**读懂diff**

作者： [阮一峰](http://www.ruanyifeng.com)

日期： [2012年8月29日](http://www.ruanyifeng.com/blog/2012/08/)

diff是Unix系统的一个很重要的工具程序。

它用来比较两个文本文件的差异，是代码版本管理的基石之一。你在命令行下，输入：

　　$ diff <变动前的文件> <变动后的文件>

diff就会告诉你，这两个文件有何差异。它的显示结果不太好懂，下面我就来说明，如何读懂diff。

**一、diff的三种格式**

由于历史原因，diff有三种格式：

　　\* 正常格式（normal diff）

　　\* 上下文格式（context diff）

　　\* 合并格式（unified diff）

我们依次来看。

**二、示例文件**

为了便于讲解，先新建两个示例文件。

第一个文件叫做f1，内容是每行一个a，一共7行。

　　a  
　　a  
　　a  
　　a  
　　a  
　　a  
　　a

第二个文件叫做f2，修改f1而成，第4行变成b，其他不变。

　　a  
　　a  
　　a  
　　**b**  
　　a  
　　a  
　　a

**三、正常格式的diff**

现在对f1和f2进行比较：

　　$ diff f1 f2

这时，diff就会显示正常格式的结果：

　　4c4  
　　< a  
　　---  
　　> b

第一行是一个提示，用来说明变动位置。

　　4c4

它分成三个部分：前面的"4"，表示f1的第4行有变化；中间的"c"表示变动的模式是内容改变（change），其他模式还有"增加"（a，代表addition）和"删除"（d，代表deletion）；后面的"4"，表示变动后变成f2的第4行。

第二行分成两个部分。

　　< a

前面的小于号，表示要从f1当中去除该行（也就是第4行），后面的"a"表示该行的内容。

第三行用来分割f1和f2。

　　---

第四行，类似于第二行。

　　> b

前面的大于号表示f2增加了该行，后面的"b"表示该行的内容。

最早的[Unix](http://www.ruanyifeng.com/blog/2010/03/unix_copyright_history.html)（即AT&T版本的Unix），使用的就是这种格式的diff。

**四、上下文格式的diff**

上个世纪80年代初，加州大学伯克利分校推出BSD版本的Unix时，觉得diff的显示结果太简单，最好加入上下文，便于了解发生的变动。因此，推出了上下文格式的diff。

它的使用方法是加入c参数（代表context）。

　　$ diff -c f1 f2

显示结果如下：

　　\*\*\* f1 2012-08-29 16:45:41.000000000 +0800  
　　--- f2 2012-08-29 16:45:51.000000000 +0800  
　　\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
　　\*\*\* 1,7 \*\*\*\*  
　　 a  
　　 a  
　　 a  
　　!a  
　　 a  
　　 a  
　　 a  
　　--- 1,7 ----  
　　 a  
　　 a  
　　 a  
　　!b  
　　 a  
　　 a  
　　 a

这个结果分成四个部分。

第一部分的两行，显示两个文件的基本情况：文件名和时间信息。

　　\*\*\* f1 2012-08-29 16:45:41.000000000 +0800  
　　--- f2 2012-08-29 16:45:51.000000000 +0800

"\*\*\*"表示变动前的文件，"---"表示变动后的文件。

第二部分是15个星号，将文件的基本情况与变动内容分割开。

　　\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

第三部分显示变动前的文件，即f1。

　　\*\*\* 1,7 \*\*\*\*  
　　 a  
　　 a  
　　 a  
　　!a  
　　 a  
　　 a  
　　 a

这时不仅显示发生变化的第4行，还显示第4行的前面三行和后面三行，因此一共显示7行。所以，前面的"\*\*\* 1,7 \*\*\*\*"就表示，从第1行开始连续7行。

另外，文件内容的每一行最前面，还有一个标记位。如果为空，表示该行无变化；如果是感叹号（!），表示该行有改动；如果是减号（-），表示该行被删除；如果是加号（+），表示该行为新增。

第四部分显示变动后的文件，即f2。

　　--- 1,7 ----  
　　 a  
　　 a  
　　 a  
　　!b  
　　 a  
　　 a  
　　 a

除了变动行（第4行）以外，也是上下文各显示三行，总共显示7行。

**五、合并格式的diff**

如果两个文件相似度很高，那么上下文格式的diff，将显示大量重复的内容，很浪费空间。1990年，GNU diff率先推出了"合并格式"的diff，将f1和f2的上下文合并在一起显示。

它的使用方法是加入u参数（代表unified）。

　　$ diff -u f1 f2

显示结果如下：

　　--- f1 2012-08-29 16:45:41.000000000 +0800  
　　+++ f2 2012-08-29 16:45:51.000000000 +0800  
　　@@ -1,7 +1,7 @@  
　　 a  
　　 a  
　　 a  
　　-a  
　　+b  
　　 a  
　　 a  
　　 a

它的第一部分，也是文件的基本信息。

　　--- f1 2012-08-29 16:45:41.000000000 +0800  
　　+++ f2 2012-08-29 16:45:51.000000000 +0800

"---"表示变动前的文件，"+++"表示变动后的文件。

第二部分，变动的位置用两个@作为起首和结束。

　　@@ -1,7 +1,7 @@

前面的"-1,7"分成三个部分：减号表示第一个文件（即f1），"1"表示第1行，"7"表示连续7行。合在一起，就表示下面是第一个文件从第1行开始的连续7行。同样的，"+1,7"表示变动后，成为第二个文件从第1行开始的连续7行。

第三部分是变动的具体内容。

　　 a  
　　 a  
　　 a  
　　-a  
　　+b  
　　 a  
　　 a  
　　 a

除了有变动的那些行以外，也是上下文各显示3行。它将两个文件的上下文，合并显示在一起，所以叫做"合并格式"。每一行最前面的标志位，空表示无变动，减号表示第一个文件删除的行，加号表示第二个文件新增的行。

**六、git格式的diff**

版本管理系统git，使用的是合并格式diff的变体。

　　$ git diff

显示结果如下：

　　diff --git a/f1 b/f1  
　　index 6f8a38c..449b072 100644  
　　--- a/f1  
　　+++ b/f1  
　　@@ -1,7 +1,7 @@  
　　 a  
　　 a  
　　 a  
　　-a  
　　+b  
　　 a  
　　 a  
　　 a

第一行表示结果为git格式的diff。

　　diff --git a/f1 b/f1

进行比较的是，a版本的f1（即变动前）和b版本的f1（即变动后）。

第二行表示两个版本的git哈希值（index区域的6f8a38c对象，与工作目录区域的449b072对象进行比较），最后的六位数字是对象的模式（普通文件，644权限）。

　　index 6f8a38c..449b072 100644

第三行表示进行比较的两个文件。

　　--- a/f1  
　　+++ b/f1

"---"表示变动前的版本，"+++"表示变动后的版本。

后面的行都与官方的合并格式diff相同。

　　@@ -1,7 +1,7 @@  
　　 a  
　　 a  
　　 a  
　　-a  
　　+b  
　　 a  
　　 a  
　　 a

**七、阅读材料**

　　\* [diff - Wikipedia](http://en.wikipedia.org/wiki/Diff)

　　\* [How to read a patch or diff](http://www.markusbe.com/2009/12/how-to-read-a-patch-or-diff-and-understand-its-structure-to-apply-it-manually/)

　　\* [How to work with diff representation in git](http://stackoverflow.com/questions/2529441/how-to-work-with-diff-representation-in-git)