gf or C-w (* indicates cursor position)

Pos*t.first	app/models/post.rb
has_many :c*omments	app/models/comment.rb
link_to 'Home', :controller => 'bl*og'	app/controllers/blog_controller.rb
<%= render 'sh*ared/sidebar' %>	app/views/shared/_sidebar.html.erb
<%= stylesheet_link_tag 'appl*ication' %>	app/assets/stylesheets/application.css
class BlogController < Applica*tionController	app/controllers/application_controller.rb
class ApplicationController < ActionCont*roller::Base	.../action_controller/base.rb
fixtures :pos*ts	test/fixtures/posts.yml
layout :pri*nt	app/views/layouts/print.html.erb
<%= link_to "New", new_comme*nt_path %>	app/controllers/comments_controller.rb (jumps to def new)

In the last example, the controller and action for the named route are determined by evaluating routes.rb as Ruby and doing some introspection. This means code from the application is executed. Keep this in mind when navigating unfamiliar applications.

Plugins

(GH: any-jump.vim) `:help any-ju mp.vim`
Normal mode: Jump to definition under cursor nnoremap <leader>j :AnyJump<CR>
Visual mode: jump to selected text in visual mode xnoremap <leader>j :AnyJumpVisual<CR>
From Modal view preview the definition/reference p
(GH: vim-test) `:help test`
Normal mode: In a test file runs all tests in the current file nmap <silent> <leader>t :TestFile<CR>
Normal mode: Runs the whole test suite nmap <silent> <leader>T :TestSuite<CR>
(GH: vim-bundle)
Normal mode: nmap <silent> <leader>b :Bundle<CR>
Normal mode: will open the gem source code in a new tab :Bundle open actionview
Normal mode: pen the source code in a split window instead of a new tab :rightbelow Bundle open actionview
(GH: splitjoin.vim) `:help splitjoin`
Normal mode: to split a one-liner into multiple lines gS
Normal mode: (with the cursor on the first line of a block) to join a block into a single-line statement. gJ