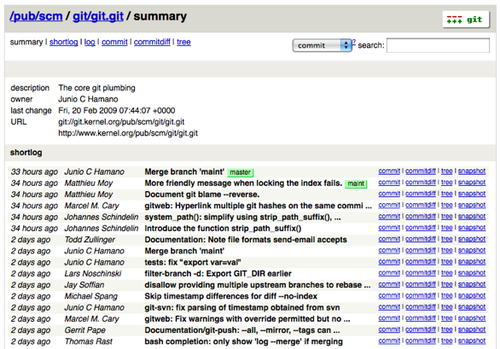
现在我们的项目已经有了可读可写和只读的连接方式，不过如果能有一个简单的 web 界面访问就更好了。Git 自带一个叫做 GitWeb 的 CGI 脚本，运行效果可以到 http://git.kernel.org 这样的站点体验下（见图 4-1）。

  
Figure 4-1. 基于网页的 GitWeb 用户界面

如果想看看自己项目的效果，不妨用 Git 自带的一个命令，可以使用类似 lighttpd 或 webrick 这样轻量级的服务器启动一个临时进程。如果是在 Linux 主机上，通常都预装了 lighttpd ，可以到项目目录中键入 git instaweb来启动。如果用的是 Mac ，Leopard 预装了 Ruby，所以 webrick 应该是最好的选择。如果要用 lighttpd 以外的程序来启动 git instaweb，可以通过 --httpd 选项指定：

$ git instaweb --httpd=webrick

[2009-02-21 10:02:21] INFO WEBrick 1.3.1

[2009-02-21 10:02:21] INFO ruby 1.8.6 (2008-03-03) [universal-darwin9.0]

这会在 1234 端口开启一个 HTTPD 服务，随之在浏览器中显示该页，十分简单。关闭服务时，只需在原来的命令后面加上 --stop 选项就可以了：

$ git instaweb --httpd=webrick --stop

如果需要为团队或者某个开源项目长期运行 GitWeb，那么 CGI 脚本就要由正常的网页服务来运行。一些 Linux 发行版可以通过 apt 或 yum 安装一个叫做 gitweb 的软件包，不妨首先尝试一下。我们将快速介绍一下手动安装 GitWeb 的流程。首先，你需要 Git 的源码，其中带有 GitWeb，并能生成定制的 CGI 脚本：

$ git clone git://git.kernel.org/pub/scm/git/git.git

$ cd git/

$ make GITWEB\_PROJECTROOT="/opt/git" \

prefix=/usr gitweb

$ sudo cp -Rf gitweb /var/www/

注意，通过指定 GITWEB\_PROJECTROOT 变量告诉编译命令 Git 仓库的位置。然后，设置 Apache 以 CGI 方式运行该脚本，添加一个 VirtualHost 配置：

<VirtualHost \*:80>

ServerName gitserver

DocumentRoot /var/www/gitweb

<Directory /var/www/gitweb>

Options ExecCGI +FollowSymLinks +SymLinksIfOwnerMatch

AllowOverride All

order allow,deny

Allow from all

AddHandler cgi-script cgi

DirectoryIndex gitweb.cgi

</Directory>

</VirtualHost>

不难想象，GitWeb 可以使用任何兼容 CGI 的网页服务来运行；如果偏向使用其他 web 服务器，配置也不会很麻烦。现在，通过 http://gitserver 就可以在线访问仓库了，在 http://git.server 上还可以通过 HTTP 克隆和获取仓库的内容。