


易水论坛开通vi/vim讨论区，欢迎到易水论坛交流和分享vi/vim使用经验！

定阅

 定阅本站 易水论坛

分类目录

- Web开发 (4)
- 吾家有女 (4)
- 嵌入系统 (10)
- 建站手记 (18)
- 开发编程 (9)
- 开源软件 (46)
- 文本处理 (51)
- 生活点滴 (0)
- 百宝箱 (13)
- 网络书签 (5)
- 网络微博 (14)

最近文章

- Easwy的Twitter微博(2010-05-30)
- Easwy的Twitter微博(2010-05-23)
- Easwy的Twitter微博(2010-05-16)
- 在scala中判断一个对象是否是元组(Tuple)
- Easwy的Twitter微博(2010-05-09)
- Easwy的Twitter微博(2010-04-25)
- Easwy的Twitter微博(2010-04-18)
- Bram将vim 7.3列入开发计划

最新评论

- Easwy: @dedekd 是的，要记住，通过插件动态生成的...
- Easwy: @dedekd 这是因为我们设置了这一选项：:set...
- Easwy: @dedekd 这个不是vim脚本，是shell...
- dedekd: lookupfile里面的hel...
- dedekd: :NERDTreeToggle [|]...
- dedekd: 好像是当so session时，session...
- dedekd: 我的session保存在了~/home /xxx下面
- dedekd: 我是右击选择terminal，然...

热门文章

- vi/vim使用进阶: 目录
- vi/vim使用进阶: 序言
- vi/vim使用进阶: vimrc初步
- vi/vim使用进阶: 在VIM中使用GDB调试 - 使用vimgdb
- vi/vim使用进阶: 智能补全
- vi/vim使用进阶: 自动补全
- vi/vim使用进阶: 使用标签(tag)文件
- vi/vim使用进阶: 使用会话和

首页 WordPress 关于

vi/vim使用进阶: lookupfile插件

vi/vim使用技巧: 在标签页中打开文件 (windows)



vi/vim使用进阶: 开启文件类型检测

Easwy

 添加评论

<< 返回vim使用进阶: 目录

本节所用命令的帮助入口:

```
:help filetype
:help setfiletype
:help modeline
:help 'modelines'
:help 'shiftwidth'
:help 'expandtab'
:help autocmd
```

打开文件类型检测功能很简单，在你的vimrc中加入下面一句话就可以了：

```
filetype plugin indent on
```

如果你用的是vim自带的示例vimrc，那么你已经打开了文件类型检测功能。或者，你也可以输入”**:filetype**”命令查看你的文件类型检测功能有没有打开。

这条命令究竟做了什么呢？我们在下面详细介绍。

其实，上面这一条命令，可以分为三条命令：

```
filetype on
filetype plugin on
filetype indent on
```

我们逐条介绍这三条命令。

”**filetype on**”命令打开文件类型检测功能，它相当于文件类型检测功能的开关。在执行”**filetype on**”命令时，vim实际上执行的是\$vimRUNTIME/filetype.vim脚本。这个脚本使用了自动命令(autocmd)来根据文件名来判断文件的类型，如果无法根据文件名来判断出文件类型，它又会调用\$vimRUNTIME/scripts.vim来根据文件的内容设置文件类型。有兴趣可以读一下这两个脚本，以获得更深的认识。

在上述步骤完成后，绝大多数已知类型的文件，都能被正确检测出文件类型。如果文件的类型未能被正确的检测出来，就需要手工设置文件类型，这可以通过”**set filetype**”完成，例如，如果你把main.c改名为main.c.bak1，那么它就无法被正确检测出文件类型。通过下面的Ex命令，就可以把它的文件类型设为c：

```
:set filetype=c
```

或者，你可以在文件中加入一个模式行，来指明这个文件的类型。vim在打开文件时，会在文件首、尾的若干行（行数由’**modelines**’选项决定，缺省为5行）检测具有vim特殊标记的行，称为模式行。如果检测到，就使用模式行中定义的选项值，来修改该缓冲区的选项。你可以留意一下vim的帮助页，每个文件的最后一行都是这样的模式行。

针对上例，我们可以在main.c.bak1的第一行或最后一行加上下面的内容，要指定这个文件的类型：

```
/* vim: ft=c */
```

这句话使用”/* */”注释起来了，因此不会影响编译。”**ft**”是”**filetype**”的缩写，vim中很多命令、选项都有缩写形式，以方便使用。注意”/*”与”**vim:**”间的空格。在”/*”与”**ft=c**”间，也需要有至少一个空格，这是模式行格式的要求，更多信息参阅”**:help modeline**”。

检测出文件的类型有什么作用呢？我们知道，不同类型的文件具有不同的格式，vim通过对文件类型的识别，可以为不同类型的文件，设置不同的选项值、定义不同键绑定等。例如，对于c类型的文件，它就和bash脚本有不同的注释格式、不同的缩进格式、不同的关键字等。vim在设置文件类型后，会触发FileType事件，执行FileType相关的自动命令，对不同类型的文件区别对待。

上面提到的”**filetype plugin on**”，允许vim加载文件类型插件。当这个选项打开时，vim会根据检测到的文件类型，在runtimepath中搜索该类型的所有插件，并执行它们。

viminfo

vi/vim使用进阶: 使用taglist插件

vi/vim使用进阶: 程序员的利器

- cscope

分类标签

android

apache

awk

cscope

ctags

cygwin

debian

delicious

docbook

easwy.com

git

google

java

linux

maven

mysql

scala

SEO

subversion

thinkpad

tomcat

twitter

Vim

windows

wordpress

xen

嵌入式Linux

幼儿园

育儿

调试

历史存档

2010年五月 (5)

2010年四月 (6)

2010年三月 (9)

2010年二月 (6)

2010年一月 (14)

2009年十二月 (3)

2009年十一月 (14)

2009年十月 (1)

2009年九月 (5)

2009年八月 (2)

2009年六月 (2)

2009年五月 (2)

2009年四月 (22)

2009年三月 (17)

2009年二月 (46)

链接表

Vimer的程序世界

“**filetype plugin on**”命令，实际上是执行\$vimRUNTIME/ftplugin.vim脚本，有兴趣可以读一下这个脚本。这个脚本中会设置自动命令，在runtimepath中搜索文件类型插件。

runtimepath的定义在不同的系统上不一样，对UNIX系统来说，这些路径包括：\$HOME/.vim、\$vim/vimfiles、\$vimRUNTIME、\$vim/vimfiles/after、\$HOME/.vim/after。

举一个例子，当我们对一个c类型的文件打开” filetype plugin on”时，它会在上述这几个目录的ftplugin子目录中搜索所有名为c.vim、c_*.vim，和c/*.vim的脚本，并执行它们。在搜索时，它按目录在runtimepath中出现的顺序进行搜索。缺省的，它会执行\$vimRUNTIME/ftplugin/c.vim，在这个脚本里，会设置c语言的注释格式、智能补全函数等等。

“**filetype indent on**”允许vim为不同类型的文件定义不同的缩进格式。这条命令也是通过一个脚本来完成加载：\$vimRUNTIME/indent.vim。和” **filetype plugin on**”类似，它也通过设置自动命令，在runtimepath的indent子目录中搜索缩进设置。对c类型的文件来说，它只是打开了cindent选项。

我们了解了文件类型检测的用途及它是如何工作的之后，就可以根据自己的需要，来对特定的文件类型进行特殊设置。

例如，我们在[上篇文章](#)中介绍过lookupfile插件，在它打开一个缓冲区时，会把缓冲区的filetype设置为lookupfile，我们可以利用这一点，在这个缓冲区里进行一些特殊的配置。例如，我们创建一个名为lookupfile.vim的文件，其内容为：

```
" close lookupfile window by two <Esc>
nnoremap <buffer> <Esc><Esc> <C-W>q
inoremap <buffer> <Esc><Esc> <Esc><C-W>q
```

它定义了两个局部于缓冲区的键绑定，无论在normal模式还是插入模式，只要连接两次ESC，就关闭当前缓冲区。

把这个文件保存到你的runtimepath所指向任一目录的ftplugin子目录中（一般是放在~/.vim/ftplugin目录中）。你在下次打开lookupfile窗口时，试试连接两次ESC，是不是lookupfile窗口就关闭了？这样设置，非常适合vim中按ESC取消命令的习惯，效率也高。

如果你对vim缺省文件类型插件的设置不太满意，那么可以把这个全局插件拷贝到\$HOME/.vim/plugin目录中，然后更改其中的设置。你可以留意一下vim缺省的文件类型插件，它们的头部都有这样的代码：

```
" Only do this when not done yet for this buffer
if exists("b:did_ftplugin")
    finish
endif
```

这类似于C语言头文件中的” #ifndef XXX | #define XXX”的语句，可以防止该插件被执行多次。因此，把这个插件拷贝到\$HOME/.vim/plugin中（这个目录在runtimepath中排在最前面），它将先于vim的缺省插件被加载；而它加载后，vim的缺省文件类型插件就不会再被加载了。这就达到了我们修改设置的目的。

不过我们通常不用这种方法。如果这样做，一旦vim的缺省插件做了改变，我们也要更新我们改过的插件才行。我们可以在载入全局插件以后否决一些设置。在Unix 上，我们可以把我们的设置放在~/.vim/after/ftplugin/目录中，这个目录中的脚本会在vim的缺省脚本后执行。这样就可以修改配置，或增加我们自己的定义。

举个例子，在多人一起开发项目时，为了使代码风格尽量保持一致，一般不允许在代码使用TAB符，而以4个空格代之。我们可以编辑一个文件，包含下面的内容：

```
set shiftwidth=4
set expandtab
```

把这个文件保存为~/.vim/after/ftplugin/c.vim。这样，每次在编辑c文件时，它的自动缩进为4个空格；当你在插入模式下使用CTRL-D、CTRL-T缩进时，它也会调整4个空格的缩进；当你按TAB键时，它将会插入8个空格……如果你想上面的设置对h文件也生效的话，需要把它另存一份：~/.vim/after/ftplugin/cpp.vim，因为h文件的文件类型被设为cpp。

我们知道，vim在设置文件类型时，会触发FileType自动命令，因此，上面的例子可以用下面的自动命令来实现：

```
autocmd FileType c,cpp set shiftwidth=4 | set expandtab
```

把这个命令放在你的vimrc中，可以起到和上例同样的效果。

vim的语法高亮功能，也要用到文件类型，来对不同的关键字进行染色。这我们将在[下一篇文章](#)中介绍。

【参考文档】

- vim手册
- [vim中文手册](#)

<< [返回vim使用进阶: 目录](#)

原创文章，请阅读页脚的[许可方式](#)，转载请注明：转载自[易水博客](#) [<http://easwy.com/blog/>]

本文链接地址: <http://easwy.com/blog/archives/advanced-vim-skills-filetype-on/>

文章脚注信息由WordPress的[wp-posturl](#)插件自动生成

相关文章

- vi/vim使用进阶: 目录
- vi/vim使用进阶: vimrc初步
- vi/vim使用进阶: 智能补全
- vi/vim使用进阶: 乱花渐欲迷人眼 - 语法高亮
- vi/vim使用进阶: 使用taglist插件
- vi/vim使用进阶: 剑不离手 - quickfix
- vi/vim使用进阶: 在VIM中使用GDB调试 - 使用pyclevn
- vi/vim使用进阶: 文件浏览和缓冲区浏览

 文本处理

 Vim

 1条评论

 添加评论

[寻觅/14 >>](#)

[Blog](#)

[Archive >>](#)

[vim使用进阶](#)

2009/07/09 20:38 | #1

[回复](#) | [引用](#)

[...] vim使用进阶: 文件浏览和缓冲区浏览 vim使用进阶: lookupfile插件 vim使用进阶: 开启文件类型检测 vim使用进阶: 乱花渐欲迷人眼 - 语法高亮 vim使用进阶: 程序员的利器 - cscope [...]

 评论本文

姓名 (必填)

邮件 (不会显示) (必填)

网站

5

6

提交评论

易水博客采用署名-非商业性使用-相同方式共享的创作共用许可协议，转载文章请遵守该协议。
由Wordpress搭建，iTheme-i18n主题由N.Design Studio设计，Easwy定制。通过XHTML 1.1和CSS 3验证。