
Introducción Ruby on Rails

— Instalación - Editores - Git —



WINDOWS



Instalar

- Ruby - version 2+
- Rails - version 4.2.3
- Git
 - <http://railsinstaller.org/en>
- Phantomjs
 - <http://phantomjs.org/>
- Sublime Text 3
 - <https://www.sublimetext.com/>

Download

Windows Downloads

↓ WINDOWS
RUBY 2.1

Packages included are

[Ruby](#) 2.1.8

[Rails](#) 4.2

[Bundler](#)

[Git](#)

[Sqlite](#)

[TinyTDS](#)

[SQL Server Support](#)

[DevKit](#)

↓ WINDOWS
RUBY 2.2

Packages included are

[Ruby](#) 2.2.4

[Rails](#) 4.2

[Bundler](#)

[Git](#)

[Sqlite](#)

[TinyTDS](#)

[SQL Server Support](#)

[DevKit](#)

Mac Downloads

↓ MAC OSX
10.7 & 10.8

Packages included are

[Ruby](#) 1.9.3

[Rails](#) 4.1

[Bundler](#)

[Git](#)

[Sqlite](#)

[osx-gcc-installer](#)

[JewelryBox](#)

[RVM](#)

[SM Framework](#)

↓ MAC OSX
10.6

Descargar e instalar
manteniendo los
defaults.

Se instala en
C:/RailsInstaller

git --version

ruby -v

rails -v

PhantomJs

- Descargar y se almacenar en algún directorio.
- Agregar el directorio/bin al path de windows. (Agregar ; antes de pegar la dirección).
- **phantomjs -v**

Sublime Text 3

Download

Sublime Text 3 is currently in beta. The latest build is 3126.

- [OS X](#) (10.7 or later is required)
- [Windows](#) - also available as a [portable version](#)
- [Windows 64 bit](#) - also available as a [portable version](#)
- [Ubuntu 64 bit](#) - also available as a [tarball](#) for other Linux distributions.
- [Ubuntu 32 bit](#) - also available as a [tarball](#) for other Linux distributions.

Sublime Text may be downloaded and evaluated for free, however a license must be [purchased](#) for continued use. There is currently no enforced time limit for the evaluation.

For notification about new versions, follow [sublimehq](#) on twitter.

Even more bleeding-edge versions are available in the [dev builds](#).

[Sublime Text 2](#) is also still available.



MAC



Instalar

- Ruby - version 2+
- Rails - version 4.2.3
- Git
 - <http://railsinstaller.org/en>
- Phantomjs
 - <http://phantomjs.org/>
- Sublime Text 3
 - <https://www.sublimetext.com/>

Pasos resumidos...

https://drive.google.com/open?id=0B9OkClVf_we2amlFX3pWNzBGaVE

LINUX - UBUNTU

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.

<https://github.com/rbenv/rbenv#rbenv-versions>

Instalación - Ubuntu

1

#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 binarios de 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

4

#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

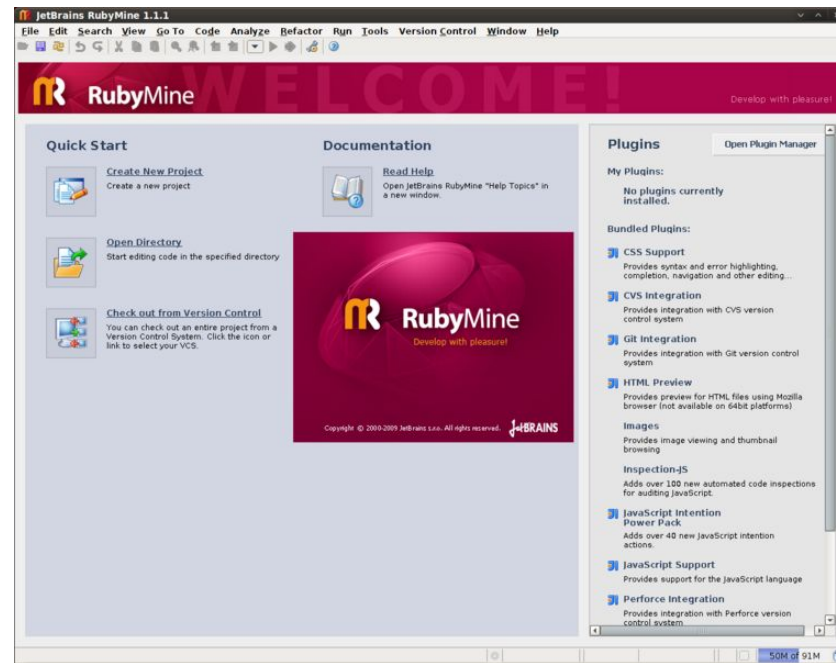
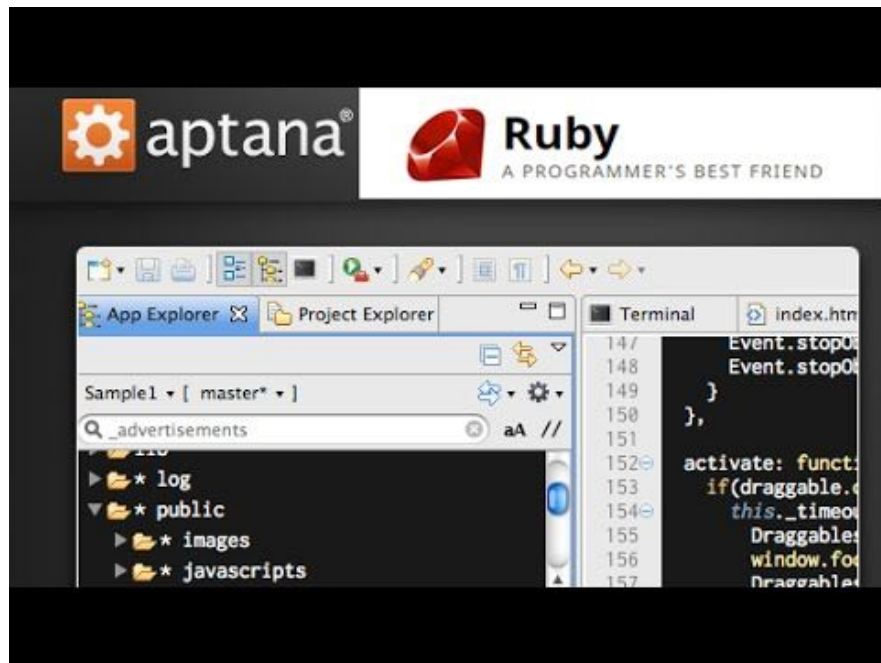
```
cd test-install
```

```
bundle install
```

```
rails server
```

<http://localhost:3000>

IDEs



Sublime Text 2

File/preferences/color scheme

File/Preferences/Settings Default

File/Preferences/Settings User

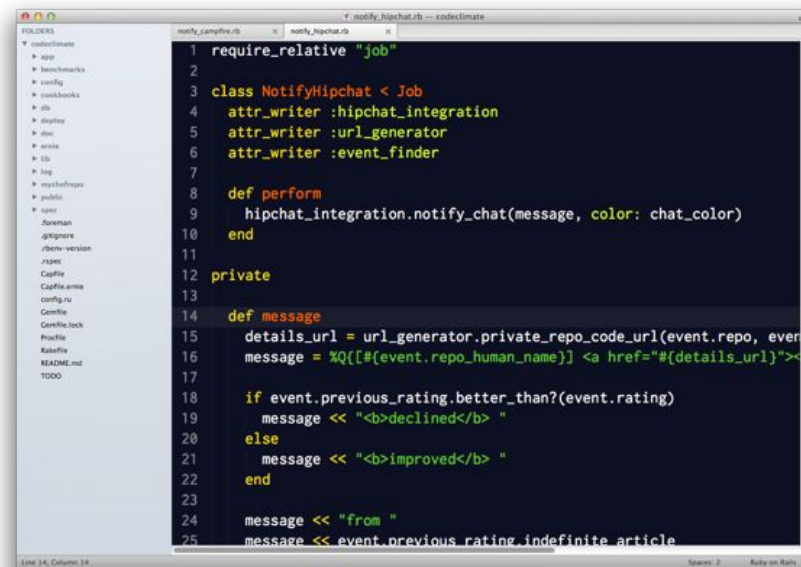
Layout two columns

Find in files // control+Shift+f

Go to anything // control+p

Collapsing code

Tools/Build



Sublime Text / Packages

Control + Shift + p

<http://packagecontrol.io>

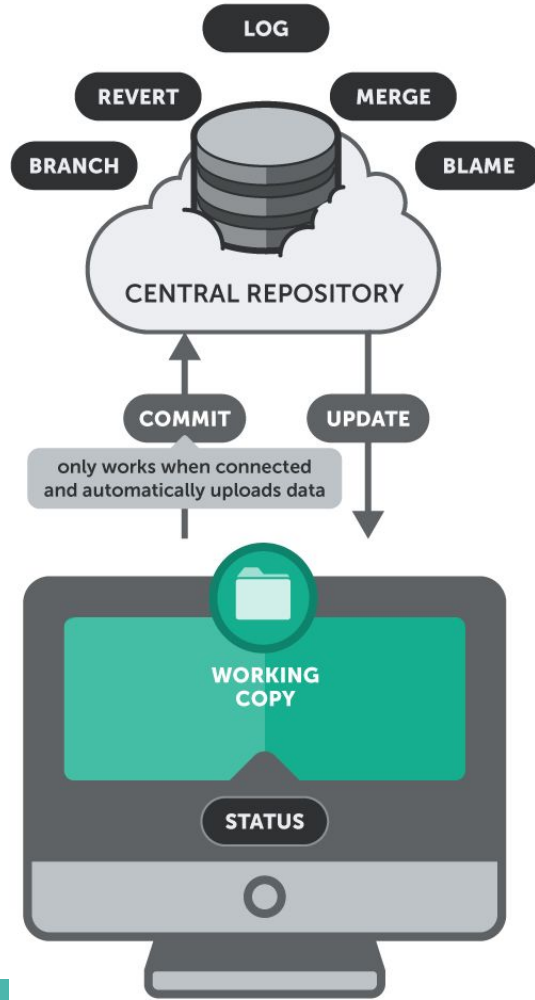
Git

Es un sistema de control de versiones.

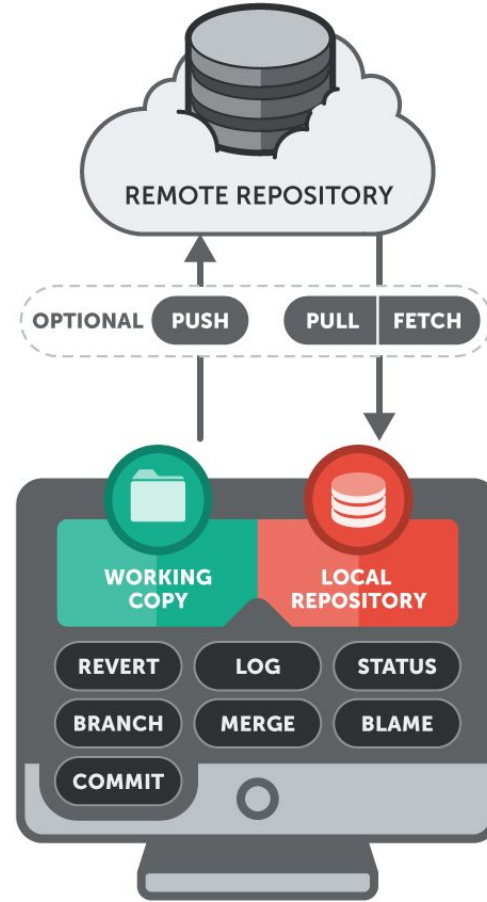
Es un sistema distribuido. Al igual que mercurial.

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

SUBVERSION



GIT



**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?

- ☒ 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.

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

2

De donde quitamos un repositorio?... Creamos uno

```
$cd working_dir
```

```
$git init
```

(Possibly create a .gitignore file)

```
$git add .
```

```
$git commit -m "Initial commit"
```

Gil local

4

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)

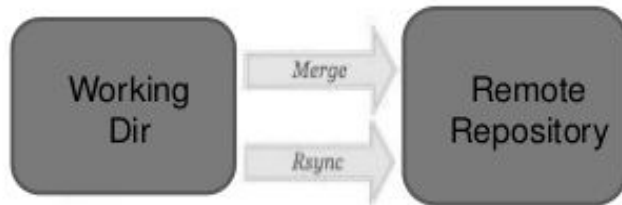
git diff Muestra la diferencia entre el staging y el directorio de trabajo.

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

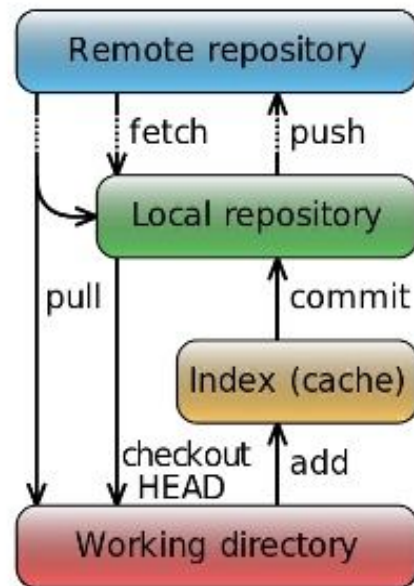
git diff HEAD Muestra las diferencias entre el HEAD y el directorio de trabajo.



SVN Dataflow



Git Data Flow



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.