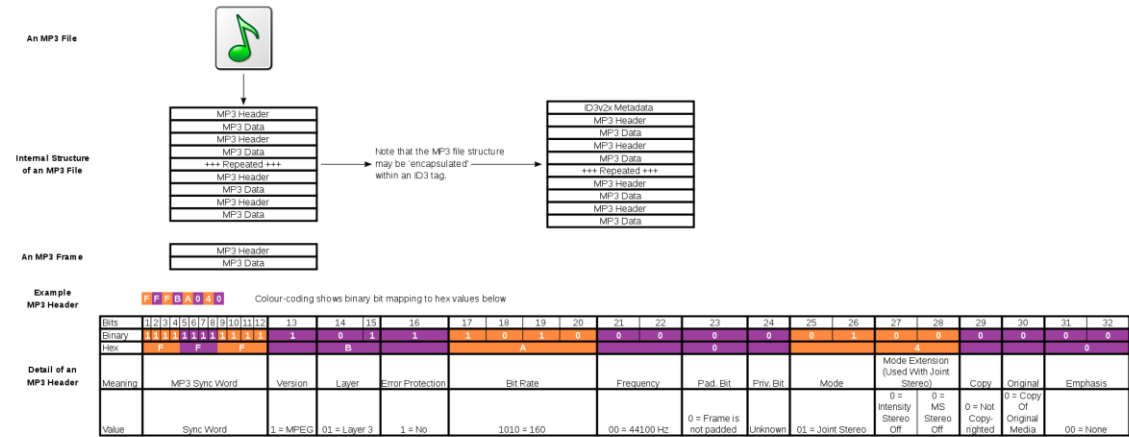
从上面 MPEG 帧头的格式中可以看出，MPEG 的音频，都是有相对固定的比特率，采样率等参数，关于这每一帧的这些参数的具体值，都是找到索引值，然后查对应的索引表，而得知具体的值。

不过，顺便提一点，在 MPEG 标准中，也描述了一种自由格式（free format），这种自由格式意思为用一个固定比特率对文件进行编码，而此固定的比特率不是我们那些索引表中预定义好的。对于这类自由格式的 MPEG 音频，一般的解码器都无法解码。

4.1.1. 举例说明 MPEG 帧头的含义

下图给出了一般 MP3 的格式及举例说明了 MPEG 的帧头所对应的信息：

图表 1MPEG 帧头含义举例



4.2. MPEG 音频数据

MPEG 帧，除了开始部分的 MPEG 帧头外，余下的就是 MPEG 的音频数据。

需要注意的，MP3 中的帧，是 MPEG 帧，其中的音频数据部分，是经过 MP3 的相关算法压缩后的数据，而不是原始采样过来的数据。

MPEG 音频数据部分，包含了 固定数目 的 音频采样（Audio Sample）。

其中关于采样个数：

MPEG 的不同规范（MPEG-1/2/3），以及同一规范中不同的 Layer(Layer I/II/III)，每一帧所对应的采样数，都是固定的，其具体的值参见下表：

表 4 MPEG 帧的采样数索引表（单位：个/帧）

	MPEG 1	MPEG 2 (LSF)	MPEG 2.5 (LSF)
Layer I	384	384	384
Layer II	1152	1152	1152
Layer III	1152	576	576

注：

(1) 上表中红色部分,就是我们此处所关心的, MPEG-1, Layer III, 即 MP3, 不论是 CBR 还是 VBR, 文件中的每一帧, 其采样的个数/采样数, 都是固定的 1152 个。

而后面要介绍如何计算 VBR MP3 的播放时间, 正是基于此前提:

MP3, 即 MPEG-1, Layer III, 不论是 CBR, 还是 VBR, 每一帧的采样个数都是固定的 1152 个。即每一帧, 都是固定的 1152 个采样。

同时, 我们还要注意另外一点, 那就是对于被某个编码器将原始音频数据编码为 MP3 数据后, 得到的 MP3 文件, 对于此单个文件, 其采样率, 始终都是一样的。也就是说, 如果解析 MP3 的第一帧 MPEG 头得到的采样率是 44100Hz 的话, 那么此 MP3 文件后面的所有的帧的采样率, 也肯定都是 44100Hz, 即这个采样率, 对于同一个 MP3 文件来说, 是固定的。

于此相对的是, VBR 中的 Variable Bitrate, 中的 Variable, 可变的, 指的是, 变化的比特率, 而不是采样率。

总结一下就是:

CBR 和 VBR 中的固定和可变, 都是指的是比特率 Bitrate, 而不是采样率 Sample Rate。对于同一 MP3 文件, 不论 CBR 还是 VBR, 采样率都是固定的。

因为该采样率对应着此 MP3 被编码器编码的那一时刻, 编码器的采样率也是最开始时候就设置好, 并且之后不会再变化 (除了你重新录制另外一个 MP3 文件)。

关于这个知识点, 一定要搞清楚, 否则就会出现我最开始遇到的情况, 以为 VBR 的采样率也是变化的, 导致别人问我那么是不是意味着对 VBR MP3 解码, 每一帧都要重新设置解码器的采样率, 如果回答是, 那 mplayer 等常见解码器的代码实现中, 没看到对应设置, 只看到了最开始解码 MP3 时候, 只设置一次其采样率, 其后解码每一帧, 都是没有重新设置采样率的。而实际结果是, VBR 变化的只是比特率, 采样率是固定的, 所以只需要在解码 MP3 最开始的时候设置一次即可。

4.3. MPEG 帧的大小

前面已经解释了, MPEG 帧= 帧头 + 数据。

下面来看看, MPEG 的帧的大小, 即帧头的大小, 加上帧数据的大小。

解释 MPEG 帧大小之前, 先要介绍个名词: Slot, 槽。

MPEG 帧, 由一个个的 Slot (槽) 组成。

Layer I 中, 一个 Slot 是 4 个字节;

Layer II 和 Layer III 中, 一个 Slot 是一个字节。

所以, 此处可以简单的理解为:

MPEG 的 Layer III 中, 帧是有一个个字节所组成。(是不是听起来像句废话, ^_^)

好了, 知道了此处 MPEG 的 Layer III 的帧的基本单位为字节之后, 我们再来看看帧的大小是多少。

首先, 帧头, 不用多说, 都是前面提到的, 固定的 32 比特=4 字节。

其次要好好解释一下 MPEG 帧的音频数据的大小, 可用如下公式计算:

Frame_Data_Size

= Audio_Data_Size + Frame_Padding_Size

= Frame_Time * Frame_Bitrate + Frame_Padding_Size

$$= (\text{Sample_Number} * \text{Time_per_Sample}) * \text{Frame_Bitrate} + \text{Frame_Padding_Size}$$

$$= (\text{Sample_Number} * (1/\text{Sample_Rate})) * \text{Frame_Bitrate} + \text{Frame_Padding_Size}$$

帧数据大小

$$= \text{音频数据大小} + \text{帧的填充大小}$$

$$= \text{帧的时长} * \text{帧的比特率} + \text{帧的填充大小}$$

$$= (\text{采样个数} * \text{每一采样的时长}) * \text{帧的比特率} + \text{帧的填充大小}$$

$$= (\text{采样个数} * (1/\text{采样率})) * \text{帧的比特率} + \text{帧的填充大小}$$

其中：

1. **帧的填充大小**：对于 MPEG 的 Layer III 来说，单位就是字节，不够 8bit 一个字节的话，添加 padding 对应的 bit，凑够一个字节。所以，具体 padding 几个 bit，要看每一帧的数据的 bit 是否是 8 的倍数，如果本身是 8 的倍数，那么 padding 就是 0，如果不是，根据具体情况决定补齐几个 bit。
2. **采样个数**：对于 MP3，即 MPEG-1，Layer III 来说，不论 CBR 还是 VBR，每一帧采样个数都是固定的 **1152 个**。
3. **采样率**：对于 MP3，即 MPEG-1，Layer III 来说，不论 CBR 还是 VBR，对于单个 MP3 文件来说，也是固定的，每一帧采样率，都是一样的。**采样率是多少，通过解析第一帧，即可得知所有帧的采样率。**
4. **帧的比特率**：
 - (1) **CBR**：每一帧都是一样的。通过解析第一帧即可得知其他所有帧的比特率。。
 - (2) **VBR**：每一帧都不同，所以要针对每一帧具体解析帧头，才能得知每一帧的比特率具体是多少。

4.3.1. 常见 MPEG 帧的音频数据大小是 418 字节



最后，来说明，

以常见的**采样率为 44100 Hz**，**比特率为 128kbps 的 CBR 的 MP3** 来计算：

帧数据大小

$$= (\text{采样个数} * (1/\text{采样率})) * \text{帧的比特率} + \text{帧的填充大小}$$

$$= (1152 * (1/44100 \text{ Hz})) * 128\text{kbps} + \text{填充大小}$$

$$= 3343.7 \text{ 比特} + \text{填充大小}$$

$$= 417.959 \text{ 字节} + \text{填充大小}$$

$$= 418 \text{ 字节}$$

$$\begin{array}{r} 1152 \times 128000 \\ \hline 44100 \times 8 \end{array}$$

对应的 MPEG 帧大小为：

MPEG 帧大小

$$= \text{帧头} + \text{帧数据}$$

$$= 4 + 418$$

$$= 422 \text{ 字节}$$

而对于 VBR 的帧的大小，就不是能这么简单计算出来的了。

因为 VBR 是每一帧的比特率都是变化的，所以对于每一帧的大小，都先要解析每一帧的帧头，得到每一帧的比特率，然后才可以计算出来每个帧的大小。

另外提及一点，由于舍入误差，官方的计算帧大小的方法和此稍有不同。根据 ISO 标准，应该以 slot 为单位进行计算，然后对结果取整，再乘以 slot 的大小。

不过，我们此处计算的是 MPEG 的 Layer III，本身 slot 就是一个字节，所以计算方法是对的。

如果计算的是 Layer I，一个 slot 是 4 字节，就要先以 4 字节为单位进行计算，然后对结果取整，再乘以 slot 大小，即再乘以 4 字节。

5. MP3 的播放时间的计算公式及 XING 和 VBRI 头介绍

5.1. CBR 的 MP3 的播放时间(duration)计算方法

对于计算 CBR 的 MP3 的播放时间，其是 Constant Bitrate，固定的比特率，每一帧的比特率也都是固定的同样的大小，所以，相对来说，很容易想得到，用文件大小，直接除以比特率，就可以得到文件的播放时间了，即就用如下公式可以计算 MP3 的播放时间：

CBR Duration

= File Size(Byte) × 8 bit/Byte ÷ (Bitrate (K bit/s) × 1000 bit/Kbit)

CBR 播放时间

= 文件大小（字节）× 8 比特/字节 ÷（比特率 千比特/秒 × 1000 比特/千比特）

【公式 1】

其中：

➤ **文件大小**：严格地说，应该是 MP3 的文件总大小，减去 MP3 的 Tag 的大小，即
文件大小 = 总的 MP3 文件大小 - MP3 的 Tag 大小

其中，MP3 的 Tag，往往和 MP3 文件总大小相比，几乎可以忽略不计，所以，一般也可以直接用总的 MP3 为文件大小，直接来计算：

文件大小 = 总的 MP3 文件大小

➤ **比特率**：可以通过解析 MP3 文件的第一帧的 MPEG 的帧头，得到比特率的索引值，然后查比特率索引表，即可得到比特率是多少。

所以，可以看出，对于 CBR 的文件，可以用上面的公式，获得 MP3 文件大小后，再去解析第一帧的 MPEG 帧头，得到比特率索引值，查表得到比特率的值，然后就算出整个 CBR MP3 文件的播放时间。

而对于 VBR，由于每一帧的比特率都是变化的，所以计算起来就相对要复杂一些，要用另外的公式，下面就来详细介绍。

5.2. VBR 的 MP3 的播放时间(duration)计算方法

想要计算 VBR 的 MP3 的播放时间，总的来说，有两种方法：

5.2.1. 平均比特率法

这个方法，就是和 CBR 同样的思路，对于 VBR 的 MP3 来说，假如也像 CBR 的 MP3 一样，也有个类似的每一帧都是固定的某个值的比特率，那么计算整个 VBR 的播放时间，也就可以用上面 CBR 一样的公式去计算了。

由此，就有了平均比特率的概念，即，将所有帧的比特率的值相加，得到一个总的比特率的值，然后除以总的帧数，就得到了一个平均比特率，这样，使得理论上，此 VBR 相当于一个比特率为该平均比特率的 CBR 了。

不过，可以看出，需要计算平均比特率之前，要先得到每一帧的比特率的值，以及总的帧的数目，然后才可以计算平均比特率的值。

而为了得到每一帧的比特率的值，就要将整个 VBR MP3 文件都遍历一遍，以此找到所有的帧，并解析每一个帧的帧头，得到比特率索引值，然后查表得到比特率的值。

如此做的话，效率显然很低。因为此处只是为了计算整个 VBR MP3 的播放时间，却要遍历整个文件，还要解析每一帧的帧头，显得很得不偿失。

所以，就有了更好的，效率更高的，下面要介绍的另一种方法，来计算 VBR MP3 的播放时间。

另外，需要提醒的是，对于平均比特率来说，往往和第一帧的比特率相差很大。因为常见的 MP3 音乐的开头部分，即第一帧或者前几帧，多数是一些无声的数据，或者本身包含信息量很少，比特率很低的数据。因此，其意味着，如果解码器对于 VBR 文件，误解为 CBR 文件，按照 CBR 所有帧的比特率都相同的逻辑，去解析第一帧，得到一个比特率，然后用此比特率来计算整个文件的播放时间的话，那么往往计算出的播放时间和实际的相差很大。这也就是后面引用中一个帖子里面遇到的情况，即，Media Player Classic 播放 VBR 的 MP3 时的时间问题。

5.2.2. 总帧数法

总帧数法，即利用总的帧的数目，来计算 VBR 的播放时间。

此方法的前提，是我们前面就强调过的：

(1) **MP3，即 MPEG-1，Layer III，不论是 CBR，还是 VBR，每一帧的采样个数都是固定的 1152 个。即每一帧，都是固定的 1152 个采样。**

(2) **CBR 和 VBR 中的固定和可变，都是指的是比特率 Bitrate，而不是采样率 Sample Rate。对于同一 MP3 文件，不论 CBR 还是 VBR，采样率都是固定的。**

了解了这两个前提后，就可以看出，对于 VBR 来说，虽然每一帧的比特率不同，但是每一帧的时间都是固定的，因为

每一帧的时间

= 该帧的采样个数 * 该帧的采样率

= 1152 * 采样率

其中：

- **采样个数：MPEG-1，Layer III，即 MP3，不论是 CBR 还是 VBR，都是固定的 1152**
- **采样率：对于单个的 VBR 文件，都是统一的，固定的，常见的是 44100Hz。采样率可以通过解析第一帧的帧头得出采样率索引，然后查表得到采样率。**

既然知道了每一帧的时间都是固定的，那么很容易就想到，如果知道 VBR MP3 一共有多少帧，那么就可以用 总的帧数 × 每一帧的时间 = 总的时间长度了。

所以，剩下的事情，就是去得到 VBR MP3 的总的帧数。

最简单，但是效率很低的方法就是，像上面方法 1 一样，遍历整个 VBR 文件，找出一共有多少帧，对于第一帧，解析第一帧的帧头，得到采样率。

这样有了采样率和总的帧数，就可以用上面的解释的原理来计算了，对应公式就是：

VBR Duration

= Total_Frame_Number * Time_Per_Frame

= Total_Frame_Number * (Sample_Number * Time_Per_Sample)

= **Total_Frame_Number * (Sample_Number * (1 / Frame_Sample_Rate))**

VBR MP3 总的时长

= 总的帧数 * 单个帧的时长

= 总的帧数 * (帧的采样个数 * 每个帧的时长)

= **总的帧数 * (帧的采样个数 * (1 / 帧的采样率))** 【公式 2】

其中：

- **总的帧数**：VBR 中的总的帧的数目。
- **帧的采样个数**：对于 MP3 (MPEG1, Layer III) 来说，是固定的 1152 个采样。
- **帧的采样率**：通过解析第一帧，即可得知帧采样率索引，查表，即可得此采样率。

但是，可以看到，虽然此遍历整个文件以得到总的帧数的方法，但是还是显得效率不高。此处我们毕竟只是需要知道总的帧数而已，却还是要遍历文件。

对此问题，想象一下，要是有人在 VBR 的文件头部，单独提供了这个总的帧数，那么不就可以省去了我们再去遍历整个文件了吗？

而实际情况是，你所想到的事情，别人已经帮你实现了。^_^。

现实中，VBR 文件中，就是已经有了对应的头 Header，用于存放 VBR 相关的信息。

这样的头信息，也就是下面将要介绍的 XING 和 VBRI。

5.2.3. VBR 的两种 Header : XING 和 VBRI

VBR 的帧头，记录了和 VBR 相关的一些信息，至少包含了我们前面介绍的，用于方便我们计算 VBR 的播放时长的总的帧数。

VBR MP3 的帧头，主要有两种类型，XING 和 VBRI。

此外，VBR 的头中，往往还包含了一个用于定位的 TOC (table of content) 目录表。即用于在快进或快退的时候，通过表中的信息，可以方便地定位到对应的位置。如果没有此 TOC 表，需要单独去计算出对应的位置，比较麻烦。

关于它们的具体格式和含义，下面就对其进行详细解释。

5.2.4. Xing TAG / Xing 头 (header)

此 tag 由 XING 公司推出的算法/规范，所以叫做 XING。

对于大多数的 VBR 文件都加了此头，但并不全是。此头位于 MPEG 音频头后面的某个特定位置 (多数是 0x24)。包含了此 XING 头的第一个帧，其后的数据是空的，所以即使解码器没有考虑到此头，也可以正常处理此帧。对于 Layer III 的文件来说，比如常见的 MP3，此 VBR 放在边信息 (side information) 之后。

下表是 XING 头的具体格式及含义：

表 5 XING 头的格式及含义

位置 (字节)	长度 (字节)	含义	示例
0	4	4 个 ASCII 字符的 VBR 头 ID, 要么是 Xing, 要么是 Info, 无 NULL 结尾 (普通字符串都以 NULL, 即 \0 结尾)	'Xing'
4	4	存放一个标志, 用于表示接下来存在哪些域 / 字段, 各字段逻辑或的结果. 0x0001 - 存在总帧数 (Frames) 字段 0x0002 - 存在文件大小 (Bytes) 字段 0x0004 - 存在 TOC 字段 0x0008 - 存在音频质量指示字段	0x0007 就表示下面存在 总 帧数, 文件大小总字 节数, TOC 表
8	4	总帧数(Frames), 大端[可选]	7344
8 或 12	4	文件总大小, 单位字节, 大端[可选]	45000
8, 12, 16	100	TOC 表, 大端[可选]	
8 或 12, 16, 108, 112 , 116	4	音频质量指示, 最差 0, 最好 100, 大端[可选]	0

虽然知道了 XING 头的具体含义, 可以去根据具体的值, 解析出对应的含义了, 但是, 由于其是放在 side information 之后的, 所以, 要先定位, 找到 XING 头, 关于其位置, 用如下公式计算:

XING 头位置

= MPEG 头位置 + MPEG 帧头大小 + 边信息大小

= MPEG 头位置 + 4 字节 + 边信息大小 【公式 3】

其中:

- MPEG 头位置: 即通过程序去找到连续的 11 个 bit 都是 1 的位置, 即可同步 MPEG 的帧, 找到对应的 MPEG 头的开始处。
- 边信息大小: 详细信息, 后面用到此公式时会具体解释。

根据头的格式, Xing 头里面必须包含 ID 和 flag 这两个段。其他字段都是可选的, 是否包含, 要看 flag 的值。有时候这个 Xing 头, CBR 里面也有, 此时, 前面的 ID 的值就是 Info, 而不是 Xing 了。

5.2.5. VBRI 头 (header)

据了解, 目前此头信息, 只有用 Fraunhofer 的编码器生成的 MPEG 音频文件, 才会用到此头。其和 Xing 头不一样, 其放在第一个 MPEG 头的后面, 大小正好是 32 字节。其位置, 长度和示例, 都是以字节为单位。

下表是 VBRI 头的具体格式及含义，单位为字节：

表 6 VBRI 头的格式及含义

位置 (字节)	长度 (字节)	含义	示例
0	4	4 个 ASCII 字符的 VBR 头 ID: “VBRI”无 NULL 结尾	“VBRI”
4	2	版本 ID, 大端, 类型: DWORD	1
6	2	延迟, 类型: float	7344
8	2	音频质量指示	75
10	4	文件总大小, 大端, 类型: DWORD	45000
14	4	总的帧数, 大端, 类型: DWORD	7344
18	2	TOC 表的表项数目, 大端, 类型: WORD	100
20	2	TOC 表项的缩放因子, 大端, 类型: DWORD	1
22	2	单个 TOC 表项的大小, 单位字节, 最大为 4, 大端, 类型: DWORD	2
24	2	帧数/表项, 大端, 类型: WORD	845
26		用于检索的 TOC 表, 整型值, 可以通过每个表项大小乘以表项个数得到此 TOC 表的总大小, 大端	

6. 计算 CBR 和 VBR 的 MP3 文件的播放时间的步骤

此处只是大概总结一下，具体解析出播放时间，需要哪些步骤，可以参考相关源码：

(1) 可以去文章引用中的【1】，注册登录后，可以下载源代码，自己看，就知道了。

(2) 也可以去参考引用【2】中的 Mplayer 的 VBR 的 patch，里面写的更加简单，也更容易看明白。

下面就解释一下，关于如何去计算 MP3 的文件的播放时间，的具体的逻辑和顺序：

6.1. 定位到 MPEG 的帧头的位置

由于在计算 MP3 播放时间之前，要先找到对应的 MPEG 的帧头，所以，先要找到 MPEG 具体在某个位置。

具体方法是，如果文件开始没有 ID3 V2 的头信息，那么一般 MPEG 的帧头位置是 0，当然，具体还是要根据帧头中的同步位（sync bit），共 11 位去定位找到帧头。

6.2. 解析 MPEG 帧头，获取必要信息

主要是根据帧头格式，解析出 MPEG 的版本，MPEG 的 Layer，以及采样率，比特率等信息，用于后面的解码和计算播放时长。

6.3. 判断是 VBR 还是 CBR，根据公式计算播放时间

解析完 MPEG 头之后，按照如下步骤去判断是 CBR 还是 VBR，并计算出播放时长：

6.3.1. 定位出 Xing 头的位置

根据：

XING 头位置 = MPEG 头位置 + 4 + 边信息大小 【公式 3】

计算定位到 XING 头所在位置。上式中：

MPEG 头位置：

如果没有 ID3 V2 这类的信息的话，那么 MPEG 头位置就是文件的最开始，即 0 的位置。

MPEG 帧头大小：

固定的 32 bit = 4 字节，所以加 4。

边信息大小：

对应 Layer III，根据 MPEG 的版本，查下表可得，单位为字节：

表 7 MPEG Layer III 的边信息（side information）（单位：字节）

	MPEG 1	MPEG 2/2.5 (LSF)
立体声，联合立体声，双声道	32	17
单声道	17	9

注：

（1）其中红色部分，就是我们最常见的，双声道的 MP3，不论是 VBR 还是 CBR，对应的边信息，都是 32 个字节。

6.3.2.如果有 Xing 头，则是 VBR，解析 XING 头，计算播放时间

如果对应 XING 头的位置有对应的” Xing”字符，那就说明是 Xing 头。

那么就可以解析 Xing 头，找到对应的我们所需要的一些值，尤其是总的帧数。

然后用之前介绍的公式去计算 VBR 的播放时间：

VBR MP3 总的时长

= 总的帧数 * (帧的采样个数 * (1 / 帧的采样率)) **【公式 2】**

计算出播放时间长度。

其中：

➤ **总的帧数 (Number of Frames) :**

可以通过解析 XING 头，找出里面总帧数 (Frames)，这个字段，一般都是存在的。

➤ **帧的采样个数 (Samples Per Frame) :**

根据前面解析 MPEG，找到 MPEG 的版本，基于属于哪个 Layer，然后根据

“[表 4 MPEG 帧的采样数索引表 \(单位: 个/帧\)](#)”

查得每帧的采样个数，得到每一帧有多少个采样。

而对于 **MP3 (MPEG-1, Layer III)** 此处肯定是对应的 **1152**。

➤ **采样率 (Sampling Rate) :**

根据前面解析 MPEG，找到 2 比特大小的采样率索引，然后根据表：

“[表 3 MPEG 帧的采样率索引表 \(单位: Hz\)](#)”

“MPEG 不同版本对应的采样率”所对应的信息，找到对应的采样率。

此处，也就是找到我们前面所说的，总的帧数，加上另外两个参数：帧的采样数和帧的采样率（都是通过解析第一帧的帧头，即可算出对应的值），然后我们就可以算出 VBR 的 MP3 文件的总的播放时间长度了。

6.3.3.如果没 XING 头，定位出 VBRI 头位置，找 VBRI 头

计算出 VBRI 的位置，如果该位置找到“VBRI”字符，那么说明是 VBRI 头。

然后解析 VBRI，找到对应的总的帧数，然后和 XING 头算法类似，用上面的公式计算出 VBR 的播放时间即可。

6.3.4.如果也没找到 VBRI 头，则是 CBR，计算 CBR 的播放时间

如果连 VBRI 头也没找到，即，Xing 和 VBRI 都没有，那么则是 CBR。

然后用 [【公式 1】CBR 播放时间计算公式](#)，计算出播放时间长度：

CBR 播放时间

= 文件大小 (字节) × 8 比特/字节 ÷ (比特率 千比特/秒 × 1000 比特/千比特)
【公式 1】

其中：

➤ 文件大小 (File Size)

文件大小 = 总文件大小 - 头信息大小

这里的头信息，指的是 ID3 V1 或 ID3 V2 之类的头信息。

而总文件大小，这个不用多解释，就是整个 MP3 的大小。

两者单位都是字节。

➤ 比特率

通过解析第一帧的 MPEG，即可得到比特率的索引值，然后查表：

“[MPEG 比特率索引表](#)”

即可得到此 MP3 的比特率大小。

然后套用上面的公式，即可算出 CBR 的 MP3 的播放时间长度了。

【小提示】

另外，多说一句，我原先以为，还有另外一种计算方法，即：

“总帧数乘于每一帧的时长法”

总的时长 = 每一帧的时长 * 总的帧数

就是先计算每一帧的时间长度，再计算一共有多少帧，然后将两者相乘，即可得到文件总时长。

每一帧的时长

= 每一帧的采样个数 * 每一采样的时长

= 每一帧的采样个数 * (1 / 每一帧的采样频率)

总的帧数

= 总的文件大小 / 单个帧的大小

= 总的文件大小 (字节) * 8 比特/字节 /

((每个帧的时长 * 比特率 (千比特/秒) * 1000 比特/千比特))

但是后来发现，此法，其实和上面的是同一个方法，因为上面两个等式相乘之后，即为：

总的时长

= 每一帧的时长 * 总的帧数

= 每一帧的时长 * [总的文件大小 (字节) * 8 比特/字节 / ((每个帧的时长 * 比特率 (千比特/秒) * 1000 比特/千比特))]

= [总的文件大小 (字节) * 8 比特/字节 / [比特率 千比特/秒 * 1000 比特/千比特)]

还是和上面的方法是，是同一个公式。

7. MP3 相关的一些知识的解释

7.1. MP3 的文件的内容组织结构

总结起来，一般的 MP3 文件所包含的内容的结构如下：

表 8 MP3 文件的内容组织结构

[ID3。。。] ID3 V2 的头，大多数最新的 MP3，都有这个头 [APE 头] 用于 APE 格式的头，现在也用于 MPEG	
第一帧，格式如下：	
MPEG 音频头： 固定的 4 字节	
边信息： 9/17/32 字节	
[Xing 头]： 8-120 字节，如果是 VBR，多数都有此 Xing 头，而且只有第一帧有	
MPEG 音频数据	
第二帧，格式如下：	
MPEG 音频头： 固定的 4 字节	
边信息： 9/17/32 字节	
MPEG 音频数据	
第二帧，格式如下：	
MPEG 音频头： 固定的 4 字节	
边信息： 9/17/32 字节	
MPEG 音频数据	
。 。 。 。 。	
最后一帧，格式如下：	
MPEG 音频头： 固定的 4 字节	
边信息： 9/17/32 字节	
MPEG 音频数据	
[TAG 。。。] 128 字节的 ID3 V1 信息，如果没有前面的 ID3 V2，多数都有这个 ID3 V1 的头	

注：[]号内的，表示，可选，即如果有的话。

7.2. MP3 帧的时长是 26ms 的来历

时长，即时间长度。

MP3，即 MPEG-1，Layer III。

MPEG1，Layer III，根据下表：

MPEG 帧的采样数（单位：个/帧）

	MPEG 1	MPEG 2 (LSF)	MPEG 2.5 (LSF)
Layer I	384	384	384
Layer II	1152	1152	1152
Layer III	1152	576	576

可以查得，每一帧的采样个数，是 1152，这个值是固定的。

而 MPEG-1 所支持的采样率，根据下表：

MPEG 的采样率索引表（单位：Hz）

采样率索引	MPEG 1	MPEG 2 (LSF)	MPEG 2.5 (LSF)
00	44100 Hz	22050 Hz	11025 Hz
01	48000 Hz	24000 Hz	12000 Hz
10	32000 Hz	16000 Hz	8000 Hz
11	保留		

可以看出有三种，44.1K，48K 和 32K Hz。而我们实际常见的 CBR 或者 VBR 的 MP3 所采用的采样率，多数都是 44.1KHz=44100Hz。

所以，每一个采样的时长 = $1/44100$ 秒

因此：

每一帧的总时长

= 每一帧的采样个数 * 每个采样的时长

= 1152 个采样 * $1/44100$ 秒/采样

= $1152 / 44100$ 秒

= 0.026 秒

= 26 ms（毫秒）

这就是所谓的，MP3 的每帧的时长，都是 26ms 这一说法的缘由。

7.3. 怎么算出来 MP3 的压缩比大概是 1:11 的

正常，所谓 CD 里面的歌曲，是无损格式，即原始的声音，没有经过压缩的。

相对来说，音质最高，但是文件体积太大。到底有多大，我们可以来算一下：

每秒的原始 CD 的音频文件的数据量

= 声道数目 * 采样率 * 每个采样占用多少 bit

= 2 声道 * 44100Hz * 16 位

= 1411200 bit

而对应的 MP3 文件来说，即将原始的音频数据文件，经过 MP3 的压缩算法压缩后，数据量就小多了。以常见的双声道的，频率为 44100Hz，比特率为 128Kbps 的 MP3 为例：

每秒的 MP3 的数据量

= 比特率 * 1 秒

= 128K bits

= 128 000 bit

因此:

MP3 的压缩率

= 每秒的原始 CD 的音频文件的数据量 / 每秒的 MP3 的数据量

= 1411200 bit / 128 000 bit

= **11.025**

≈ **11**

这就是大家常说的，MP3 的压缩比大概有 1:11，即在保持不错的音质前提下，将数据量减小到了原来的 1/11。通俗点说，原先 CD 音质的，无压缩的歌曲是 11M 的话，那么对应的 MP3 就只有 1M。相对很不错的压缩比，大大减小了歌曲的体积，在互联网时代，此优势得到很好的体现，因此 MP3 才真正流行了起来。

当然，如果其他参数同上，而 MP3 比特率是 64Kbps，那么对应压缩比有 1:22；

同理，如果是 256Kbps 的 MP3，那么压缩比就小了，只有 1:5.5 左右了。

7.4. 问：每一个帧都包含帧头，会不会很浪费空间？

可能有人看到这里会有疑问，MP3 文件有很多数据帧所组成，而对于解码器去解码 MP3 文件，很多参数，只需要对于第一帧进行解码，就可以获得对应参数，用于解码了。也就意味着，MP3 的其他的帧中的 MPEG 帧头，就显得是多余的，没什么用。

问：为何却要每一帧都保存一个帧头，那不是很浪费空间吗？

答：理论上这么理解，是对的。但是实际上，还是要保留每一帧的 MPEG 帧的帧头，有两个原因：

7.4.1. 原因之一

实际上，如果清楚了 MP3 的音频帧的大小，你就会发现，这些帧头很小，对于数据帧本身和 MP3 文件大小的影响，可以忽略不计。

因为，MPEG 的帧头，一共就 32bit=4 字节，相对于每一帧的音频数据，以常见的采样率为 44100 Hz，比特率为 128kbps 的 CBR 的 MP3 来说，是 418 个字节，只相当于音频数据的 1/100 左右，所以，相对来说，对于整个文件的影响很小。

所以说，每个帧都加上这 4 个字节的帧头，对于数据帧本身和 MP3 文件大小的影响，可以忽略不计。

这只是每一帧都保留一个 MPEG 帧的帧头的原因之一。

7.4.2. 原因之二

虽然对于 CBR 的 MP3，除了第一帧之外的每一帧的 MPEG 帧头，是可以去掉。但是对于 VBR 的 MP3，每一帧的 MPEG 帧头，都包含了当前帧的比特率的信息，而 VBR 的每一帧的比特率的大小，都是不一定相同的，因此每一帧都必须要有对应的 MPEG 帧的帧头，来提供这些相关信息，供解码器正确解码和播放 MP3。

8. 后记

最后想说的是，其实很无语，对于很多东西，尤其是计算机相关的技术，网上能找到的中文资料，多数都是说的不是太清楚。其中，包括我这里要找的，如何去计算 VBR 的播放时间，多数情况是，不论是去百度 Google 一下，还是去 Google 百度一下，找了半天，还是没看到有几个人能说清楚。让人越发感叹道，国内的计算机方面的技术，和国外，差的的确不是一点两点。人家把规范都定好了，过了 N 年了，结果咱们到现在，也没几个搞清楚咋回事。。。因此，对于计算机方面的资料，尤其是涉及底层技术的，能详细解释清楚你的问题的，多数都是一些英文原版资料。所以，在此，再次，感谢一下原作者。。。

9. 文章引用

【1】 MPEG Audio Frame Header [登陆该页面后，有源码和程序供下载]

<http://www.codeproject.com/KB/audio-video/mpegaudioinfo.aspx>

【2】 MPlayer-1.0rc1 和 MPlayer-1.0rc2 的 VBR patch

http://hi.baidu.com/serial_story/blog/item/9180ba35f07d1d1891ef39e8.html

【3】 MPEG AUDIO FRAME HEADER (mp3 format)

<http://www.datavoyage.com/mpgscript/mpeghdr.htm>

【4】 .MPEG Audio Layer I/II/III frame header

http://www.mp3-tech.org/programmer/frame_header.html

【5】 MP3 Profi Info

http://www.goat.cz/index.php?path=MP3_MP3ProfiInfo

【6】 MP3 Tech

<http://www.mp3-tech.org/>

【7】 Media Player Classic 播放 VBR 的 MP3 时的时间问题

<http://xialulee.spaces.live.com/blog/cns!4ee324c8acfa82db!230.entry?wa=wsignin1.0&sa=835890451>

【8】 什么是 MP3

<http://wenwen.soso.com/z/q140151046.htm>

【9】 Variable bitrate

http://en.wikipedia.org/wiki/Variable_bitrate

【10】 MPEG Audio Compression Basics

<http://www.datavoyage.com/mpgscript/mpeghdr.htm>

【11】 ID3 tag version 2

<http://id3.org/id3v2-00>

【12】 MP3

<http://en.wikipedia.org/wiki/MP3>

【13】 Mp3tag

<http://zh.wikipedia.org/zh/Mp3tag>

【14】 CBR, VBR and bit reservoir

<http://www.hydrogenaudio.org/forums/lofiversion/index.php/t50345.html>

【15】 Difference Between ABR and VBR

<http://www.differencebetween.net/technology/difference-between-abr-and-vbr/>