Introducción Ruby on Rails

Instalación - Editores - Git

Ruby - Manejo de versiones

Es común que sea necesario manejar varias versiones de ruby cuando se trata de desarrollos grandes y diversos.

Por ello se recomienda la instalación con RBENV, que sería un entorno virtual de programación, que permite la instalación de varias versiones de ruby/rails, y su elección para cada proyecto o globalmente.

Instalación - Ubuntu


```
#Actualizar los paquetes
sudo apt-get update -y

# Instalamos git
sudo apt-get install -y git gitk git-gui

# Instamamos el compilador de C y las librerías requeridas por rbenv para
construir los binariosd e ruby
sudo apt-get install -y gcc build-essential libpq-dev libssl-dev
libreadline-dev libsqlite3-dev zlib1g-dev

# Instrucciones del sitio web de RBENV en Github
# https://github.com/sstephenson/rbenv
# Clonar el repositorio git de rbenv en ~/.rbenv
cd
git clone git://github.com/sstephenson/rbenv.git .rbenv
```

Instalación - Ubuntu 2

```
#Agregar "rbenv init" al shell para habilitar autocomplete
echo 'eval "$(rbenv init -)"' >> ~/.bashrc
tail ~/.bashrc

#Agregar los contenidos modificados a la sesión actual
set +e
source ~/.bashrc
set -e
which rbenv
rbenv help

#Agregar el comando install al rbenv
git clone git://github.com/sstephenson/ruby-build.git
~/.rbenv/plugins/ruby-build
echo 'export PATH="$HOME/.rbenv/plugins/ruby-build/bin:$PATH"' >> ~/.bashrc
tail ~/.bashrc
```

Instalación - Ubuntu 3

```
#Source de la nueva ubicación del path en la sesión actual y verificar que tengamos el comando “install”
set +e
source ~/.bashrc
set -e
which rbenv
rbenv help | grep install

# Instalar una versión de Ruby (puede ser otra)
rbenv install -v 2.2.2

# Setear la versión del ruby global a usar
rbenv global 2.2.2
ruby -v
```

Instalación - Ubuntu


```
#Setear por defecto que las gemas no generen documentación local
echo "gem: --no-document" > ~/.gemrc
gem install bundler

#Instalar rails (~5min)
gem install rails -v 4.2.3
rails -v

# install shims for newly installed Ruby gems that provide commands
rbenv rehash
```

Instalación - Ubuntu 5

```
#Instalar Node.js
sudo apt-get install -y software-properties-common python-software-properties
sudo add-apt-repository ppa:chris-lea/node.js
sudo apt-get update -y
sudo apt-get install -y nodejs

#install phantomJS
sudo apt-get install -y bzip2
export PHANTOM_JS="phantomjs-1.9.8-linux-x86_64"
cd /tmp
curl -L https://bitbucket.org/ariya/phantomjs/downloads/$PHANTOM_JS.tar.bz2 |
tar xvjf -
sudo mv $PHANTOM_JS /usr/local/share
sudo ln -sf /usr/local/share/$PHANTOM_JS/bin/phantomjs /usr/local/bin
phantomjs --version
```

Editor - Sublime Text 2

```
cd /tmp
curl http://c758482.r82.cf2.rackcdn.com/Sublime%20Text%202.0.2%20x64.tar.bz2 |
tar -xjf -
sudo mv 'Sublime Text 2' /opt/SublimeText2
echo export PATH='$PATH:/opt/SublimeText2' >> ~/.bashrc

#inspect installation
tree ~/ -L 1
tree ~/.rbenv -L 1
```

Test / Crear una aplicación rails

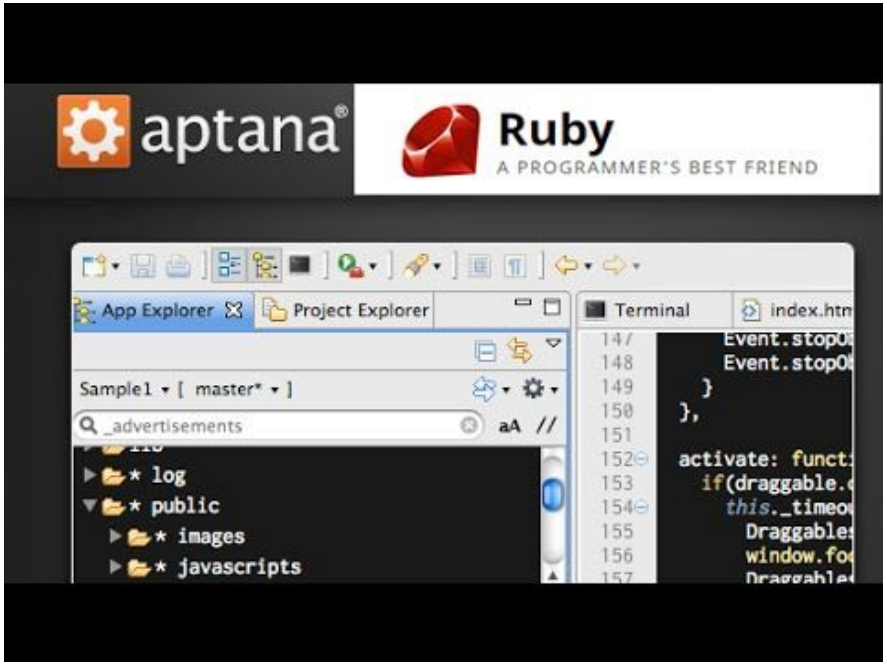
```
#crear aplicación
rails new test-install -q

#levantar el servidor rails
cd test-install
bundle install

rails server
```

http://localhost:3000

IDEs


aptana
Ruby
A PROGRAMMER'S BEST FRIEND
App Explorer
Project Explorer
Terminal
index.htm
Sample1 [master*]
_advertisements
aA //
log
public
images
javascripts
activate: funct
if(draggable.
this._timeo
Draggable
window.fo


JetBrains RubyMine 1.1.1
File Edit Search View Go To Code Analyze Refactor Run Tools Version Control Window Help
RubyMine
WELCOME!
Develop with pleasure!
Quick Start
Create New Project
Create a new project
Open Directory
Start editing code in the specified directory
Check out from Version Control
You can check out an entire project from a Version Control System. Click the icon or link to select your VCS.
Documentation
Read Help
Open JetBrains RubyMine "Help Topics" in a new window.
RubyMine
Develop with pleasure!
Plugins
Open Plugin Manager
My Plugins:
No plugins currently installed.
Bundled Plugins:
CSS Support
Provides syntax and error highlighting, completion, navigation and other editing...
CVS Integration
Provides integration with CVS version control system
Git Integration
Provides integration with Git version control system
HTML Preview
Provides preview for HTML files using Mozilla browser (not available on 64bit platforms)
Images
Provides image viewing and thumbnail browsing
Inspection-JS
Adds over 100 new automated code inspections for auditing JavaScript.
JavaScript Intention Power Pack
Adds over 40 new JavaScript intention actions.
JavaScript Support
Provides support for the JavaScript language
Perforce Integration
Provides integration with Perforce version control system
50M of 91M

Sublime Text 2

File/preferences/color scheme

File/Preferences/Settings Default

File/Preferences/Settings User

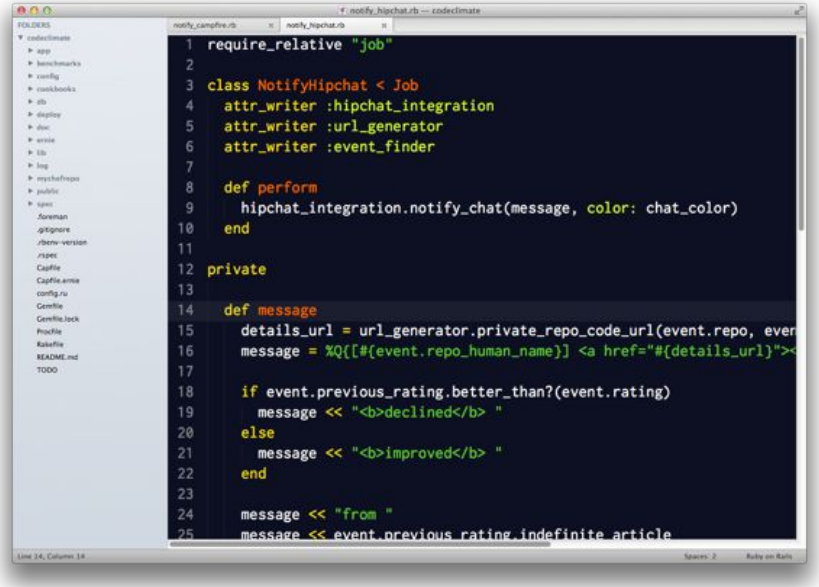
Layout two columns

Find in files // control+Shift+f

Go to anything // control+p

Collapsing code

Tools/Build



Sublime Text / Packages

Control + Shift + p

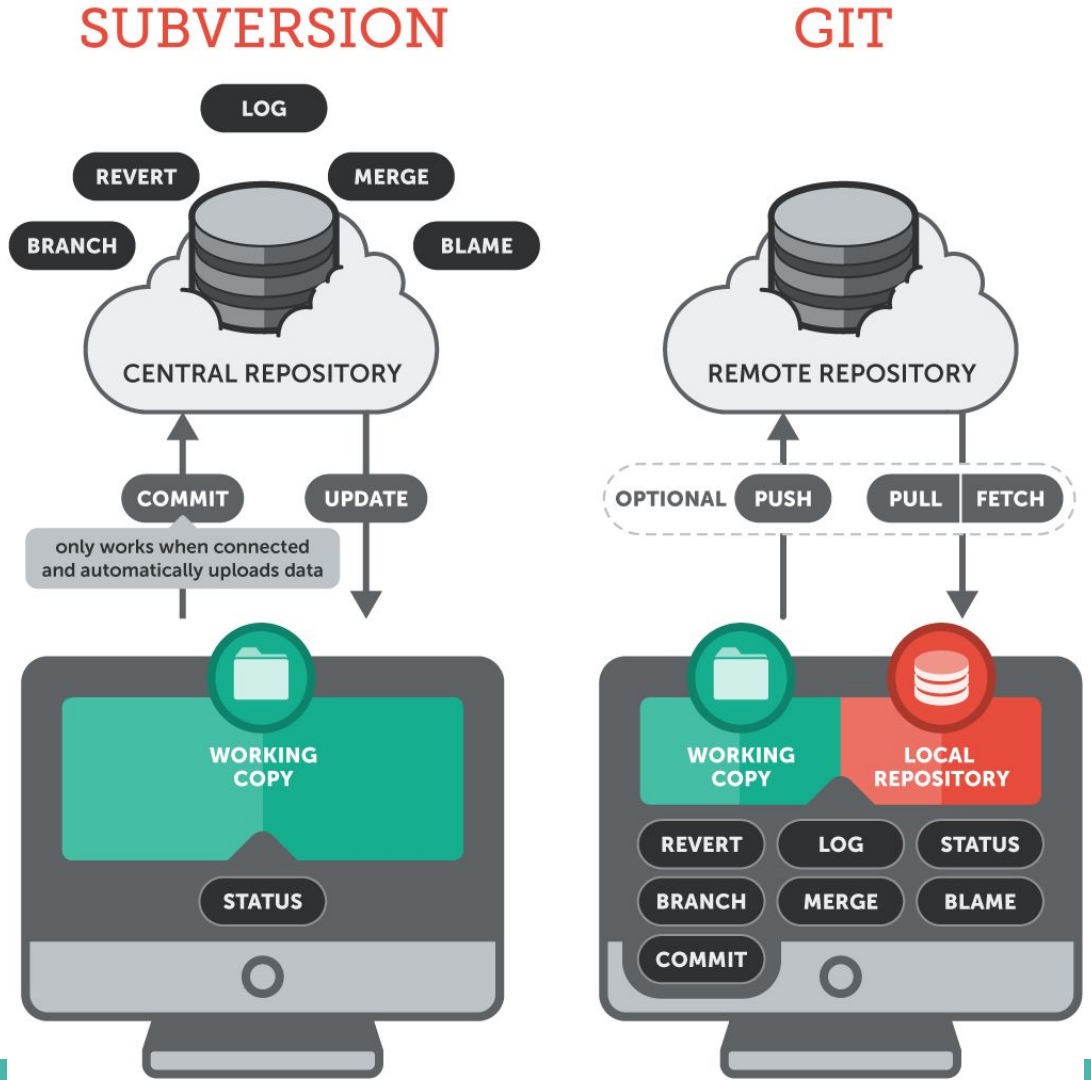
http://packagecontrol.io

Git

Es un sistema de control de versiones.

Es un sistema distribuido. Al igual que mercurial.

Se mantiene toda la historia en el servidor central y en la copia local.


SUBVERSION
LOG
REVERT
MERGE
BRANCH
BLAME
CENTRAL REPOSITORY
COMMIT
UPDATE
only works when connected
and automatically uploads data
WORKING
COPY
STATUS
GIT
REMOTE REPOSITORY
OPTIONAL
PUSH
PULL
FETCH
WORKING
COPY
LOCAL
REPOSITORY
REVERT
LOG
STATUS
BRANCH
MERGE
BLAME
COMMIT

Todo el repositorio de un sistema de versiones distribuido reside en?

Localmente!

Git Basics

- Solo existe un .git que está en la raíz del repositorio.
- Flujo:
  - Create or clone.
  - Add changes en el área de stage.
  - Commit de cambios al repositorio **local**.
  - Push de cambios al repositorio **remoto**.
- **Referencias:**
  - www.git-scm.org
  - www.gitreference.org

Git Quizz

Cual es la abreviación utilizada para describir la función de Git?

- ☐ SCM - Source Code Management
- ☐ VCS - Version Control System
- ☐ SSH - Secure Socket Shell
- ☐ HTTP - Hypertext Transfer Protocol

Cuales son las ventajas de un Sistema de Control de Versiones Distribuido, como Git?

- ☐ No se requiere conexión a internet para operaciones de git comunes.
- ☐ Los backups son triviales
- ☐ Las funciones de git son exactamente iguales a las de CVS y SVN.

Git Quizz

Cual es la abreviación utilizada para describir la función de Git?

- [x] SCM - Source Code Management
- [x] VCS - Version Control System
- [ ] SSH - Secure Socket Shell
- [ ] HTTP - Hypertext Transfer Protocol

Cuales son las ventajas de un Sistema de Control de Versiones Distribuido, como Git?

- [x] No se requiere conexión a internet para operaciones de git comunes.
- [x] Los backups son triviales
- [ ] Las funciones de git son exactamente iguales a las de CVS y SVN.

Gil local

1

```
$git config --global user.name “Kalman Hazins”

$git config --global user.email my@example.com

$git config user.name

$git help <command name>
```

Gil local

De donde quitamos un repositorio?... Creamos uno

```
$cd workding_dir

$git init

(Possibly create a .gitignore file)

$git add .

$git commit -m “Initial commit”
```

Gil local

De donde quitamos un repositorio?... Creamos uno

Clonamos un repositorio existente (Por ejemplo desde Github)

```
$git clone https://repourl.git
```

Comandos Git

git status

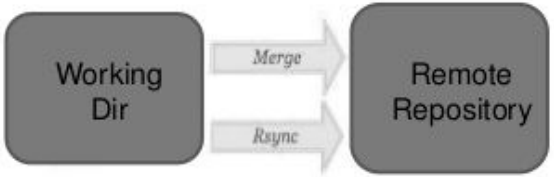
git add <file/dir> (agregar archivos/directorios no trackeados, o modificados en el stage)

git diff Muestra la diferencia entre el staging y el directorio de trabajo.

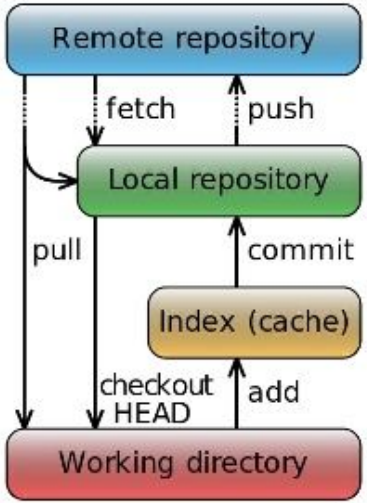
git diff --staged Muestra los cambios entre el HEAD y el staging

git diff HEAD Muestra las diferencias entre el HEAD y el directorio de trabajo.

SVN Dataflow



Git Data Flow



CARBOOK PLUS

Flujo SVN vs. Git

Comandos Git

git commit

Comitea los cambios en el repositorio.

Abre un prompt para ingresar un mensaje

Si se utiliza -m se puede incluir en el comando el mensaje.

git commit -m “Your msg here”

git commit -a -m “” (Saltarse el staging area)

Volver en el tiempo

```
# Before committing

git checkout . # Re-checkout de todos los archivos omitiendo los cambios

git checkout -- <file> # Re-checkout un archivo específico

# After committing

git revert HEAD # Reverts revierte el commit más reciente
```

Remote Repos – Set Up and Push

- **git remote add** alias remote_url
  - Se enlaza un repositorio remoto al repositorio local
- git remote -v
- El alias por default para el repositorio es **origin**
- **git push alias** branch_name
  - Envía los cambios a un remote en un determinado branch

Ejercicio

1. Clonar el repositorio
2. Usuario a, realiza un cambio.
3. Usuario b, realiza otro cambio en el mismo archivo.
4. Ambos realizan un comitt a su repositorio local.
5. Usuario a, realiza otro cambio sobre el mismo archivo. **Qué pasó?**
6. Pull del repositorio remoto.
7. Resolver merges si los hay
8. Push al repositorio remoto.