

第七章

枯燥 看不懂

第八章

这是因为Linux 支援的压缩指令非常多，且不同的指令所用的压缩技术并不相同，当然彼此之间可能就无法互通压缩/解压缩档案啰

```
*.Z compress 程式压缩的档案；
*.zip zip 程式压缩的档案；
*.gz gzip 程式压缩的档案；
*.bz2 bzip2 程式压缩的档案；
*.xz xz 程式压缩的档案；
*.tar tar 程式打包的资料，并没有压缩过；
*.tar.gz tar 程式打包的档案，其中并且经过gzip 的压缩
*.tar.bz2 tar 程式打包的档案，其中并且经过bzip2 的压缩
*.tar.xz tar 程式打包的档案，其中并且经过xz 的压缩
```

tar: -j *.tar.bz2 -Z *.tar.gz -J *.tar.xz

压缩：tar -j c v -f filename.tar.bz2 要被压缩的档案或目录名称

查询：tar -j t v -f filename.tar.bz2

解压缩：tar -j x v -f filename.tar.bz2 -C 欲解压缩的目录

```
[dmtsai@study ~]$ tar [-z|-j|-J] [cv] [-f待建立的新档名] filename... <==打包与压缩
[dmtsai@study ~]$ tar [-z|-j|-J] [tv] [-f既有的tar档名] <==察看档名
[dmtsai@study ~]$ tar [-z|-j|-J] [xv] [-f既有的tar档名] [-C目录] <==解压缩
```

选项与参数：

- c：建立打包档案，可搭配 -v 来察看过程中被打包的档名(filename)
- t：察看打包档案的内容含有哪些档名，重点在察看『档名』就是了；
- x：解打包或解压缩的功能，可以搭配 -C (大写) 在特定目录解开
特别留意的是，-c, -t, -x 不可同时出现在一串指令列中。
- z：透过gzip 的支援进行压缩/解压缩：此时档名最好为*.tar.gz
- j：透过bzip2 的支援进行压缩/解压缩：此时档名最好为*.tar.bz2
- J：透过xz 的支援进行压缩/解压缩：此时档名最好为*.tar.xz
特别留意，-z, -j, -J 不可以同时出现在一串指令列中
- v：在压缩/解压缩的过程中，将正在处理的档名显示出来！
- f filename：-f 后面要立刻接要被处理的档名！建议 -f 单独写一个选项啰！（比较不会忘记）
- C 目录：这个选项用在解压缩，若要在特定目录解压缩，可以使用这个选项。

其他后续练习会使用到的选项介绍：

- p(小写)：保留备份资料的原本权限与属性，常用于备份(-c)重要的设定档
- P(大写)：保留绝对路径，亦即允许备份资料中含有根目录存在之意；
- exclude=FILE：在压缩的过程中，不要将FILE 打包！

第九章

9.3.6 vim 常用指令示意图

为了方便大家查询在不同的模式下可以使用的vim 指令，鸟哥查询了一些vim 与Linux 教育训练手册，发现底下这张图非常值得大家参考！可以更快速有效的查询到需要的功能喔！看看吧！

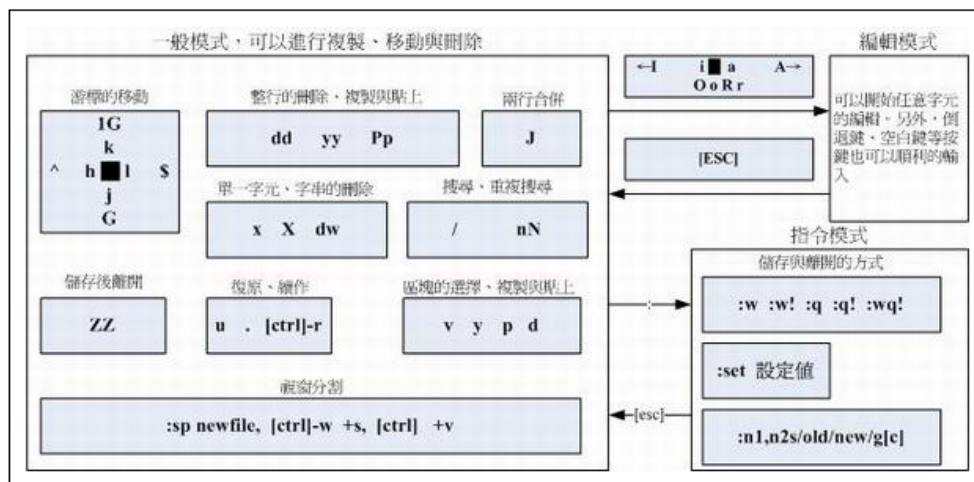


图9.3.10、vim 常用指令示意图