**字体**

1.字体类型分类？

字体类型有如下表：

|  |  |
| --- | --- |
| 类型 | SubType赋值（描述） |
| Type0 | Type0(一个复合字体从派生的CIDFont字形组合) |
| Type | Type1（使用1型字体定义字体的类型） |
| MMType1（一种多主字体，1型字体的扩展，允许从单个字体生成多种字体样式） |
| Type3 | Type3（使用 PDF 图形操作符的字符串来定义字形的字体） |
| TrueType | TrueType基于 TrueType 字体格式的字体 |
| CIDFont | CIDFontType0（一个 CIDFont，字形基于 1 型 字 体 技 术） |
| CIDDontType2（一个 CIDFont，字形基于 TrueType 型字体技术） |

1. 编码

除了Type3的字体，每个字体的程序有一个内置的编码。在某些情况下，一个

PDF 字体字典可以更改字体的内置编码来匹配生成的文本应用程序。编码字符的

灵活性有以下两个原因：

• 它允许根据现有的各种约定来显示文本编码。例如，微软 Windows 和苹果Mac 操作系统使用不同编码标准的拉丁文字，而且许多应用程序使用自己的专用编码。

• 它允许应用程序指定字符怎样从大的字符集里选中进行编码。一些字符集包括超过 256 个字符，包括连字，重音符号，以及高质量排版或非拉丁书写系统所需的其他符号。在相同的字符设置中不同的编码可以选择不同的子集。

3.简单字体与复杂字体

**简单字体：**

有以下的特征：

 1. 字体的字形由一个从某字符串获得的单字节字符代码确定，该字符串由文本显示运算符给出。可以推出，这些代码索引形成 256 个字形的表格；从代码到字形的映射就称为字体的译码过程。

 2.每个字形有一个单一的指标，包括水平位移或设置，宽度。也就是说，简单的字体只支持横向书写模式。

 3.除了Type0字体，在非标记的 PDF 文件的Type3字体，以及某些标准 Type1字体，每个字体列表中包含一个子列表，字体描述符，包含字体宽度指标以及字体的其他属性。在那些属性中的是一个可选的字体文件流，包含着字体构成程序。

**复杂字体：**

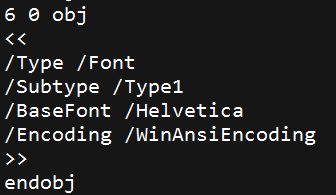
一种复合字体也称为Type0的字体,是一个叫 CIDFont 的一个字体对象中得到字形的字体。一种复合字体由字体字典的子类型值是Type0表示。Type0字体被称为根字体，和其关联的 CIDFont 被称为及其子体。

**区别：**

当前字体是复合时，文本显示运算符与简单的字体是不同的。对于简单字体而言，显示的字符串的每个字节选择一种字形，而为复合字体时，可以解码的一个或多个字节序列从子代 CIDFont 选择一种字形。这项设施支持大量的字符集的使用，如中国、日语以及韩语。复合字体还简化了有扩音器的编码要求的字体的系统。

4.pdf中hello world 如何显示出来？

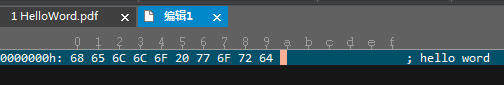
如下图所示：



首先在这里，我们可以知道，该文档使用的是TYPE1的字体规范，该规范表明了一个字节对应与一个字符编码。其次编码规范为WinAnsiEncoding，这是在附录D中的默认支持的编码格式，默认去拉丁字符集当中映射成8进制的编码。

根据后面附录中的拉丁字符集，将hello world 这几个字符按照单个字母对应编码成相应的8进制，然后根据PDFDocEncoding Character Set这个表中，找到  
默认的从字符到Unicode编码的映射关系。然后再根据cmap表中的(3.1)表取到相应的字形，根据上下文环境中的资源字典中的BaseFont的值来唯一确定使用的绘制字体的字体样式是helvetica，最后绘制到pdf中。

5.为什么pdf中的文档能够复制到其他编译器进行显示？



我们可以看到，将hello word这个单词进行复制的时候，实际上将数据先进行一步编码，转成通用的Unicode编码，然后将Unicode编码复制到相应的txt文件中，用户在文本编辑器中看到的文字样式其实是将复制过去的Unicode编码根据该文本编辑器的默认编码重新进行了一次转码解码生成的文字。

TrueType字体

TrueType 字体字典与Type1字体词典，有以下区别：

1. Subtype 项的值为 TrueType。

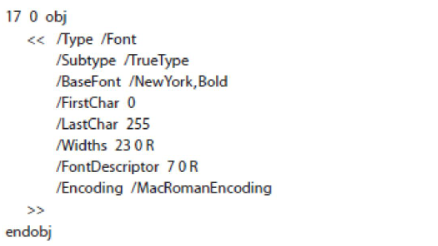
2. Base Font 的值根据以下两种方式确定：

 a.如果TrueType字体表中”name”表中含有描述项，则使用该描述项。

b.若在“name”表没有该描述项的情况下，从字体在主机作业系统中的名称来衍生补充名称。在 Windows 系统上，该名称基于 LOGFONT 结构中的 lfFaceName 区域；在 Mac 操作系统中，它基于 FOND 资源的名称。如果名称包含任何空格，空格将被删除。

3.encoding 中不推荐使用Difference这个属性。

结合具体的实例说明TrueType字体如何绘制出来？



1.首先根据Tj从当前上下文环境中获得文本的Tf所指向的字体资源引用找到对应的字体字典中的SubType的值为TrueType，从而确定了是一个字符对应一个字节。

2.然后根据当前上下文环境中的字体字典中的Encoding的值MacRomanEncoding，确定了是符合pdf规范的字体，可以从附录D中找到对应的name值。

3.得到name的值之后，根据Adobe官方给出的Glyph List将第二步找到的字形名称，也就是name的值映射成为相应的Unicode编码。

4.最后根据TrueType的映射规则，从cmap表中找到从Unicode值映射到字形的的映射关系(这里采用的是默认的(3.1)表，即默认是在windows环境中)

5.以上的步骤完成了从字节流到字形的映射，接下来通过上下文环境中的字体资源字典中的BaseFont的值来绘制相应的字形到pdf中。

6.如果在源文档的字体使用粗体或斜体样式的字体，但没有该风格字体的数据，主机操作系统将自动合成该风格。

样例解析：HelloWord.pdf