Bash源码分析(2)

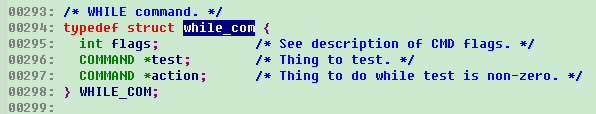
周荣华

作者简介：10年通讯底层研发经验，熟悉linux/vxworks等实时操作系统的内核原理和实现，在虚拟化的openstack，kubernetes，docker等领域也初有涉猎。

摘要：前文主要讲述了通用数据结构和for、case语句的定义，本文继续讲解WHILE、IF、CONNECTION和SIMPLE\_COM的讲解。

# 3 bash使用到的主要数据结构介绍

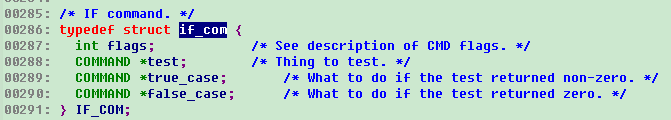
## 3.4 WHILE\_COM



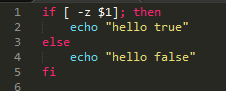
WHILE\_COM比较简单，主体有两部分组成，判断条件和执行体。上篇文章调试过程中导致CPU冲高到100%的例子就是用的WHILE\_COM，不过例子中的WHILE\_COM的判断条件留空，相当于永远为true。



## 3.5 IF\_COM

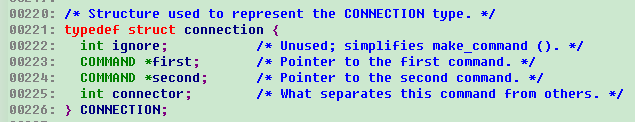


IF\_COM由3段组成，条件判断，条件为true时的执行体和条件为false时的执行体，其中false情况下的执行体可以为空。



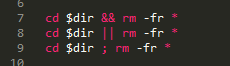
从IF\_COM的定义看，3部分都可以是复杂的COMMAND结构，所以嵌套起来也可以做的非常复杂，例如可以在test部分通过执行脚本，依靠脚本的返回值来判断是应该执行true\_case还是false\_case。

## 3.6 CONNECTION



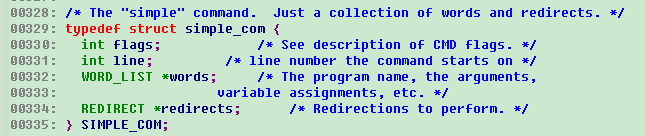
CONNECTION由4个属性组成：ignore字段对应其他命令结构种的flags，但对连接命令实际上没有用first对应第一条命令，second对应第二条命令，connector对应两条命令之间的连接符。难道只能两个命令一起用，不能多余两个命令一起调用？显然不是，一个CONNECTION对应一个连接符连接起来的两段命令，每段命令又可以是一个CONNECTION，这样就形成了级联的效果。

CONNECTION有3种：AND\_AND对应“&&”，表示first执行返回结果为0的时候执行second；OR\_OR对应“||”，表示first执行返回结果为非0的时候执行second；分号对应的connector还是分号，表示无论first执行结果是0还是非0，都执行second。是不是有点像C语言里面的&&，||和;？



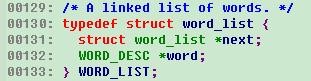
上面的三个例子别真的执行，后果很严重(☺)。第一条表示删除$dir对应值的目录中的所有文件；第二条表示$dir不存在的时候删除当前目录下面的所有文件；第三条表示，如果$dir存在就删除$dir目录下的所有文件，如果不存在就删除当前目录下面的所有文件。

## 3.7 SIMPLE\_COM

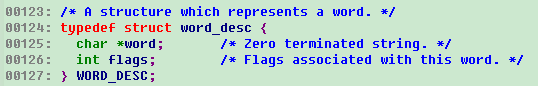


SIMPLE\_COM按字面意思就是简单命令，结构体由四部分组成，通用的flags，行号line，命令队列WORD\_LIST，重定向队列REDIRECT。

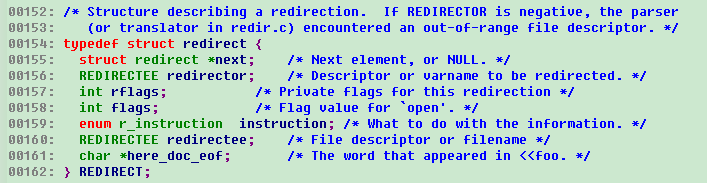
WORD\_LIST队列比较容易理解。一堆命令的集合：



其中单个命令还可以独立设置flags：



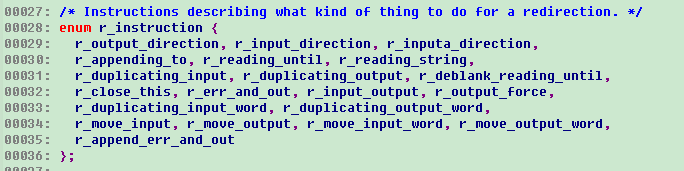
REDIRECT就是重定向的意思，这里也有很多种重定向，结构体的各个属性的含义：next，组成重定向链表的指针；redirector，重定向的源；rflags，重定向时使用的私有flags；flags，打开重定向目标文件时的flags；instruction，重定向的实际功能指令，这个又有很多种，下面会详细描述；redirectee，重定向的目的文件描述符或者文件名；here\_doc\_eof，本地文件。



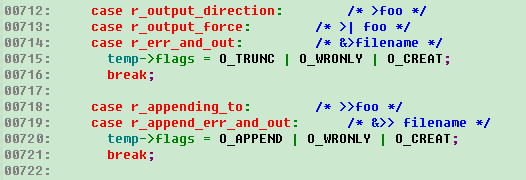
从REDIRECTEE的定义看，它既可以是一个文件描述符，例如0表示标准输入，1表示标准输出，2表示错误输出，也可以是一个文件名。



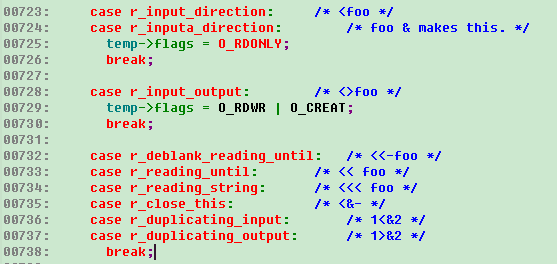
Bash支持二十种不同的重定向，后面会根据bash的源代码来一一解释一下具体内容（bash源代码的注释对重定向的含义理解也有很多帮助）：



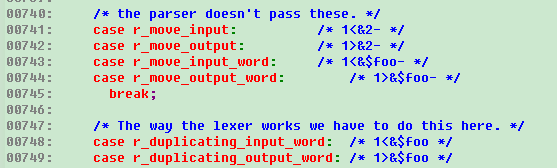
先来五种输出的重定向：普通输出，强制输出，错误和标准输出，叠加输出，错误和标准叠加输出。统一说一下几个概念，标准输出就是2级stdout，错误输出就是3级sdterr，强制的意思是文件存在的情况下会被先清空，再增加，叠加输出的意思是原有内容后面再增加。



再来九种输入和输出重定向：普通输入重定向，后台执行，输入和输出同时重定向，去掉空格的输入重定向，输入重定向，字符串作为输入，关闭重定向源（怎么还有这种应用场景？），复制输入，复制输出。



紧接着六种输入输出的重定向分别为输入剪切，输出剪切，字符指向的输入剪切和字符指向的输出剪切，字符指向的输入复制和字符指向的输出复制。



Bash的重定向真是博大精深！！明天继续其他COMMAND的定义讲解。