

Laboratorio de Computación IV

Clase 3

Andrés Fortier

¿Consultas?

- rvm / Ruby
- Ejercicios
 - 1 y 6 de la Práctica 2 de Programación 3.
 - 7 de la Práctica 2 de Programación 3.

Comando de la semana: grep

- grep == búsquedas en archivo (o stream).
- Bajar el archivo del sitio.

```
$ ls -l
drwxrwxr-x 2 andres andres 4096 mar 21 18:40 don_rodrigo
-rw-rw-r-- 1 andres andres  72 mar 21 18:33 texto1
-rw-rw-r-- 1 andres andres 2179 mar 21 18:36 texto2
-rw-rw-r-- 1 andres andres  231 mar 21 18:43 texto3
```

```
$ grep "de" texto1
primera línea de texto
segunda línea de texto
```

Comando de la semana: grep

- Buscar en todos los archivos del directorio

```
$ grep "de" *  
grep: don_rodrigo: Is a directory  
texto1:primera línea de texto  
texto1:segunda línea de texto  
texto2:Los críticos recién comenzaron a apreciar las  
obras de Mastropiero, cuando ya era grandecito. Cuando  
ya eran GRANDES HITOS... en la historia de la música.  
texto2:Por ejemplo, un conocido crítico se resfrió. Se  
REFIRIÓ... se refirió a Mastropiero; punto. Con esto  
termino. Con ESTOS TÉRMINOS... le falta el... con estos  
términos... no le han puesto... ¿no ves?... arriba en  
la "t"... ahí... a veces se borra... ¿no ves?... la  
diéresis, ¿no entiendes?...  
texto2:Mastropiero se ha creado fama de artista  
espiritual pero
```

Comando de la semana: grep

- Buscar en forma recursiva

```
$ grep -r "de" *
```

- *Case-insensitive*

```
$ grep -i "texto" *
```

- Mostrar número de línea

```
$ grep -n "texto" *
```

- Líneas que no satisfacen la búsqueda

```
$ grep -v "texto" *
```

Comando de la semana: grep

- Mostrar N líneas antes/después

```
$ grep -B 2 -A 3 "Mastropiero" *
```

- Contar las coincidencias por archivo

```
$ grep -c "Mastropiero" *
```


- Sólo palabras completas

```
$ grep "primera" texto1
```

```
$ grep -w "primera" texto1
```

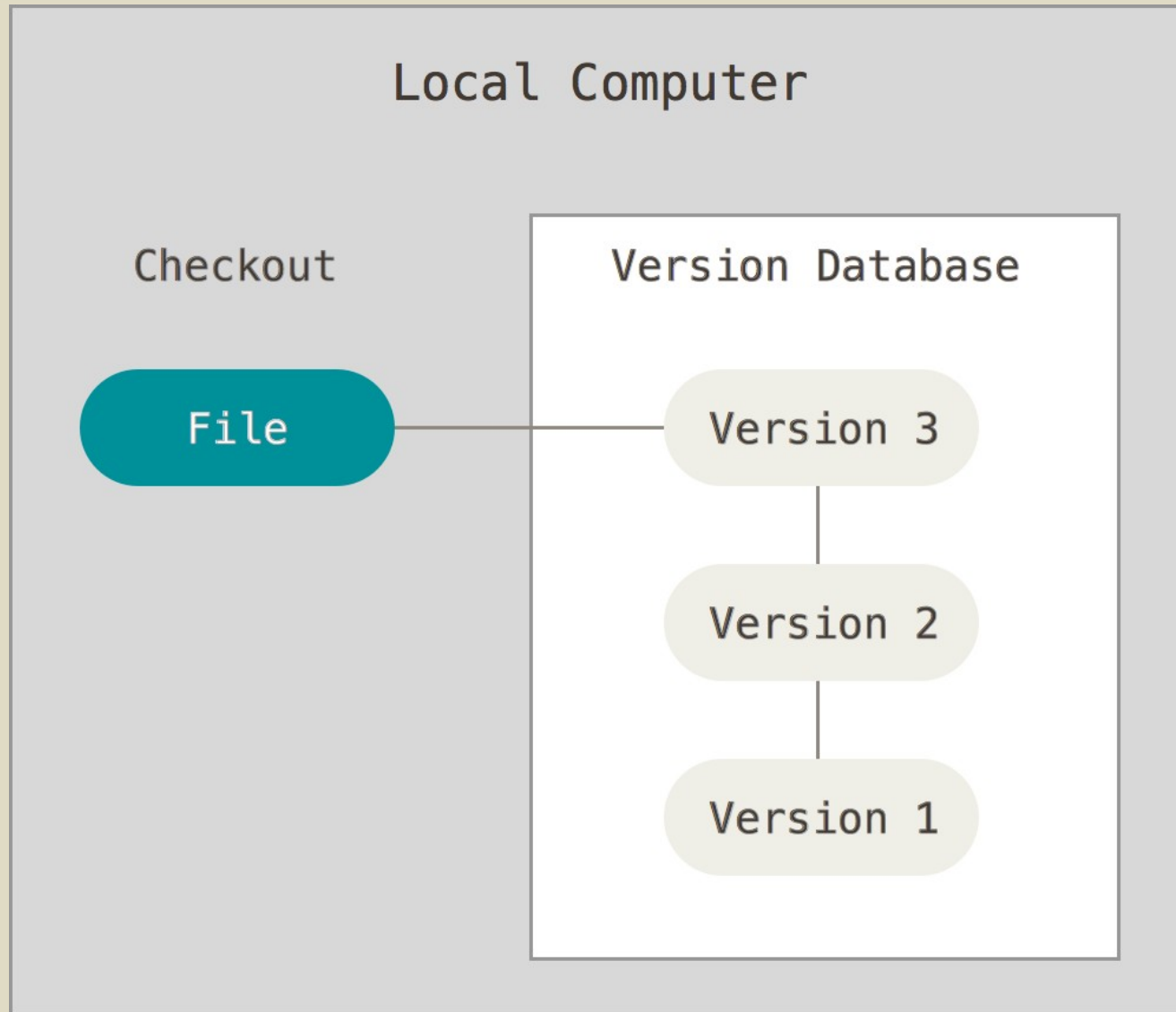
- Expresiones regulares

Version Control Systems (VCSs)

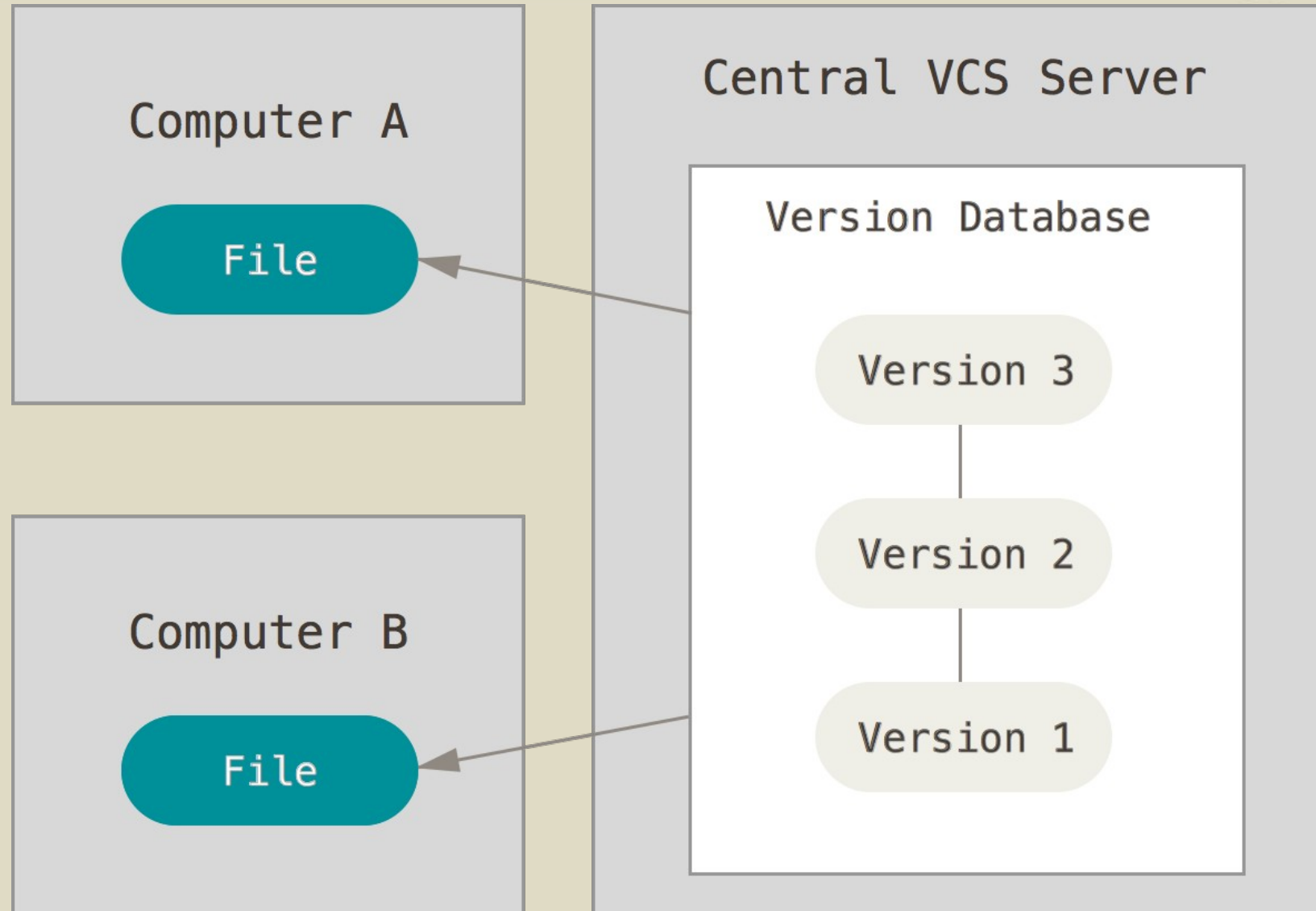


- Registra los cambios en uno o mas archivos a medida que pasa el tiempo.
- Permite
 - Revertir a un estado anterior.
 - Comparar cambios.
 - Determinar quién realizo un determinado cambio.

Version Control Systems (VCSs)



Version Control Systems (VCSs)



Version Control Systems (VCSs)

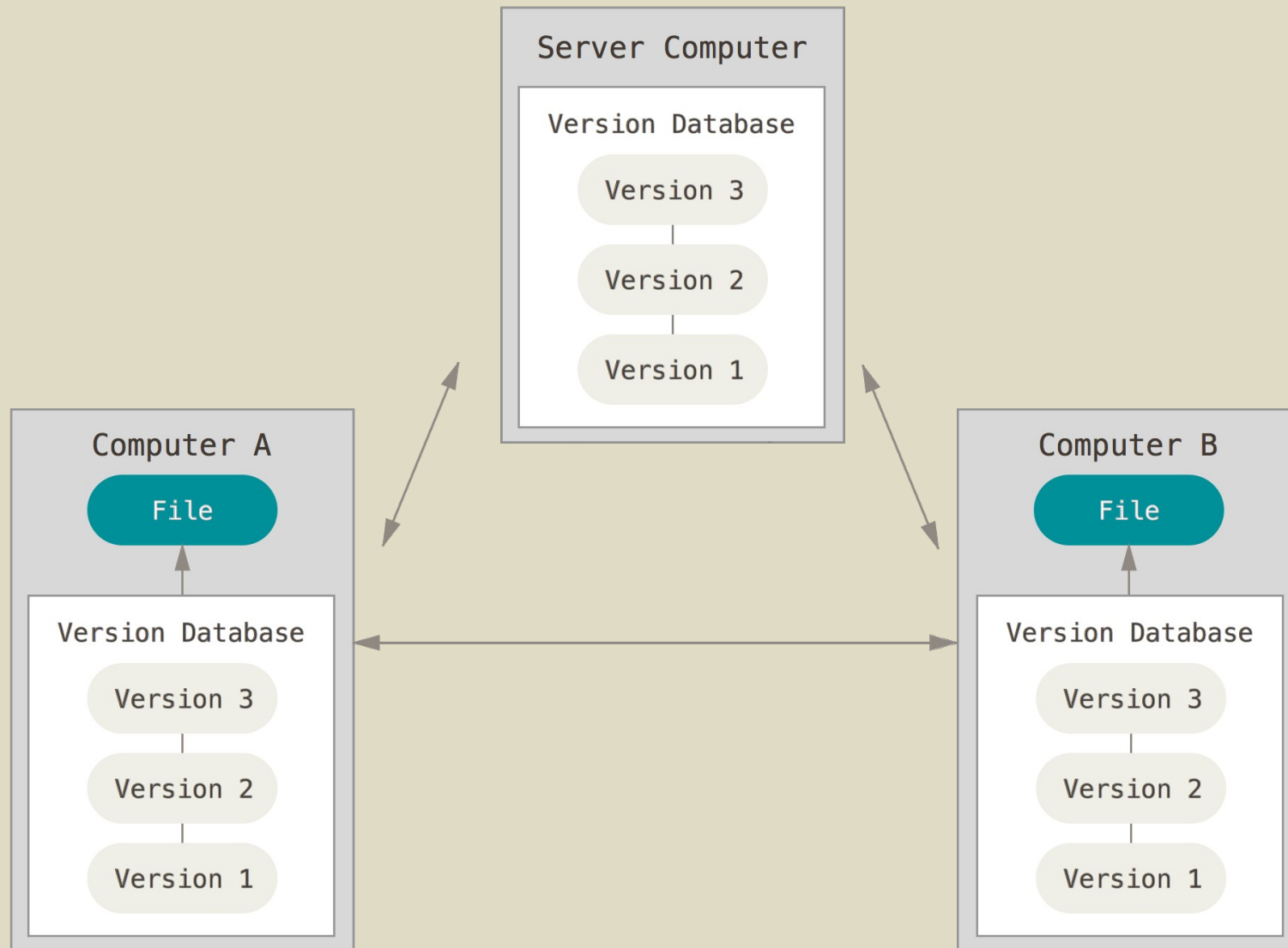
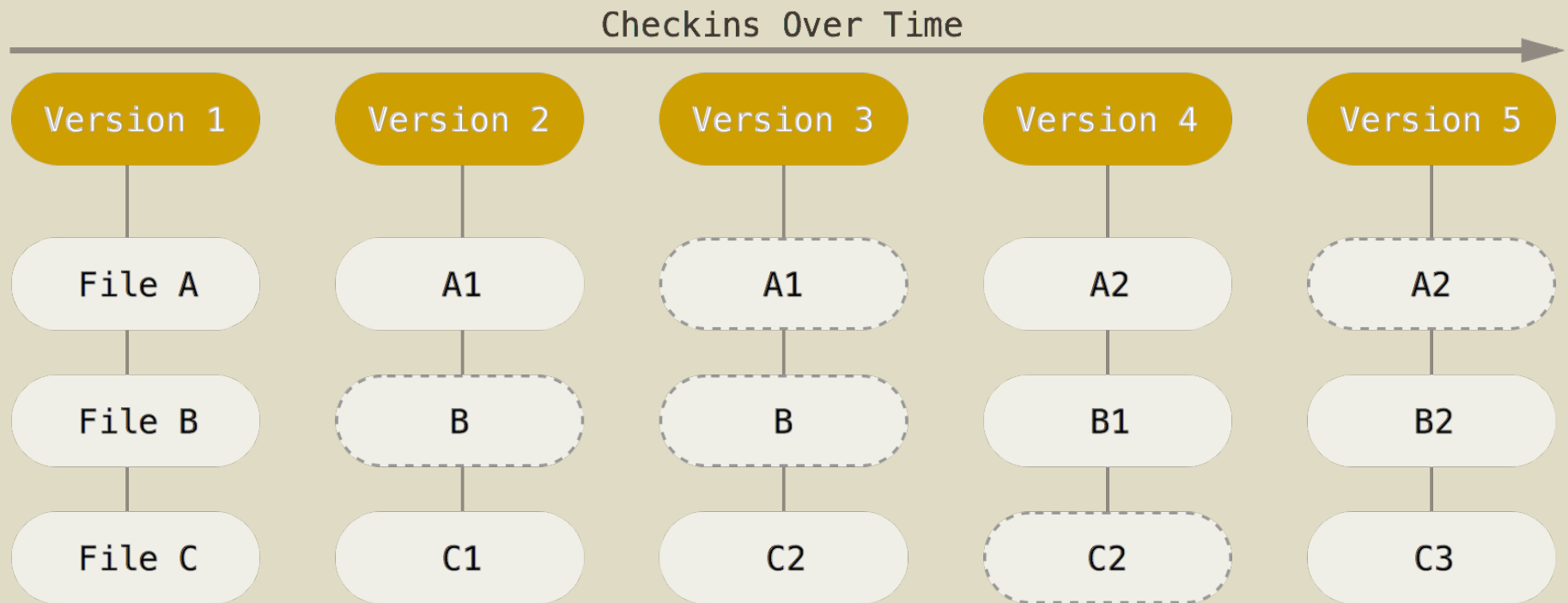
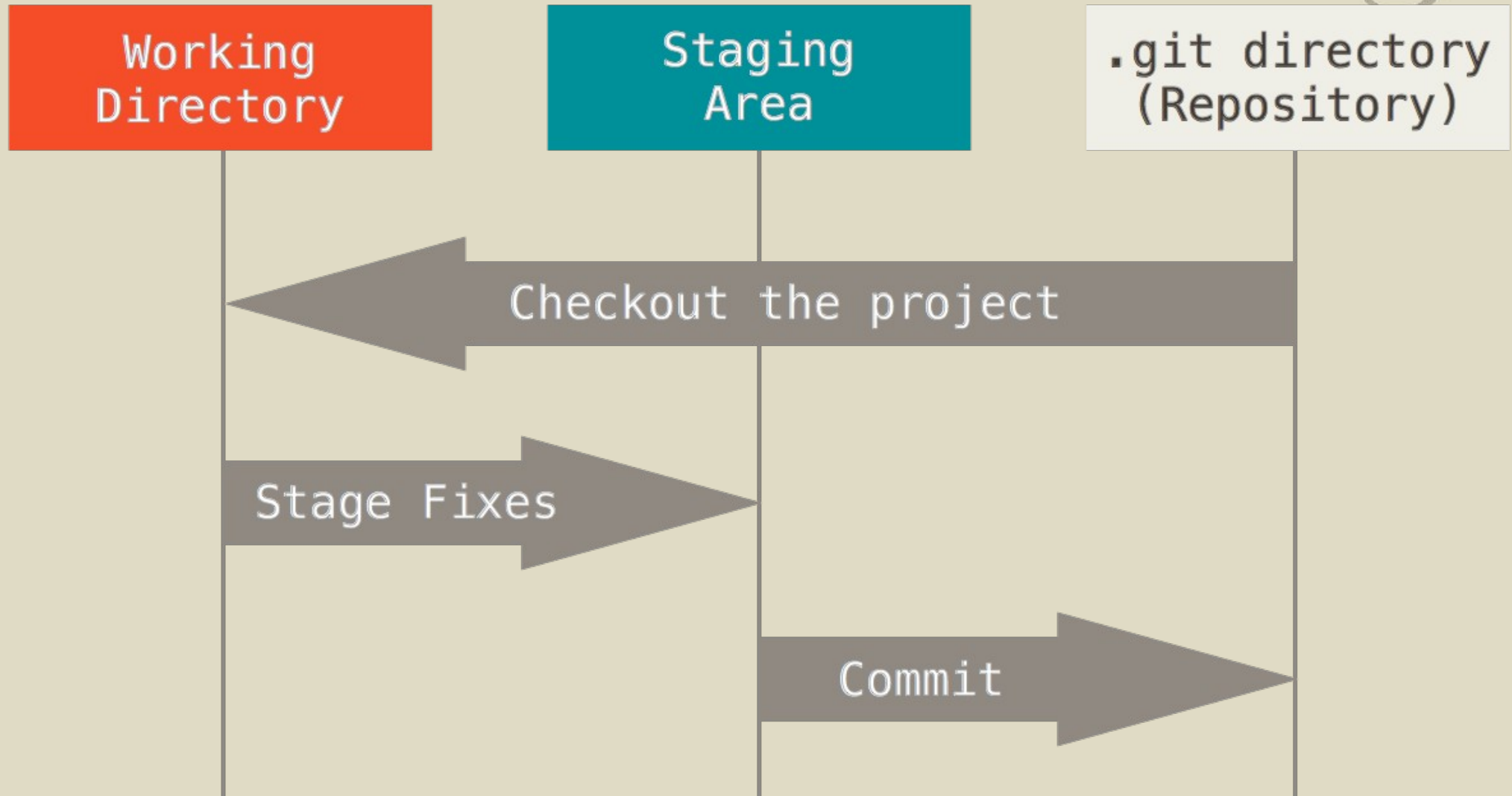


Imagen modificada de "Pro Git" <http://git-scm.com/book/en/v2/>, escrito por Scott Chacon y Ben Straub.
Licenciada bajo Creative Commons Attribution Non Commercial Share Alike 3.0 license.

Versiones en git



Areas en git



Vayamos a la práctica

- Verificar si git está instalado

```
$ git --version
```

- Si no, instalarlo

```
$ sudo apt-get install git
```

- Configurar lo básico y verificarlo

```
$ git config --global user.name "José"  
$ git config --global user.email jose@example.com
```

```
$ git config --list
```

Crear un nuevo repositorio



- Crear un directorio

```
$ mkdir pruebas
```

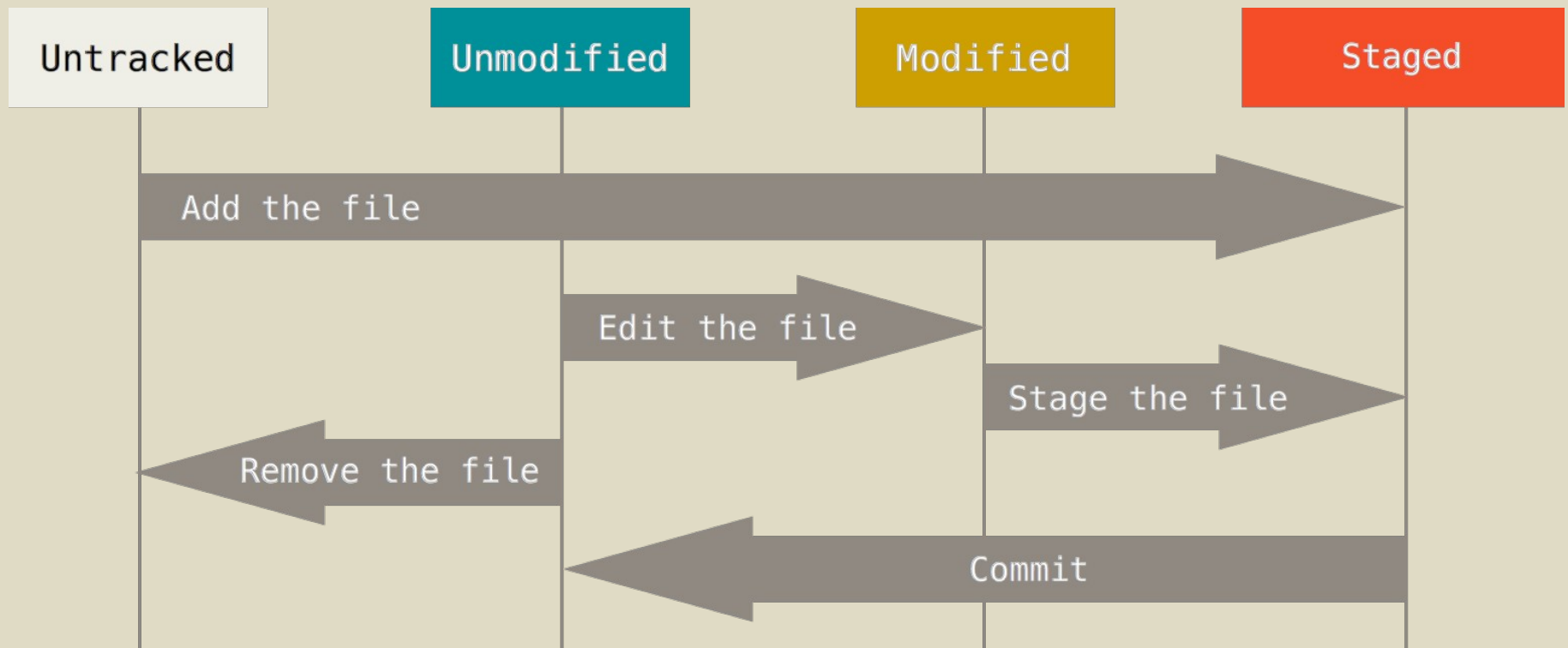
- Acceder al mismo

```
$ cd pruebas
```

- Inicializar el repositorio

```
$ git init
```

Ciclo de vida de un archivo



Trabajando con archivos

- Verificar el estado del repositorio

```
$ git status
On branch master

Initial commit

nothing to commit (create/copy files and use "git add"
to track)
```

- Crear un archivo

```
$ gedit archivo1.txt
```

```
Hola
```


Trabajando con archivos

- Verificar el estado del repositorio

```
$ git status  
On branch master
```

```
Initial commit
```

Untracked files:

(use "git add <file>..." to include in what will be committed)

archivo1.txt

nothing added to commit but untracked files present
(use "git add" to track)

Trabajando con archivos

- Agregar el archivo y verificar el estado del repositorio

```
$ git add archivo1.txt
```

```
$ git status  
On branch master
```

```
Initial commit
```

```
Changes to be committed:
```

```
(use "git rm --cached <file>..." to unstage)
```

```
new file:    archivo1.txt
```

Trabajando con archivos

- Realizar un *commit* para persistir el cambio.

```
$ git commit -m 'El primer commit'  
[master (root-commit) 879f655] El primer commit  
1 file changed, 1 insertion(+)  
create mode 100644 archivo1.txt
```

```
$ git status  
On branch master  
nothing to commit, working directory clean
```

Trabajando con archivos



- Modificar el archivo

```
$ gedit archivo1.txt
```

```
Hola mundo
```

Trabajando con archivos

```
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be
  committed)
  (use "git checkout -- <file>..." to discard
  changes in working directory)

    modified:   archivo1.txt

Untracked files:
  (use "git add <file>..." to include in what will be
  committed)

  archivo1.txt~

no changes added to commit (use "git add" and/or "git
commit -a")
```

Trabajando con archivos

```
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be
  committed)
  (use "git checkout -- <file>..." to discard changes
  in working directory)
```

```
    modified:   archivo1.txt
```

```
Untracked files:
```

```
  (use "git add <file>..." to include in what will
  be committed)
```

```
    archivo1.txt~
```

```
no changes added to commit (use "git add" and/or "git
commit -a")
```

Trabajando con archivos

- Eliminemos el archivo secundario

```
$ rm archivo1.txt~
```

```
$ git status
```

```
On branch master
```

```
Changes not staged for commit:
```

```
  (use "git add <file>..." to update what will be  
committed)
```

```
  (use "git checkout -- <file>..." to discard  
changes in working directory)
```

```
    modified:   archivo1.txt
```

```
no changes added to commit (use "git add" and/or "git  
commit -a")
```

Trabajando con archivos

- Descartemos los cambios

```
$ git checkout -- archivo1.txt
```

```
$ git status  
On branch master  
nothing to commit, working directory clean
```


Trabajando con archivos

- Modifiquemos nuevamente el archivo

```
$ gedit archivo1.txt
```

```
Adios
```

```
$ rm archivo1.txt~
```

```
$ git status
```

```
On branch master
```

```
Changes not staged for commit:
```

```
(use "git add <file>..." to update what will be  
committed)
```

```
(use "git checkout -- <file>..." to discard  
changes in working directory)
```

```
modified:    archivo1.txt
```

```
no changes added to commit (use "git add" and/or "git  
commit -a")
```

Trabajando con archivos

```
$ git add archivo1.txt
```

```
$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   archivo1.txt
```

```
$ git commit -m 'Modificamos archivo1.txt'
[master dad7ae1] Modificamos archivo1.txt
1 file changed, 1 insertion(+), 1 deletion(-)
```

Historial de cambios

```
$ git log  
commit dad7ae163558d32b90207f5081dfa839111c050a  
Author: Andrés Fortier <andres@creativa77.com>  
Date: Thu Mar 19 21:43:41 2015 -0300
```

Modificamos archivo1.txt

```
commit 879f655705648182914372b5de9566c2783b1047  
Author: Andrés Fortier <andres@creativa77.com>  
Date: Thu Mar 19 21:25:52 2015 -0300
```

El primer commit

Historial de cambios

```
$ git diff 879f655..dad7ae1
diff --git a/archivo1.txt b/archivo1.txt
index a19abfe..3ec006e 100644
--- a/archivo1.txt
+++ b/archivo1.txt
@@ -1,1 @@
-Hola
+Adios
```

Trabajando con archivos

- Algunos cambios mas

```
$ gedit archivo1.txt
```

```
Adios  
Amigos
```

```
$ rm archivo1.txt~
```

```
$ git status  
On branch master  
Changes not staged for commit:  
  (use "git add <file>..." to update what will be  
committed)  
  (use "git checkout -- <file>..." to discard changes  
in working directory)  
  
    modified:   archivo1.txt  
  
no changes added to commit (use "git add" and/or "git  
commit -a")
```

Trabajando con archivos

```
$ git add archivo1.txt
```

```
$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   archivo1.txt
```

```
$ git reset HEAD archivo1.txt
Unstaged changes after reset:
M   archivo1.txt
```

Trabajando con archivos

- Vuelvo al mismo estado anterior a hacer el *add*

```
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be
  committed)
  (use "git checkout -- <file>..." to discard changes
  in working directory)

    modified:   archivo1.txt

no changes added to commit (use "git add" and/or "git
commit -a")1
```

Trabajando con archivos

- Continuemos

```
$ git add archivo1.txt
```

```
$ git commit -m 'Agregamos linea en archivo'  
[master 0839195] Agregamos linea en archivo  
1 file changed, 1 insertion(+)
```

```
$ git status  
On branch master  
nothing to commit, working directory clean
```


Historial de cambios

```
$ git log
commit 0839195770a487adeace0b8214ce43762d8fbbb4
Author: Andrés Fortier <andres@creativa77.com>
Date: Thu Mar 19 21:58:41 2015 -0300
```

Agregamos linea en archivo

```
commit dad7ae163558d32b90207f5081dfa839111c050a
Author: Andrés Fortier <andres@creativa77.com>
Date: Thu Mar 19 21:43:41 2015 -0300
```

Modificamos archivo1.txt

```
commit 879f655705648182914372b5de9566c2783b1047
Author: Andrés Fortier <andres@creativa77.com>
Date: Thu Mar 19 21:25:52 2015 -0300
```

El primer commit

Historial de cambios

```
$ git diff dad7ae1..0839195
diff --git a/archivo1.txt b/archivo1.txt
index 3ec006e..6ed6d9b 100644
--- a/archivo1.txt
+++ b/archivo1.txt
@@ -1 +1,2 @@
  Adios
+Amigos
```

```
$ git diff 879f655..0839195
diff --git a/archivo1.txt b/archivo1.txt
index a19abfe..6ed6d9b 100644
--- a/archivo1.txt
+++ b/archivo1.txt
@@ -1 +1,2 @@
-Hola
+Adios
+Amigos
```

Trabajando con archivos

- Uso del “.”

```
$ touch archivo2.txt
```

```
$ touch archivo3.txt
```

```
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be
committed)
```

```
    archivo2.txt
```

```
    archivo3.txt
```

```
nothing added to commit but untracked files present
(use "git add" to track)
```

Trabajando con archivos




```
$ git add .
```

```
$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   archivo2.txt
    new file:   archivo3.txt
```

Ruby gems



- Gem == Librería o plugin.
- Gran ecosistema (casi 100.000 gems).
- <https://rubygems.org/> - Servicio de hosting (+búsqueda) de gemas.
- Instalar una gema

```
$ gem install <nombre>
```

rvm gemsets

- Nos permite mantener diversos conjuntos de gemas.
- Muy útil para trabajar con diversos proyectos en Ruby.
- <https://rvm.io/gemsets>

Bundler

- Ayuda a manejar (y distribuir) las dependencias de un proyecto.
- Declara las dependencias en un archivo Gemfile

```
source 'https://rubygems.org'

gem 'nokogiri'
gem 'rails', '3.0.0.beta3'
gem 'rack', '>=1.0'
gem 'thin', '~>1.1'
```

Contador



- La clase pasada definimos la clase contador. Pero no la testeamos!
- En ruby hay varias librerías de testing. Nosotros vamos a usar rspec.

Contador

- Crear el directorio “clase03”

```
$ mkdir clase03
```

- Posicionarse en él

```
$ cd clase03
```

- Seleccionar la versión de Ruby y crear un gemset

```
$ rvm use ruby-2.0.0-p598@clase03 --create
ruby-2.0.0-p598 - #gemset created
/home/andres/.rvm/gems/ruby-2.0.0-p598@clase03
ruby-2.0.0-p598 - #generating clase03
wrappers.....
Using /home/andres/.rvm/gems/ruby-2.0.0-p598 with
gemset clase03
```

Contador



- Alternativamente

```
$ rvm use ruby-2.0.0-p598  
  
$ rvm gemset create clase03  
  
$ rvm gemset use clase03
```

Contador

- Instalar rake

```
$ gem install rake  
Fetching: rake-10.4.2.gem (100%)  
Successfully installed rake-10.4.2  
Parsing documentation for rake-10.4.2  
Installing ri documentation for rake-10.4.2  
1 gem installed
```

- Instalar bundler

```
$ gem install bundler  
Fetching: bundler-1.9.1.gem (100%)  
Successfully installed bundler-1.9.1  
Parsing documentation for bundler-1.9.1  
Installing ri documentation for bundler-1.9.1  
1 gem installed
```

Contador

- Crear el archivo `Gemfile`
 - <https://rubygems.org/gems/rspec>

```
source 'https://rubygems.org'  
  
gem 'rspec', '~> 3.2.0'
```

Contador

- Indicarle a bundler que satisfaga las dependencias

```
$ bundle install
Fetching gem metadata from https://rubygems.org/.....
Fetching version metadata from https://rubygems.org/..
Resolving dependencies...
Installing diff-lcs 1.2.5
Installing rspec-support 3.2.2
Installing rspec-core 3.2.2
Installing rspec-expectations 3.2.0
Installing rspec-mocks 3.2.1
Installing rspec 3.2.0
Using bundler 1.9.1
Bundle complete! 1 Gemfile dependency, 7 gems now
installed.
Use `bundle show [gemname]` to see where a bundled gem
is installed.
```

Contador

```
$ cat Gemfile.lock
GEM
remote: https://rubygems.org/
specs:
  diff-lcs (1.2.5)
  rspec (3.2.0)
    rspec-core (~> 3.2.0)
    rspec-expectations (~> 3.2.0)
    rspec-mocks (~> 3.2.0)
  rspec-core (3.2.2)
    rspec-support (~> 3.2.0)
  rspec-expectations (3.2.0)
    diff-lcs (>= 1.2.0, < 2.0)
    rspec-support (~> 3.2.0)
  rspec-mocks (3.2.1)
    diff-lcs (>= 1.2.0, < 2.0)
    rspec-support (~> 3.2.0)
  rspec-support (3.2.2)
```

...

Contador

- Inicializar rspec

```
$ rspec --init  
create    .rspec  
create    spec/spec_helper.rb
```

- Tratemmos de ejecutar los tests

```
$ rspec  
No examples found.  
  
Finished in 0.00027 seconds (files took 0.20727 seconds  
to load)  
0 examples, 0 failures
```

Contador - contador.rb

```
class Contador

  attr_reader :valor

  def initialize(valor_inicial = 0)
    @valor = valor_inicial
  end

  def limpiar
    @valor = 0
  end

  def siguiente
    @valor = @valor + 1
  end

  def vacio?
    return @valor == 0
  end

end
```


Contador - contador_circular.rb

```
class ContadorCircular < Contador

  def siguiente
    super
    if @valor == 3
      @valor = 0
    end
    return @valor
  end

end
```

Contador - contador_circular.rb

- Intentemos abrir una consola irb

```
$ irb -r ./contador_circular.rb
/home/andres/Docencia/UTN/2015/Laboratorio
4/Clase03/clase03/contador_circular.rb:1:in `
```

Contador - contador_circular.rb

```
require_relative 'contador'

class ContadorCircular < Contador

  def siguiente
    super
    if valor == 3
      self.valor = 0
    end
    return valor
  end

end
```

Contador - contador_circular.rb

- Volvamos a irb

```
$ irb -r ./contador_circular.rb
2.0.0-p598 :001 > c = ContadorCircular.new(1)
=> #<ContadorCircular:0x000000025f0eb8 @valor=1>
2.0.0-p598 :002 > c.siguiente
=> 2
2.0.0-p598 :003 > c.siguiente
=> 0
2.0.0-p598 :004 > exit
```

Primer test

spec/contador_spec.rb

```
require_relative '../contador'

describe Contador do

  it "Has a value 0 after creation" do
    expect(Contador.new.valor).to be(0)
  end

end
```

```
$ rspec
```

```
.
```

```
Finished in 0.00083 seconds (files took 0.10982 seconds
to load)
1 example, 0 failures
```

Extendamos el test

```
require_relative '../contador'

describe Contador do

  it "Has a value 0 after creation" do
    expect(Contador.new.valor).to be(0)
  end

  it "Uses the given value on creation" do
    expect(Contador.new(7).valor).to be(7)
  end

end
```

```
$ rspec
```

```
..
```

```
Finished in 0.00123 seconds (files took 0.11436 seconds
to load)
2 examples, 0 failures
```

Un poco de estructura

```
require_relative '../contador'

describe Contador do

  describe "initialize" do

    it "Has a value 0 after creation" do
      expect(Contador.new.valor).to be(0)
    end

    it "Uses the given value on creation" do
      expect(Contador.new(7).valor).to be(7)
    end

  end

end
```

Un poco de estructura

```
$ rspec
```

```
..
```

```
Finished in 0.00123 seconds (files took 0.11436 seconds  
to load)
```

```
2 examples, 0 failures
```


Y formato de salida

```
$ rspec --help
Usage: rspec [options] [files or directories]

-f, --format FORMATTER  Choose a formatter
                        [p]rogress (default - dots)
                        [d]ocumentation (group and example names)
                        [h]tml
                        [t]extmate
                        custom formatter class name
```

```
$ rspec -fd
```

Contador

initialize

Has a value 0 after creation

Uses the given value on creation


```
Finished in 0.00121 seconds (files took 0.11323 seconds to load)
2 examples, 0 failures
```

Git - Links



- <http://git-scm.com/book/en/v2/>
<http://git-scm.com/book/es/v2/>
- <http://rogerdudler.github.io/git-guide/index.html>
<http://rogerdudler.github.io/git-guide/index.es.html>
- <http://gitimmersion.com/>
- <https://try.github.io/>

rspec - Links



- <http://blog.davidchelimsky.net/blog/2007/05/14/an-introduction-to-rspec-part-i/>
- <http://code.tutsplus.com/tutorials/ruby-for-newbies-testing-with-rspec--net-21297>
- <https://relishapp.com/rspec/docs/gettingstarted>

Tarea para el hogar



- Crear un nuevo repositorio git para el contador y para los ejercicios que resolvieron.
- Hacer los tests para los contadores y para los ejercicios.
- Cuando ejerciten git
 - Hagan *commits* periódicos y vean los *diffs*.
 - Pierdan el miedo a mover un archivo entre el *working directory* y *staging area*.