**git格式文件结构**

|  |  |  |
| --- | --- | --- |
| GIF署名 | | 文件头 |
| 版本号 | |
| 逻辑屏幕标识符 | | GIF数据流 |
| 全局颜色列表 | |
| … | |
| 图象标识符 | 图象块 |
| 图象局部颜色列表图 |
| 基于颜色列表的图象数据 |
| … | |
| GIF结尾 | | 文件结尾 |

GIF格式的文件结构整体上分为三部分：文件头、GIF数据流、文件结尾。其中，GIF数据流分为全局配置和图像块。接下来我们将逐一分析GIF格式各部分的作用，并结合Glide的代码，学习如何解析。

**GIF署名(Signature)和版本号(Version)**

GIF的前6个字节内容是GIF的署名和版本号。我们可以通过前3个字节判断文件是否为GIF格式，后3个字节判断GIF格式的版本。

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BYTE | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | BIT |
| 1 | ‘G’ | | | | | | | | GIF文件标识 |
| 2 | ‘I’ | | | | | | | |
| 3 | ‘F’ | | | | | | | |
| 4 | ‘8’ | | | | | | | | GIF文件版本号:  87a - 1987年5月  89a - 1989年7月 |
| 5 | ‘7’或’9’ | | | | | | | |
| 6 | ‘a’ | | | | | | | |

**逻辑屏幕标识符(Logical Screen Descripotr)**

逻辑屏幕标识符配置了GIF一些全局属性，我们通过读取解析它，获取GIF全局的一些配置

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BYTE | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | BIT |
| 1 | 逻辑屏幕宽度 | | | | | | | | 像素数，定义GIF图象的宽度 |
| 2 |
| 3 | 逻辑屏幕高度 | | | | | | | | 像素数，定义GIF图象的高度 |
| 4 |
| 5 | m | cr | | | s | pixel | | | 具体描述见下… |
| 6 | 背景色 | | | | | | | | 背景颜色（在全局颜色列表中的索引，如果没有全局颜色列表，该值没有意义） |
| 7 | 像素宽高比 | | | | | | | | 像素宽高比(Pixel Aspect Radio) |

屏幕逻辑宽度：定义了GIF图像的像素宽度，大小为2字节;

屏幕逻辑高度：定义了GIF图像的像素高度，大小为2字节;

m - 全局颜色列表标志(Global Color Table Flag)，当置位时表示有全局颜色列表，pixel值有意义;

cr - 颜色深度(Color ResoluTion)，cr+1确定图象的颜色深度;

s - 分类标志(Sort Flag)，如果置位表示全局颜色列表分类排列;

pixel - 全局颜色列表大小，pixel+1确定颜色列表的索引数（2^(pixel+1)）;

背景颜色：背景颜色在全局颜色列表中的索引（PS:是索引而不是RGB值，所以如果没有全局颜色列表时，该值没有意义）;

像素宽高比：全局像素的宽度与高度的比值;

全局颜色列表(Global Color Table)

文件终结

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BYTE | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
| 1 | 文件终结 | | | | | | | | GIF Trailer – 标识GIF文件结束，固定值0x3B |