

批量处理文件 蒋砚军 北京邮电大学计算机学院

▶ 问题



find src -name *.c -exec grep -n -- --help {} /dev/null \;

借助find的"壳"功能去遍历目录,对遍历到的每个符合条件的文件执行grep命令。

缺点:效率低,因为每个命中的对象都需要执行grep命令:创建一个进程,完成任务后进程消亡,然后再创建,再消亡,.....。

grep命令也提供了-r选项,可以递归地搜索子目录下的文件,例如:

grep -nr -- --help *.c 这样无法检索,因为那些子目录名字不能被*.c匹配 grep -nr -- --help * 这样可以检索,但是却检索了太多非C源程序的不感兴趣的文件,输出结果被无用的信息淹没。应该使用find精确筛选的功能。

2

▶ 利用find与xargs的组合



find src -name *.c -exec grep -n -- --help {} /dev/null \;

上述命令存在的问题。

如果能把

find src -name *.c -print

生成的文件名列表追加在下列命令后面就可以了

grep -n -- --help filelist

命令xargs可以用来完成这个工作:

find src -name *.c -print | xargs grep -n -- --help

xargs命令把标准输入追加到它的参数列表后面,也就是上述grep...的后面,再作为一个命令来执行。这样利用find精确筛选,利用grep批量处理文件,提高效率。

3

xargs:将标准输入组织成命令执行



- ◆将标准输入构造为命令的命令行参数
- ◆如果标准输入的数据量很大,xargs指定的处理程序会启动多个进程运行,每个进程处理一批,会是几千个参数(命令行参数占满128K字节)
- ◆可以使用xargs的-n选项指定每批处理多少个
- ◆xargs经常与find配合使用,也可以与其它命令组合使用



▶ xargs:构造参数列表并运行命令



■解决shell文件名生成时,因为文件太多,缓冲区空间受限而文件名 展开失败的问题

```
rm -f *.dat 文件名*.dat展开失败,可以使用下面的命令
ls | grep ".dat$" | xargs rm -f
```

find命中目录名因删除目录导致目录遍历过程遇到麻烦

```
find . -name CVS -exec rm -rf {} \;
```

改为:

find . -name CVS -print | xargs rm -rf





