

文本编辑器Vi的使用

文本编辑器是所有计算机系统中最常使用的一种工具。在所有 UNIX 和 Linux 操作系统中, Vi 编辑器是最基本的文本编辑器, 无论是一般的文本文件、数据文件, 还是编写的源程序文件, 都能使用 Vi 来建立、编辑、显示和处理。Vi 是“Visual Interface”的简称, 它在 Linux 上的地位就仿佛 DOS 下的 EDIT、Windows 下的记事本, 但它可以执行输出、删除、查找、替换、块操作等众多文本操作, 而且用户可以根据自己的需要对其进行定制, 这是其他编辑程序所没有的。尽管现代的 Linux 发行版也都提供了最新的基于图形环境下的其他文本编辑器, 但 Vi 开销较小, 功能却较强, 以及兼容各种 Linux 系统, 使得它历经数十年仍然被广泛使用。Vi 虽然使用起来稍显复杂, 可一旦掌握了它, 就可在整个 Linux 的世界里畅行无阻。

3.1 执行与结束Vi

要进入 Vi, 可直接在系统提示符下键入 Vi 并加上文件名来打开一个现存的文件或是新创建一个文件, 也可以直接输入 Vi 来编辑一个新文件。若直接输入 Vi, 则系统会打开如图 1-63 所示的界面, 此方式下, 只需在退出命令后输入文件名即创建了新的文件。

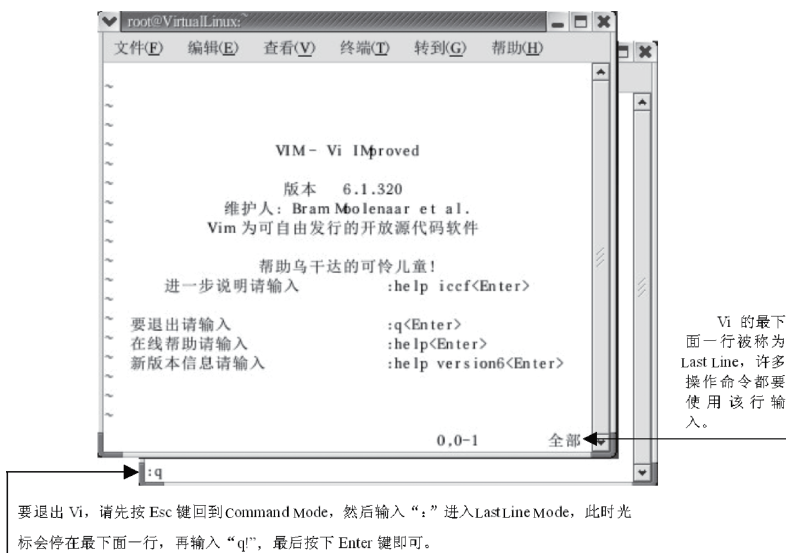


图1-63 在GNOME的终端窗口中打开的Vi文本编辑器



在 Red Hat Linux 9 中, Vi 已被发展成为 VIM (VI Improved), 即 Vi 的加强版。若在系统提示符下键入 vim 也会出现同样的界面。Vi 的界面分为两个部分: 编辑区和命令区。命令区在屏幕的最下方一行, 可以在此处输入命令; 而除了命令区之外的范围都属于编辑区, 此区也就是实际进行文件编辑或程序编写的地方。

3.2 Vi的三种模式及相互切换

1. Vi的三种工作模式

(1) 编辑模式 (Command Mode): 这是进入 Vi 时的默认模式, 主要功能是控制光标的移动、删除字符、区段复制。

(2) 插入模式 (Insert Mode): 唯一的功能是文字数据的输入, 按 Esc 键可以回到编辑模式 (Command Mode)。

(3) 命令模式 (Last Line Mode): 保存文件、退出 Vi, 以及其他的设置, 例如, 查找或取代字符串等。

2. Vi三种工作模式之间的相互切换

(1) 编辑模式 (Command Mode) → 插入模式 (Insert Mode)。

- 按 a 键: 从当前光标所在位置的下一个字符开始输入。
- 按 i 键: 从光标所在位置插入新输入的字符。
- 按 o 键: 新增加一行, 并将光标移动到下一行的开头。

注意: 在 Command Mode 下输入命令时, 如 a、i、o 等字符并不会显示出来。

(2) 插入模式 (Insert Mode) → 编辑模式 (Command Mode): 只需按下 Esc 键。

(3) 编辑模式 (Command Mode) → 命令模式 (Last Line Mode): 输入 “:” 键即可。

(4) 命令模式 (Last Line Mode) → 编辑模式 (Command Mode): 输入命令后回车即可。

不同模式的切换及主要功能如图 1-64 所示。

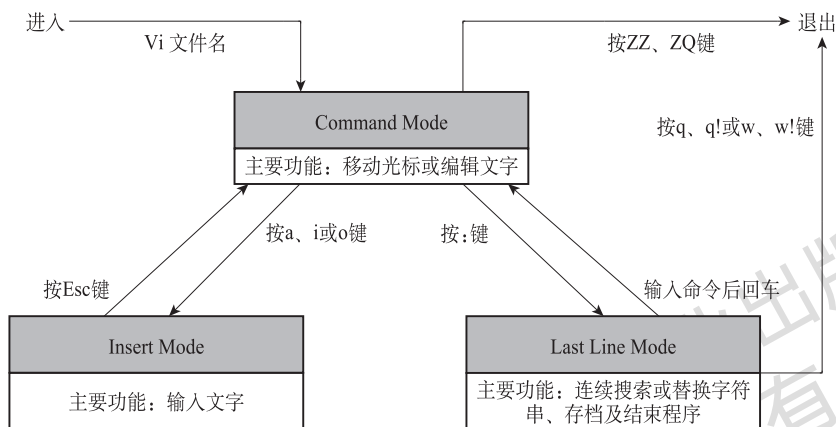


图1-64 Vi三种模式之间的转换示意图



3.3 编辑模式下的操作

编辑模式（Command Mode）中可用的功能键有很多，这里列出常用的编辑命令。

1. 插入文本（见表1-1）

表1-1 插入文本

命令	说明
i	在光标前插入文本
a	在光标后插入文本
I	在当前行的前端插入文本
A	在当前行的末端插入文本
O	在当前行前插入一行
o	在当前行后插入一行

2. 移动光标（见表1-2）

表1-2 移动光标位置

命令	说明
h、j、k、l	分别用于光标左移、下移、上移、右移一个字符
Ctrl + b	向后滚动一个屏幕（即 PgUp）
Ctrl + f	向前滚动一个屏幕（即 PgDn）
Ctrl + u	向后滚动半个屏幕
Ctrl + d	向前滚动半个屏幕
Ctrl + e	向后滚动一行
Ctrl + y	向前滚动一行
0（数字）	光标移至该行的行首
\$	光标移至该行的行尾
G	光标移至文本的最后一行的行首
W 或 w	光标移至下一个单词的词首
e	光标移至下一个单词的词尾
b	光标移至上一个单词的词首
{	光标向前移动一个段落
}	光标向后移动一个段落
H	光标移至当前屏幕首行的行首
M	光标移至当前屏幕中间
L	光标移至当前屏幕底部
n + H	光标移至当前屏幕第 n 行的行首（n 表示数字键）
n + G	光标移至文本的第 n 行的行首（n 表示数字键）
o（字母）	在光标下面插入一行
O（字母）	在光标上面插入一行
~	改变字母大小写



3. 剪贴板（见表1-3）

表1-3 剪贴板常用的命令

命令	说明
y + y	将当前行的内容复制到剪贴板
n + y + y	将当前开始的 n 行内容复制到剪贴板（n 表示数字键）
y + w	将光标所在位置的一个单词复制到剪贴板
n + y + w	将光标开始向右的总共 n 个单词复制到剪贴板（n 表示数字键）
p	将剪贴板的内容粘贴到光标所在的位置

4. 替换和删除文字（见表1-4）

表1-4 替换和删除文字常用的命令

命令	说明
r + c	r 表示取代（Replace），用字符 c 替换光标所指向的当前字符
n + r + c	用字符 c 替换光标所指向的前 n 个字符（n 表示数字键）
d + ←	可将光标所在位置的前一个字符删除
d + →	可将光标所在位置的字符删除
d + ↑	可将光标所在位置的行与上一行同时删除
d + ↓	可将光标所在位置的行与下一行同时删除
n + d + ↑	可将光标所在位置的行开始往上共 n 行删除（n 表示数字键）
n + d + ↓	可将光标所在位置的行开始往下共 n 行删除（n 表示数字键）
x	删除光标处的字符
n + x	删除从光标所在位置开始向右的 n 个字符（n 表示数字键）
n + X	删除从光标所在位置前的 n 个字符（n 表示数字键）
d	删除到当前行行末
d + w	若光标停在某单词的第一个字符上，则删除整个单词 若光标停在某单词的中间一个字符上，则从光标所在位置开始删至词尾并连同后面的空格 若光标停在两个单词之间，则删除从光标所在位置开始至下一个单词之前的所有空格
n + d + w	删除 n 个指定的单词（n 表示数字键）
d + b	删除光标所在位置之前的一个单词
n + d + b	删除光标所在位置之前的 n 个单词（n 表示数字键）
d + d	删除光标所在的整行
n + d + d	删除光标所在行开始的共 n 行内容（n 表示数字键）



5. 其他命令（见表1-5）

表1-5 其他常用的命令

命令	说明
Ctrl + g	在窗口的最后一行显示文本内容的总行数及目前光标所在的行数
R	同 r 键不同，按下 R 键后，一直处于替换字符状态，直到按 Esc 键为止
u	表示复原（Undo）的功能
.	重复执行上一个命令
Z + Z	存盘退出
Z + Q	作废退出
%	在光标位置移到其中一个括号时，按 % 键可找到对应的另一个括号

3.4 命令模式下的操作

要使用命令模式（Last Line Mode），请先按 Esc 键以确定回到 Command Mode，然后再按“:”、“?”或“/”等键都可进入 Last Line Mode。

命令模式常用的命令参见表 1-6。

表1-6 命令模式下常用的命令

命令	说明
:e 文件名	按文件名创建一个新文件
:n 文件名	在 Vi 中加载另一个文件
:f 文件名	把当前文件的文件名改为新文件名
:w	将文件保存。若原先没有指定文件名，则要输入“:w filename”
:q	退出 Vi。若文件内容已经被更改，则不作保存
:q!	强制退出 Vi，并且不保存文件
:wq	将当前文件内容存盘，并且退出 Vi
:set nu	在文件中的每行行首进入行号
:set nonu	取消行号
:n	光标移至该行的行首（n 表示数字键）
/pattern	从光标开始处向文件末尾搜索字符串 pattern。输入 n 在同一方向重复上一次搜索命令。输入 N 在反方向重复上一次搜索命令
?pattern	从光标开始处向文件首部搜索字符串 pattern。输入 n，在同一方向重复上一次搜索命令。输入 N，则在反方向重复上一次搜索命令
:s/str1/str2/g	将当前行中所有字符串 str1 均用 str2 替代
:g/str1/s//str2/g	将文件中所有字符串 str1 均用 str2 替代
:g/str1/s//str2/gc	在将文件中每一字符串 str1 用 str2 替代前，由用户输入“y”或“n”来决定是否要替换掉