

Controle de Versão pra Quê?

Leslie Harlley Watter

September 19, 2014

Outline

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

Necessidade

Você já passou pela situação em que disse:

1. "Cadê a última versão desse arquivo?"
2. "Eu tenho certeza de que já tinha escrito isso no documento ..."
3. "Quando foi que foi retirada aquela cláusula do contrato?"

Se disse sim para alguma das perguntas anteriores, você provavelmente precisa de um controle de versão ;-)

Controle de Versão, o que é ?

Um sistema de controle de versão (ou versionamento), VCS (do inglês version control system) ou ainda SCM (do inglês source code management) na função prática da Ciência da Computação e da Engenharia de Software, é um software com a finalidade de gerenciar diferentes versões no desenvolvimento de um documento qualquer.

Esses sistemas são comumente utilizados no desenvolvimento de software para controlar as diferentes versões — histórico e desenvolvimento — dos códigos-fontes e também da documentação.

Fonte: Wikipedia - <http://tinyurl.com/278k3vj>

Onde usar controle de versão?

- ▶ Desenvolvimento de Código
- ▶ Manutenção de Configurações
- ▶ Praticamente qualquer coisa em que se precise controlar as modificações executadas
- ▶ Atividades que exigem colaboração entre diferentes indivíduos.

Por que usar um controle de versão?

- ▶ Colaboração
- ▶ Armazenamento de Versões Adequado
- ▶ Restaurar Versões Anteriores
- ▶ Compreender O Que Aconteceu
- ▶ Backup

Basicamente porquê...



SHIT HAPPENS

(shut up and deal with it)

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

Controle Manual

- ▶ O que normalmente fazemos?
 - ▶ Arquivos de backup com data e hora
 - ▶ acrescentamos um número ou nome ao final do arquivo. . .

Controle Manual

- ▶ O que normalmente fazemos?
 - ▶ Arquivos de backup com data e hora
 - ▶ acrescentamos um número ou nome ao final do arquivo...
 - ▶ contrato.txt (nome original)

Controle Manual

- ▶ O que normalmente fazemos?
 - ▶ Arquivos de backup com data e hora
 - ▶ acrescentamos um número ou nome ao final do arquivo...
 - ▶ contrato.txt (nome original)
 - ▶ contrato-v1.txt
 - ▶ contrato-v2.txt

Controle Manual

- ▶ O que normalmente fazemos?
 - ▶ Arquivos de backup com data e hora
 - ▶ acrescentamos um número ou nome ao final do arquivo...
 - ▶ contrato.txt (nome original)
 - ▶ contrato-v1.txt
 - ▶ contrato-v2.txt
 - ▶ contrato-final-leslie.txt
 - ▶ contrato-final-marcos.txt

Controle Manual

- ▶ O que normalmente fazemos?
 - ▶ Arquivos de backup com data e hora
 - ▶ acrescentamos um número ou nome ao final do arquivo...
 - ▶ contrato.txt (nome original)
 - ▶ contrato-v1.txt
 - ▶ contrato-v2.txt
 - ▶ contrato-final-leslie.txt
 - ▶ contrato-final-marcos.txt
 - ▶ contrato-final-revisado.txt

Controle Manual

- ▶ O que normalmente fazemos?
 - ▶ Arquivos de backup com data e hora
 - ▶ acrescentamos um número ou nome ao final do arquivo...
 - ▶ contrato.txt (nome original)
 - ▶ contrato-v1.txt
 - ▶ contrato-v2.txt
 - ▶ contrato-final-leslie.txt
 - ▶ contrato-final-marcos.txt
 - ▶ contrato-final-revisado.txt
 - ▶ contrato-final-mesmo.txt

Controle Manual

- ▶ O que normalmente fazemos?
 - ▶ Arquivos de backup com data e hora
 - ▶ acrescentamos um número ou nome ao final do arquivo...
 - ▶ contrato.txt (nome original)
 - ▶ contrato-v1.txt
 - ▶ contrato-v2.txt
 - ▶ contrato-final-leslie.txt
 - ▶ contrato-final-marcos.txt
 - ▶ contrato-final-revisado.txt
 - ▶ contrato-final-mesmo.txt
- ▶ e como saber exatamente o que mudou de uma versão para outra ?

E como descobrir as diferenças entre esses arquivos?

- ▶ Comparando Visualmente lado a lado arquivo por arquivo

E como descobrir as diferenças entre esses arquivos?

- ▶ Comparando Visualmente lado a lado arquivo por arquivo
 - ▶ pode levar uma "eternidade" e é muito sujeito a erros

E como descobrir as diferenças entre esses arquivos?

- ▶ Comparando Visualmente lado a lado arquivo por arquivo
 - ▶ pode levar uma "eternidade" e é muito sujeito a erros
- ▶ Marcando em diferentes cores as modificações (dentro do *office)

E como descobrir as diferenças entre esses arquivos?

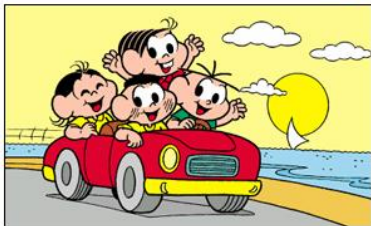
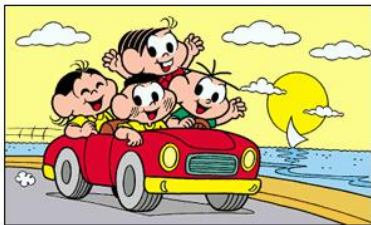
- ▶ Comparando Visualmente lado a lado arquivo por arquivo
 - ▶ pode levar uma "eternidade" e é muito sujeito a erros
- ▶ Marcando em diferentes cores as modificações (dentro do *office)
 - ▶ vai chegar uma hora em que o texto inteiro estará em amarelo, vermelho, verde, e irão faltar cores, dependendo do número de pessoas editando.

E como descobrir as diferenças entre esses arquivos?

- ▶ Comparando Visualmente lado a lado arquivo por arquivo
 - ▶ pode levar uma "eternidade" e é muito sujeito a erros
- ▶ Marcando em diferentes cores as modificações (dentro do *office)
 - ▶ vai chegar uma hora em que o texto inteiro estará em amarelo, vermelho, verde, e irão faltar cores, dependendo do número de pessoas editando.
- ▶ Acaba virando um ...

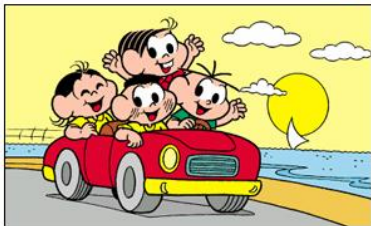
Jogo dos 7 erros

7 erros pequenos são mais fáceis de serem corrigidos que 7 erros grandes, embora seja muito mais difícil de encontrá-los no meio de muita informação.



Jogo dos 7 erros

7 erros pequenos são mais fáceis de serem corrigidos que 7 erros grandes, embora seja muito mais difícil de encontrá-los no meio de muita informação.



- ▶ A solução está em usar uma ferramenta para facilitar esse trabalho!

diff

- ▶ O diff é uma ferramenta que faz a comparação de dois arquivos e exibe as diferenças entre esses arquivos.
- ▶ Possui inúmeros formatos e opções ...

Aplicação do Diff

Arquivo Original

```
lista de compras  
maçã  
banana  
melão
```

Arquivo Modificado

```
lista de compras  
maçã fuji  
banana  
melancia
```

Diferenças

```
/tmp$ diff -u orig.txt modificado.txt
```

```
--- orig.txt      2014-09-18 14:39:46.208575472 -0300  
+++ modificado.txt      2014-09-18 14:40:29.779574747 -0300  
@@ -1,4 +1,4 @@  
   lista de compras  
-maçã  
+maçã fuji  
   banana  
-melão  
+melancia
```

Diff

```
diff --git a/net/socket.c b/net/socket.c
index 791d71a..6d47165 100644
--- a/net/socket.c
+++ b/net/socket.c
@@ -736,7 +736,7 @@ static ssize_t sock_sendpage(struct file *file, struct page *page,
     if (more)
         flags |= MSG_MORE;

-    return sock->ops->sendpage(sock, page, offset, size, flags);
+    return kernel_sendpage(sock, page, offset, size, flags);
 }

 static ssize_t sock_splice_read(struct file *file, loff_t *ppos,
```

Figure : Exemplo de Diff usando o Git

patch

- ▶ O patch é uma ferramenta que aplica as diferenças (patch) entre dois arquivos no arquivo original, de forma a obter-se o arquivo modificado. Confuso não ?



patch

- ▶ O patch é uma ferramenta que aplica as diferenças (patch) entre dois arquivos no arquivo original, de forma a obter-se o arquivo modificado. Confuso não ?



- ▶ Fica mais fácil mostrando o que ele faz ...

Criação do Patch

Arquivo Original

```
lista de compras  
maçã  
banana  
melão
```

Arquivo Modificado

```
lista de compras  
maçã fuji  
banana  
melancia
```

Gerando o Patch

```
$ diff -u orig.txt modificado.txt > modifica.patch
```

Patch Gerado

modifica.patch (exatamente a saída do diff)

```
/tmp$ cat modifica.patch
--- orig.txt      2014-09-18 14:39:46.208575472 -0300
+++ modificado.txt      2014-09-18 14:40:29.779574747 -0300
@@ -1,4 +1,4 @@
     lista de compras
 -maçã
 +maçã fuji
     banana
 -melão
 +melancia
```

Aplicação de um patch

Aplicando o Patch

```
/tmp$ patch -p0 < modifica.patch  
patching file orig.txt  
/tmp$
```

O Arquivo Original foi modificado pelo Patch

```
/tmp$ cat orig.txt  
lista de compras  
maçã fuji  
banana  
melancia
```

Então ...

- ▶ Usar patches pode ser uma maneira de compartilhar as alterações entre diferentes pessoas, porém

Então ...

- ▶ Usar patches pode ser uma maneira de compartilhar as alterações entre diferentes pessoas, porém
- ▶ conviver com diffs e patches pode ser **bem** complicado quando a quantidade aumenta.

Então ...

- ▶ Usar patches pode ser uma maneira de compartilhar as alterações entre diferentes pessoas, porém
- ▶ conviver com diffs e patches pode ser **bem** complicado quando a quantidade aumenta.
- ▶ para automatizar essa tarefa é que existem os **sistemas de controle de versão** ;-)

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

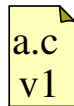
Conceitos

- ▶ Gerenciamento de Controle de Versões
 - ▶ registra as modificações em arquivos
 - ▶ repositório / banco de dados de mudanças
 - ▶ diretório de trabalho / estado corrente
- ▶ SCM Centralizado
 - ▶ servidor: banco de dados único
 - ▶ cliente: diretório de trabalho & estado
- ▶ SCM Descentralizado
 - ▶ qualquer um pode ser um servidor
 - ▶ repositório associado ao diretório de trabalho
 - ▶ histórico completo
 - ▶ operação desconectada

Componentes de um SCM

Conteúdo do Repositório

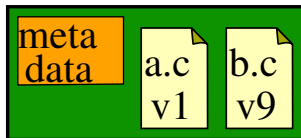
- ▶ arquivos



Componentes de um SCM

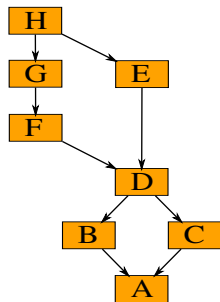
Conteúdo do Repositório

- ▶ arquivos
- ▶ commits



Componentes de um SCM

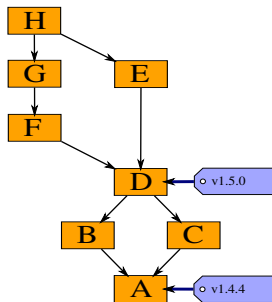
Grafo Acíclico Dirigido “DAG”



Componentes de um SCM

Referências

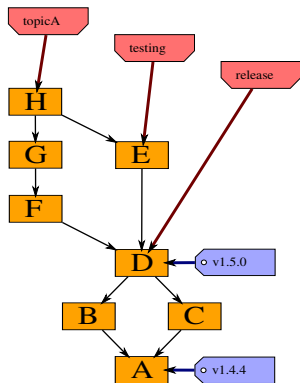
- ▶ tags



Componentes de um SCM

Referências

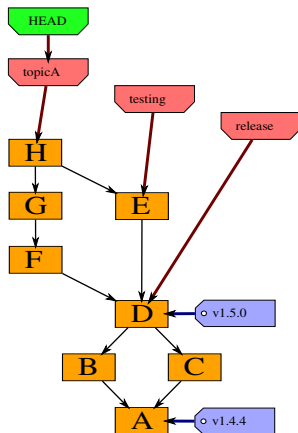
- ▶ tags
- ▶ branches (ramos)



Componentes de um SCM

HEAD

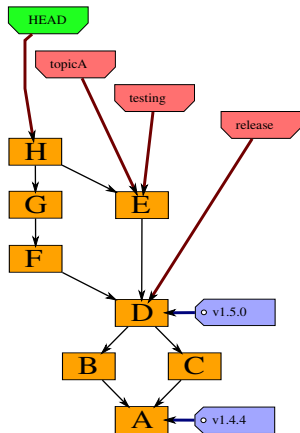
- ▶ checkout atual
- ▶ aponta para um branch



Componentes de um SCM

HEAD

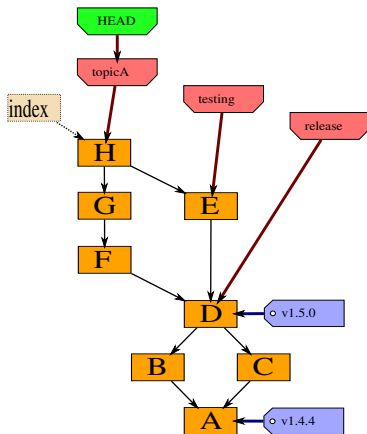
- ▶ checkout atual
- ▶ aponta para um branch
- ▶ pode estar “desconectado”



Componentes de um SCM

Index

- ▶ “staging area” – área intermediária
- ▶ o que está para ser comitado

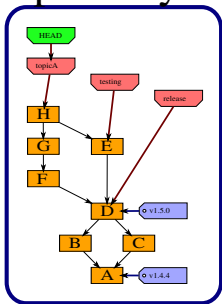


Operações em um SCM

- ▶ Inicialização
 - ▶ init
 - ▶ checkout
- ▶ Modificações
 - ▶ add, delete, rename
 - ▶ commit
- ▶ Informações
 - ▶ status
 - ▶ diff
 - ▶ log
- ▶ Referência
 - ▶ tag
 - ▶ branch

Estrutura

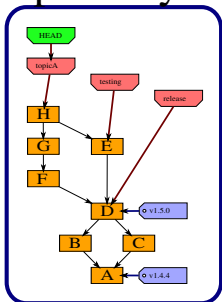
repository



mantém o histórico

Estrutura

repository



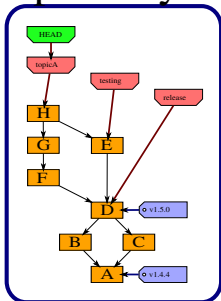
index

```
100644 20b024 0 bar
100644 1d52a6 0 baz
100644 20b024 0 sub/fiz
100644 43dbe0 0 sub/foo
```

staging area – área intermediária

Estrutura

repository



index

```
100644 20b024 0 bar
100644 1d52a6 0 baz
100644 20b024 0 sub/fiz
100644 43dbe0 0 sub/foo
```

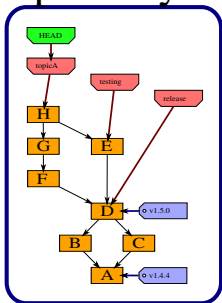
work tree

```
.
|-- bar
|-- baz
'-- sub
    |-- fiz
    '--- foo
```

árvore de trabalho – arquivos que você modifica

Estrutura

repository



index

```
100644 20b024 0 bar
100644 1d52a6 0 baz
100644 20b024 0 sub/fiz
100644 43dbe0 0 sub/foo
```

work tree

```
.
|-- bar
|-- baz
'-- sub
    |-- fiz
    '-- foo
```

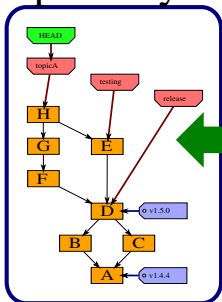


“staging” – informando a modificação

add
remove
rename

Estrutura

repository



index

```
100644 20b024 0 bar
100644 1d52a6 0 baz
100644 20b024 0 sub/fiz
100644 43dbe0 0 sub/foo
```

work tree

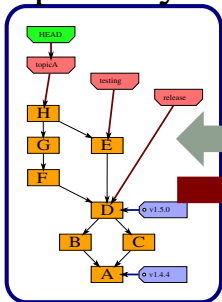
```
.
|-- bar
|-- baz
'-- sub
    |-- fiz
    '-- foo
```

“committing” – efetivando a modificação no repositório

commit

Estrutura

repository



index

```
100644 20b024 0 bar
100644 1d52a6 0 baz
100644 20b024 0 sub/fiz
100644 43dbe0 0 sub/foo
```

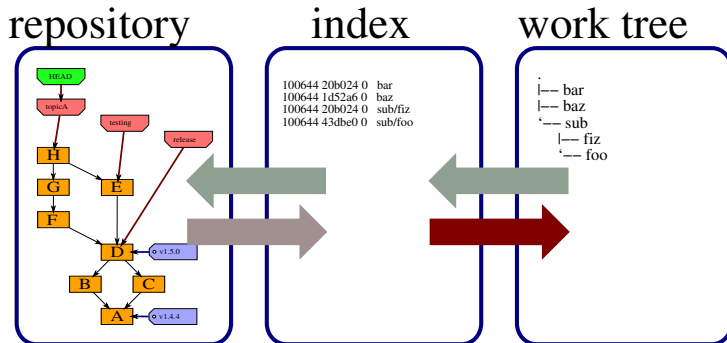
work tree

```
.
|-- bar
|-- baz
'-- sub
    |-- fiz
    '--- foo
```

“reading tree” – lendo a árvore do repositório

checkout
read-tree
reset

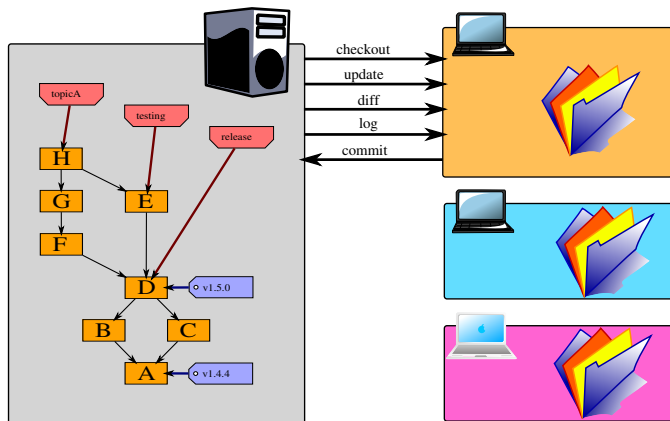
Estrutura



“checking out” – efetivando a recuperação dos arquivos

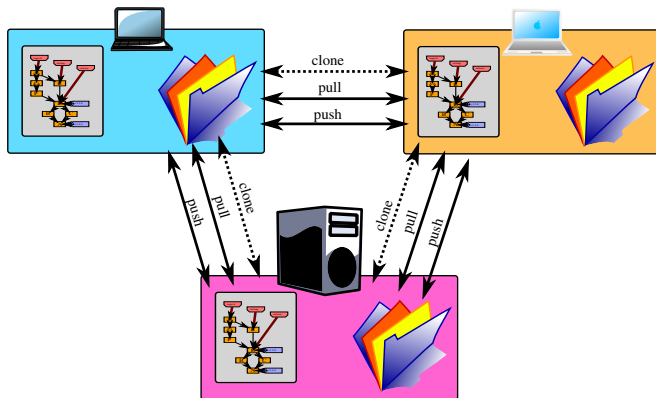
checkout
checkout-index
reset

SCM Centralizado



- ▶ As operações **precisam** de um **servidor**
 - ▶ um único ponto de falha
 - ▶ ponto de gargalo

SCM Descentralizado



- ▶ qualquer um pode ser um servidor
- ▶ operações adicionais: clone, pull, fetch, push

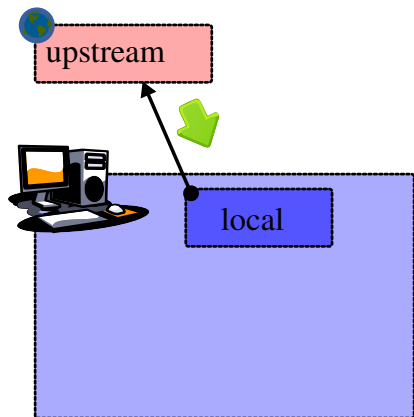
Descentralização



upstream

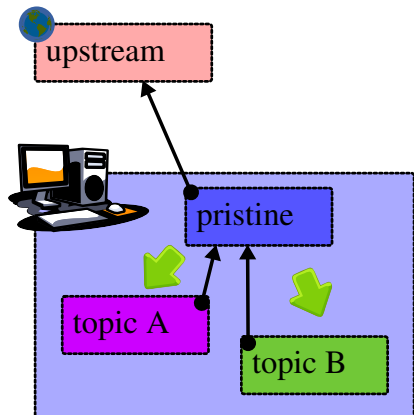
- ▶ repositório público

Descentralização



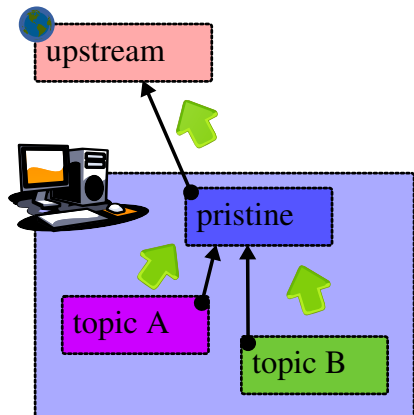
- ▶ criando um clone local
 - ▶ http, ssh, git
 - ▶ `git clone git://git.kernel.org/.../linux-2.6.git linux-2.6`

Descentralização



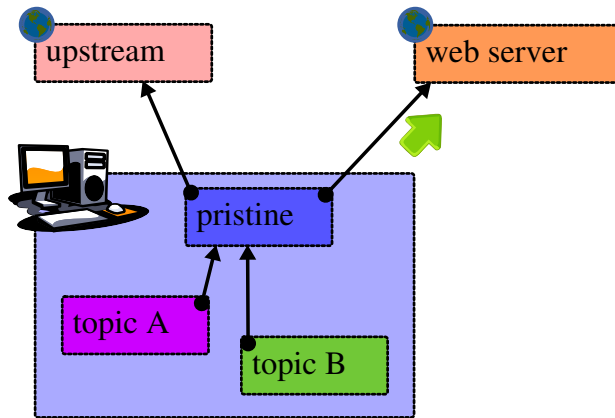
- ▶ clonar localmente é extremamente simples
- ▶ `git clone /scm/linux-2.6 /tmp/linux-2.6-tmp`

Descentralização



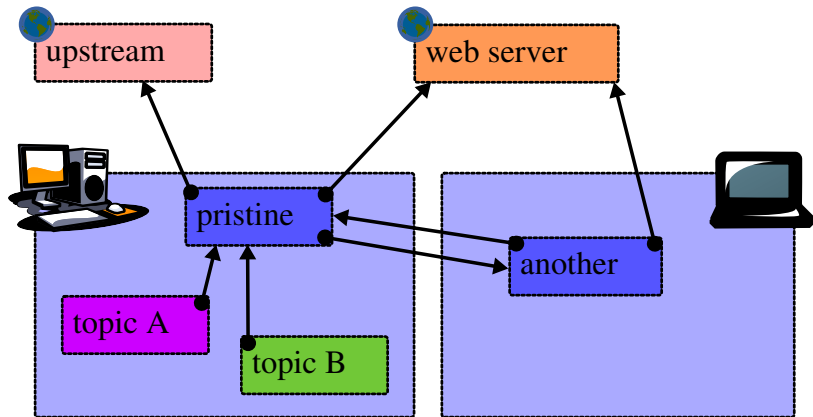
- ▶ envio (push) de modificações entre quaisquer repositórios
- ▶ `git push origin master` (dentro do repositório)

Descentralização



- publica-se mudanças em um servidor público

Descentralização



- ▶ compartilhar mudanças com máquinas confiáveis
- ▶ `git clone ssh://192.168.1.2/home/git/emacs.git emacs`

Por quê descentralizar?

- ▶ micro-commits não intrusivos
- ▶ operação desconectada
- ▶ não há um único ponto de falha
- ▶ backups são triviais

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

Sistemas de Controle de Versão

Centralizados

- ▶ RCS
- ▶ CVS
- ▶ SVN

Distribuídos

- ▶ GIT
- ▶ Mercurial
- ▶ Bazaar
- ▶ Darcs

RCS

Autor	Walter F. Tichy
Desenvolvedor	Projeto GNU
Plataforma	Sistema operacional tipo Unix
Lançamento	1982 (31–32 anos)
Versão estável	5.9.0 (6 de maio de 2013; há 71 semanas e 4 dias)
Escrito em	C
Gênero(s)	Sistema de controle de versão
Licença	GNU General Public License
Página oficial	www.gnu.org/s/rcs/

O RCS funciona apenas em arquivos individuais. Não há maneira de trabalhar com um projeto inteiro.

Fonte: wikipédia


GNU Operating System
Sponsored by the [Free Software Foundation](#)

[JOIN THE FSF](#)
Free Software Supporter
 email address [Sign up](#)

BREAKING: [Knocking Down The HACIENDA](#)


GNU hackers opened the [GHM](#) by revealing the offensive **HACIENDA** global surveillance program for [TWD](#), and **how to [knock it down with stealth TCP services!](#)** [Watch it now!](#) [\[more\]](#)

[About GNU](#)
[Philosophy](#)
[Licenses](#)
[Education](#)
[Software](#)
[Documentation](#)
[Help GNU](#)

GNU RCS

The Revision Control System (RCS) manages multiple revisions of files. RCS automates the storing, retrieval, logging, identification, and merging of revisions. RCS is useful for text that is revised frequently, including source code, programs, documentation, graphics, papers, and form letters.

RCS was first developed by Walter F. Tichy at Purdue University in the early 1980s -- paper: [RCS: A System for Version Control \(1991\)](#) ([troff](#), [PostScript](#), [PDF](#)). See also the [Purdue RCS Homepage](#).

RCS design is an improvement from its predecessor Source Code Control System (SCCS) (see [GNU CSSC](#)). The improvements include an easier user interface and improved storage of versions for **faster retrieval**. **RCS improves performance** by storing an entire copy of the most recent version and then stores reverse differences (called **deltas**). RCS uses GNU Diffutils to **40 %** differences between versions.

Figure : RCS

CVS

Desenvolvedores	The CVS Team
Lançamento	November 19, 1990; 23 anos atrás
versão estável	1.11.23 / May 8, 2008; 6 anos atrás
Versão em desenv	1.12.13 / July 26, 2006; 8 anos atrás
Escrito em	C
Sistema Operacional	Unix-like, Windows
Gênero(s)	Revision control
Licença	GNU General Public License
Página oficial	savannah.nongnu.org/projects/cvs

CVS utiliza uma arquitetura cliente-servidor: um servidor armazena a(s) versão(ões) atuais do projeto e seu histórico, e os clientes se conectam a esse servidor para obter uma cópia completa do projeto, trabalhar nessa cópia e então devolver suas modificações.

Fonte: wikipédia

CVS

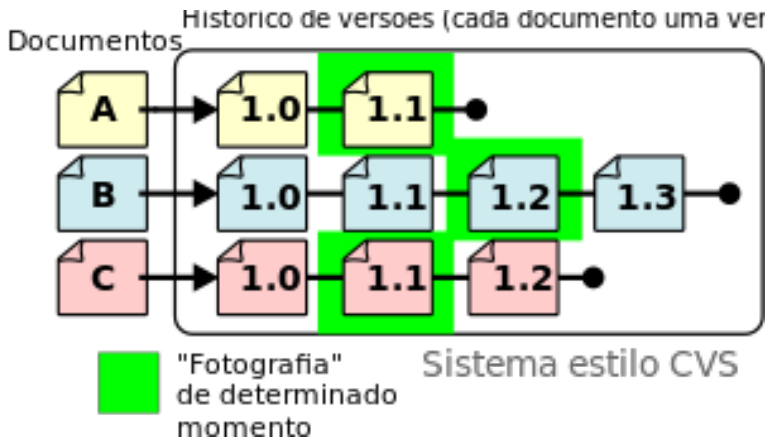


Figure : Versionamento Estilo CVS

Apache Subversion

Desenvolvedores	Apache Software Foundation
Primeira Versão	20 October 2000; 13 anos atrás
Versão estável	1.8.10 (Agosto 11, 2014; 38 dias atrás)
Versão em desenv	1.9.0-alpha2 (Abril 14, 2014; 5 meses atrás)
Estado de Desenvolvimento	Active
Escrito em	C
Sistema Operacional	Cross-platform
Gênero(s)	Revision control
Licença	Apache License
Página oficial	subversion.apache.org

Apache Subversion (também conhecido por svn) é um sistema de controle de versão desenhado especificamente para ser um substituto moderno do CVS, que se considera ter algumas limitações.

Fonte: wikipédia

SVN

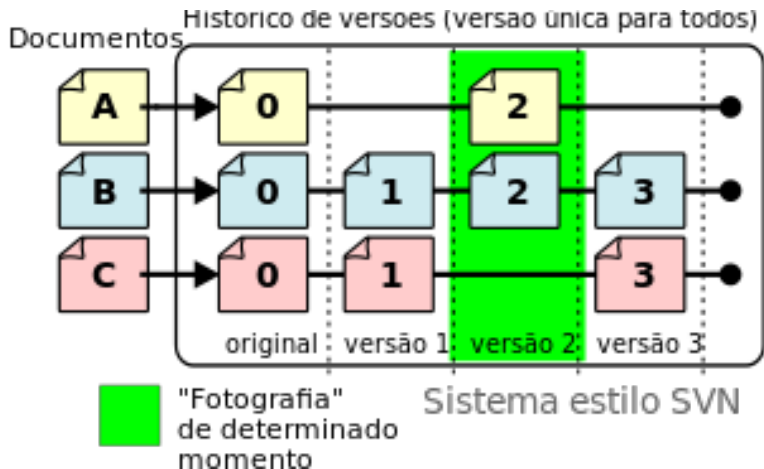


Figure : Versionamento Estilo SVN

GIT

Autor Original	Linus Torvalds
Desenvolvedores	Junio Hamano, Linus Torvalds, e muitos outros
Versão inicial	7 April 2005; 9 anos atrás
Release Estável	2.1.0 (15 Agosto de 2014; 34 dias atrás)
Escrito em	C, Bourne Shell, Tcl, Perl
Sistema Operacional	Linux, POSIX, Windows, OS X
Gênero	Version control
Licença	GNU General Public License v2
Página oficial	git-scm.com

Git pronunciado é um sistema de controle de versão distribuído e um sistema de gerenciamento de código fonte, com ênfase em velocidade. O Git foi inicialmente projetado e desenvolvido por Linus Torvalds para o desenvolvimento do kernel Linux, mas foi adotado por muitos outros projetos.

Cada diretório de trabalho do Git é um repositório com um histórico completo e habilidade total de acompanhamento das revisões, não dependente de acesso a uma rede ou a um servidor central.

GIT

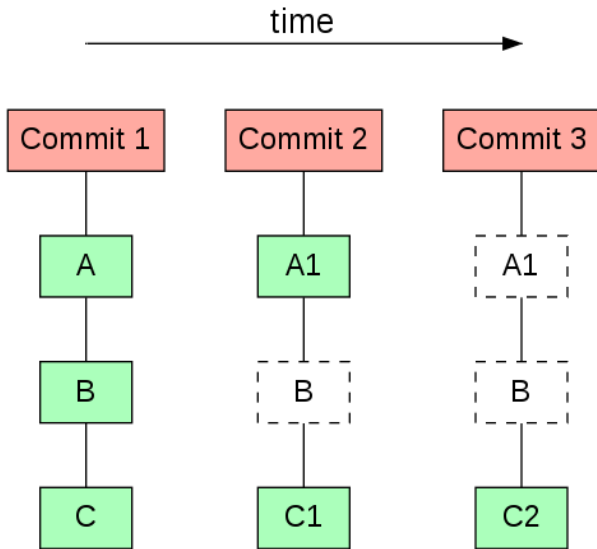


Figure : Commit no GIT

Outras Opções Distribuídas



Figure :

<http://mercurial.selenic.com/>

...



Figure :

<http://bazaar.canonical.com/>



Figure : <http://darcs.net/>

► e a lista continua ...

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

GIT / Commits

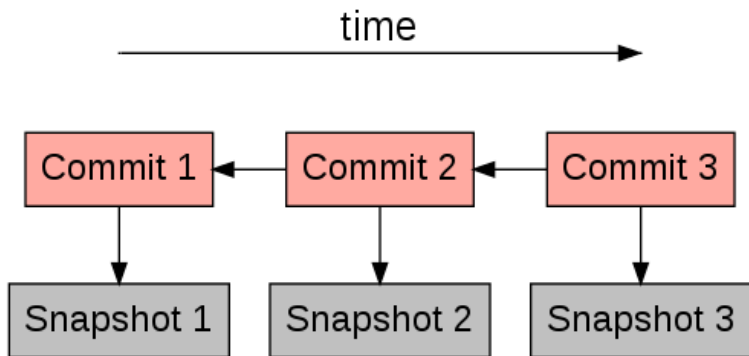


Figure : Commits e seus pais no GIT

GIT - Estrutura de Commit

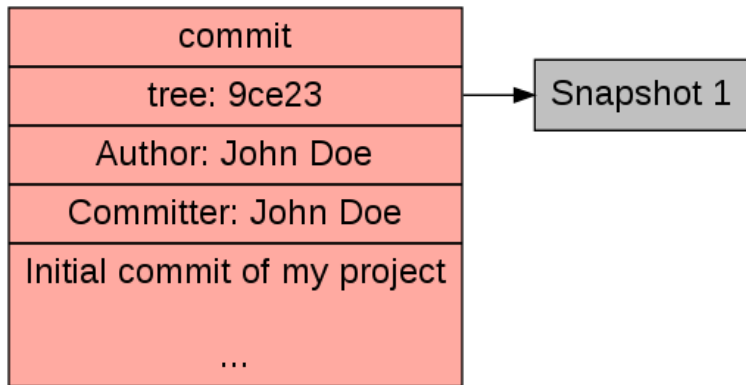


Figure : Estrutura de um Commit

GIT - Estrutura de Branches

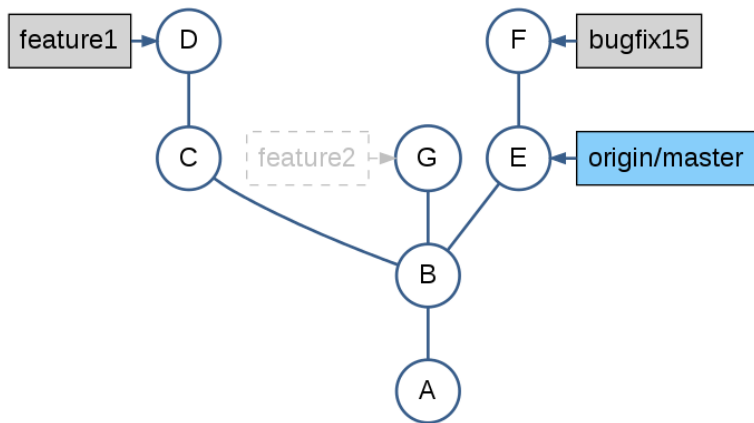


Figure : Estrutura de um Branch

- ▶ começando simples ...

Começando com um repositório

► clone

```
~/git$ git clone https://github.com/watter/moneylog-el.git
Cloning into 'moneylog-el'...
remote: Counting objects: 7, done.
remote: Total 7 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (7/7), done.
Checking connectivity... done.
~/git$ ls
moneylog-el/
~/git$
```

► init

```
~$ mkdir git-exemplo
~$ cd !$
cd git-exemplo
~/git-exemplo$ git init
Initialized empty Git repository in /home/leslie/git-exemplo/.git/
~/git-exemplo$
```

Fazendo alterações

► add

```
~/git-exemplo$ cat lista-compras.txt
```

```
lista de compras
```

```
banana
```

```
laranja
```

```
~/git-exemplo$ git add lista-compras.txt
```

```
~/git-exemplo$
```

Verificando o estado atual

► status

```
~/git-exemplo$ git status  
No ramo master
```

Submissão inicial.

Mudanças a serem submetidas:

(utilize "git rm --cached <arquivo>..." para não apresentar)

```
new file:   lista-compras.txt
```

```
~/git-exemplo$
```

Efetivando as alterações

► commit

```
~/git-exemplo$ git commit lista-compras.txt -m 'versão inicial da lista de compras'
[master (root-commit) 5e832d1] versão inicial da lista de compras
1 file changed, 4 insertions(+)
create mode 100644 lista-compras.txt
```

► conferindo o estado atual

```
~/git-exemplo$ git status
No ramo master
nada a submeter, diretório de trabalho vazio
~/git-exemplo$
```

Histórico

► log

```
~/git-exemplo$ git log
commit 5e832d140ea6793e08cead4bfe8b10c1ce2b71c5
Author: Leslie H. Watter <leslie@watter.net>
Date:   Fri Sep 19 20:25:17 2014 -0300
```

```
    versão inicial da lista de compras
~/git-exemplo$
```

Distribuição

► repositórios remotos

```
~/git/moneylog-el$ git remote -v  
origin  https://github.com/watter/moneylog-el.git (fetch)  
origin  https://github.com/watter/moneylog-el.git (push)  
~/git/moneylog-el$
```

Recebendo as modificações

► pull

```
~/scm/linux$ git pull
remote: Counting objects: 874, done.
remote: Compressing objects: 100% (575/575), done.
remote: Total 874 (delta 476), reused 559 (delta 296)
Receiving objects: 100% (874/874), 1.25 MiB | 194.00 KiB/s, done.
Resolving deltas: 100% (476/476), done.
From git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux
 1536340..46be7b7  master      -> origin/master
 * [new tag]         v3.17-rc5  -> v3.17-rc5
Updating 1536340..46be7b7
Fast-forward
 Documentation/devicetree/bindings/input/atmel,maxtouch.txt | 4 --
 Documentation/devicetree/bindings/sound/rockchip-i2s.txt   | 2 +-
 Documentation/devicetree/bindings/spi/spi-rockchip.txt      | 8 +++-
 MAINTAINERS                                                  | 5 ++-
 Makefile                                                      | 2 +-
 arch/arm64/mm/init.c                                         | 3 +-
 arch/ia64/include/uapi/asm/unistd.h                          | 2 +-
 arch/ia64/pci/fixup.c                                        | 24 +----
 ....
```

Enviando as modificações

► push

```
~/scm/origem/destino/git$ git push origin master
Counting objects: 7, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 366 bytes | 0 bytes/s, done.
Total 4 (delta 3), reused 0 (delta 0)
To /home/leslie/scm/origem/destino/../git.git
    celd3a9..6a7baf9  master -> master
~/scm/origem/destino/git$
```


Que a força esteja com você



<http://superrebbies61.deviantart.com/>



Funcionalidades Legais

rebase

cherrypick

bisect

git rebase

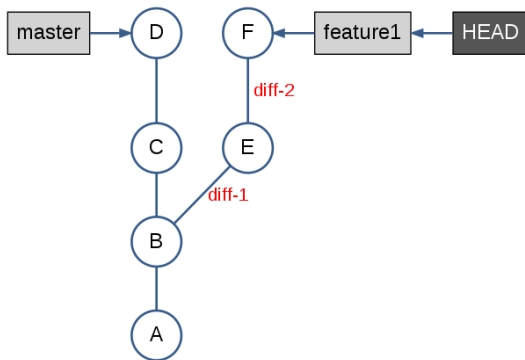
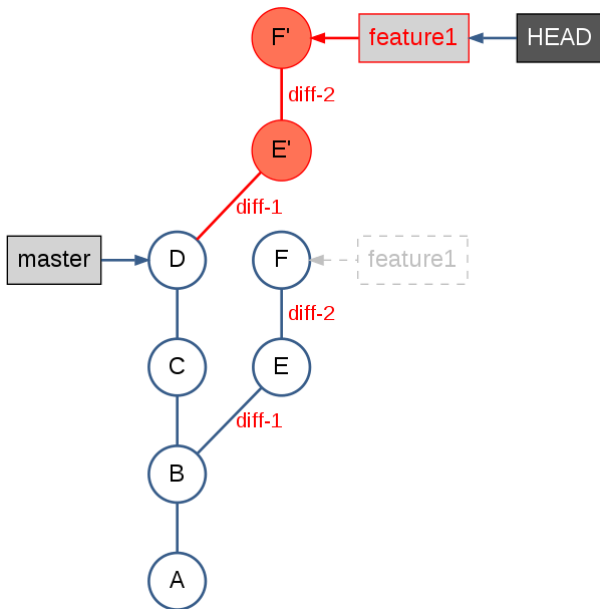


Figure : Antes do Rebase

git rebase



git cherrypick

E se fosse necessário pegar somente as modificações feitas no commit F do branch feature1?

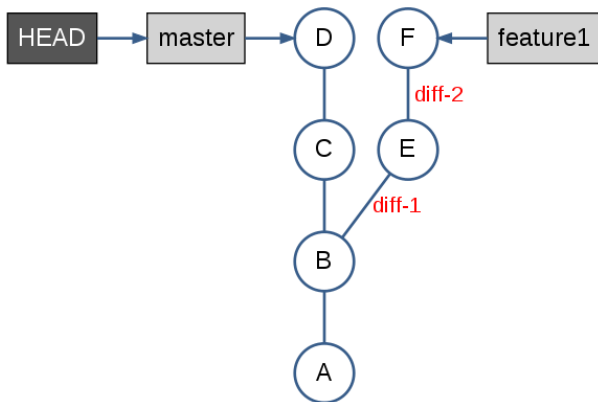
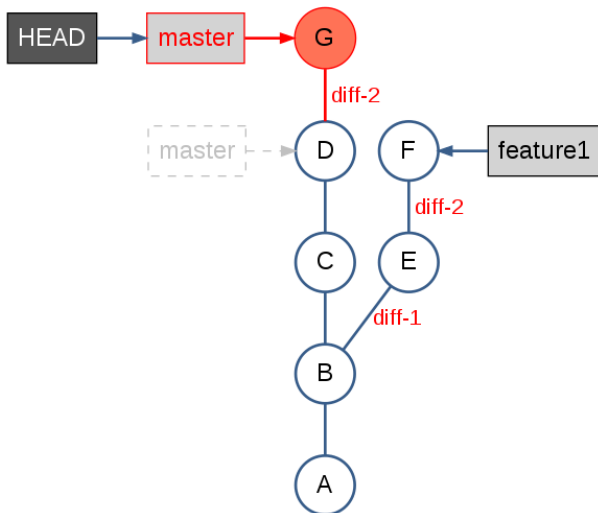


Figure : Cherrypick

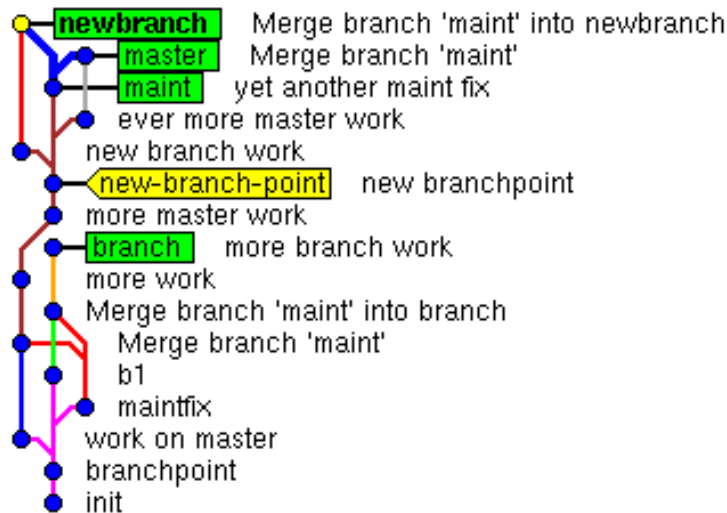
git cherrypick

```
git cherry-pick feature1
```

Aplica somente as mudanças feitas por F, isto é: diff-2



Descobrimos onde está um problema



git bisect

- ▶ `git bisect start`
- ▶ `git bisect bad $VERSION`
- ▶ `git bisect good $VERSION`
- ▶ `git bisect log`
- ▶ `git bisect replay $LOGFILE`
- ▶ `git bisect reset` - Terminei, volta pro head!

Interfaces

Algumas interfaces para o git:

- ▶ Visualização
 - ▶ terminal
 - ▶ gitk
 - ▶ gitX
 - ▶ gitweb
 - ▶ cgit
- ▶ Manipulação
 - ▶ magit
 - ▶ git-gui
 - ▶ eget
 - ▶ tortoisegit

git "tree" (git log --graph --pretty=oneline)

```
* 07a2039b8eb0af4ff464efd3dfd95de5c02648c6 Linux 2.6.30
* 96050dfb25966612008dcea7d342e91fa01e993c char: mxser, fix ISA board lookup
* a61d90d75d0f9e86432c45b496b4b0fbf0fd03dc jbd: fix race in buffer processing in
* 463aea1a1c49f1a7d4b50656dfd6c8bb33358b1b autofs4: remove hashed check in valida
* 586c7e6a280580fd94b662bf486f9bb31098d14b shm: fix unused warnings on nommu
* 13df635f080d411a152c92087a384135c1562fac Merge git://git.kernel.org/pub/scm/l
\
* 8437a617708d014d6f220df201a24960e00d57b1 kvm: fix kvm reboot crash when MAXSM
* eaa958402ea40851097d051f52ba1bb7a885efe9 cpumask: alloc zeroed cpumask for st
* 0281b5dc0350cbf6dd21ed558a33cccce77abc02 cpumask: introduce zalloc_cpumask_va
* 9cdba30283ecad22970e954c5595c9079d7a7f63 Merge branch 'for-linus' of git://
\
* | c1d4c41f2fdfe66dea957b76d005affba3e56b26 bsg: setting rq->bio to NULL
|/
* | fd4d34295b36c5d9a1aba3d3f2764d721db093be Merge git://git.kernel.org/pub/scm
\
* | 52ea3a56a3268bc2a5a7c75e98c81463004e38ef cls_cgroup: Fix oops when user sen
* | fdd7b4c3302c93f6833e338903ea77245eb510b4 r8169: fix crash when large packet
* | 4d11eff66889e65e9351111b1cc39f066f9f9729 Merge branch 'for-linus' of git:
\
\
|/
|/
* | 0e6e0271a210817e202c8a4bfffbd3e3c0616d1 md/raid5: fix bug in reshape code
* | a8c906ca3f63d05f0d25490cf82276f73c6fe095 md/raid5 - avoid deadlocks in get_
* | f001a70cdc61c01452d42e8b32fd7c7842ef62d5 md/raid5: use conf->raid_disks in
* | 3af968e066d593bc4dacc021715f3e95dddf0996f async: Fix lack of boot-time conso
* | 46056be71c37378f518711da651e4bb76d650ee9 Merge branch 'upstream' of git:/
\
\
* | 5636919b5c909fee54a6ef5226475ecae012ad02 MIPS: Outline udelay and fix a f
* | 3a553147eaad5d4de90ab1f695aa13ddbea684ec MIPS: ioctl.h: Fix headers_check
* | e25bfc9243f2eab12a2ce92b7f4b8a2e3e6949a6 MIPS: Cobalt: PCI bus is always
* | c9d89d97f0d174b9154820dd5c6726d1c794cd99 MIPS: Kconfig: Remove "Support f
```

gitk: linux-2.6

File Edit View Help

SHA1 ID: 07a20398e6b0af4ff4c4ef83df695d5a502c48ac Row 1 / 27298 Reading

Find next prev commit containing: Exact All fields

Search

Diff Old version New version Lines of context: 3 Ignore space change

Author: Linus Torvalds <torvalds@linux-foundation.org> 2009-06-10 00:05:27
 Committer: Linus Torvalds <torvalds@linux-foundation.org> 2009-06-10 00:05:27
 Tags: v2.6.30
 Parent: 96050d7b25956612008dcea74342e91fa01e993c (char: mxcser, fix ISA board lookup)
 Branch:
 Follows:
 Precedes:

Linux 2.6.30

Makefile

```

index 1065154..03373bb 100644
--- a/Makefile
+++ b/Makefile
@@ -1,7 +1,7 @@
-VERSION = 2
+PATCHLEVEL = 6
+SUBLEVEL = 30
+EXTRAVERSION = -rc8
+EXTRAVERSION =
+NAME = Msn-Eating Seals of Antiquity
+
# *DOCUMENTATION*
```

Linux 2.6.30

char: mxcser, fix ISA board lookup
 jbd: fix race in buffer processing in commit code
 autofs4: remove hashed check in validate_wait()
 shm: fix unused warnings on nommu
 Merge git://kernel.org/pub/scm/linux/kernel/git/trusty/linux-2.6-for-linus
 kvm: fix kvm reboot crash when MAXSMP is used
 cpumask: alloc zeroed cpumask for static cpumask_var_ts
 cpumask: introduce zalloc_cpumask_v
 Merge branch 'for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/davem/net-2.6
 Merge git://kernel.org/pub/scm/linux/kernel/git/davem/net-2.6
 cls_cgroup: Fix oops when user send improperly 'tc filter add' request
 r8169: fix crash when large packets are received
 Merge branch 'for-linus' of git://nelli.brown.name/md
 netfilter: fix hint in xtables.c when rthunk_size decreases

Linus Torvalds <torvalds@linux-foundation.org> 2009-06-10 00:05:27
 Peter Botha <peterb@goldcircle.co.za> 2009-06-09 21:16:32
 Jan Kara <jack@suse.cz> 2009-06-09 20:26:26
 Ian Kent <craven@themaw.net> 2009-06-09 20:26:24
 Mike Frysinger <vapier@gentoo.org> 2009-06-09 20:26:23
 Linus Torvalds <torvalds@linux-foundation.org> 2009-06-09 12:48:32
 Avi Kivity <avi@redhat.com> 2009-06-08 18:52:35
 Yinghai Lu <yinghai@kernel.org> 2009-06-08 18:51:36
 Yinghai Lu <yinghai@kernel.org> 2009-06-08 18:50:36
 Linus Torvalds <torvalds@linux-foundation.org> 2009-06-09 12:47:43
 FUJITA Tomonori <fujita.tomonori@lab.ntt.co.jp> 2009-06-09 10:17:37
 Linus Torvalds <torvalds@linux-foundation.org> 2009-06-09 12:47:27
 Minoru Usui <usui@mnm.nes.nec.co.jp> 2009-06-09 08:03:09
 Eric Dumazet <eric.dumazet@gmail.com> 2009-06-09 08:01:02
 Linus Torvalds <torvalds@linux-foundation.org> 2009-06-09 12:41:22
 NeilBrown <nrb@redhat.com> 2009-06-09 03:37:27

Comments
 Makefile

MacOS? GitX!

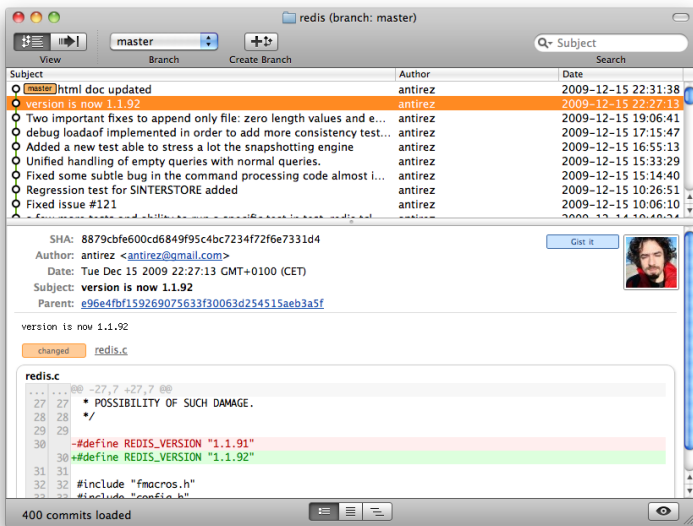


Figure : GitX - <http://gitx.frim.nl>

/pub/scm /

To clone one of these trees, install [git](#), and run:

```
git clone git://git.kernel.org/pub/scm/ + project path.
```

For convenience, the clone URL is available as the "git" link below, or on the status page of any repository.

For more information about [git](#), see an [overview](#), the [tutorial](#) or the [man pages](#).

Project	Description	Owner	Last Change
bluetooth/bluez-gnome.git	Bluetooth applications for ...	Marcel Holtmann	6 months ago summary shotlog log tree git
bluetooth/bluez-hcidump.git	Bluetooth packet analyzer	Marcel Holtmann	12 months ago summary shotlog log tree git
bluetooth/bluez.git	Bluetooth protocol stack for ...	Marcel Holtmann	10 hours ago summary shotlog log tree git
bluetooth/eglib.git	Embedded GLib	Marcel Holtmann	9 months ago summary shotlog log tree git
bluetooth/libgdbus.git	D-Bus integration with GLib	Marcel Holtmann	8 months ago summary shotlog log tree git
bluetooth/obexd.git	OBEX Server	Marcel Holtmann	13 days ago summary shotlog log tree git
bluetooth/openobex.git	Object Exchange (OBEX) protocol	Marcel Holtmann	2 weeks ago summary shotlog log tree git
boot/grub-fedora/grub-fedora.git	Development tree for Fedora's ...	Peter Jones	11 months ago summary shotlog log tree git
boot/syslinux/gpxe-for-syslinux.git	gPXE tree for SYSLINUX integration	H. Peter Anvin	6 months ago summary shotlog log tree git
boot/syslinux/syslinux.git	The Syslinux boot loader suite	H. Peter Anvin	65 min ago summary shotlog log tree git
boot/u-boot/galak/u-boot.git	Kumar's upstream u-boot repo	Kumar Gala	10 months ago summary shotlog log tree git
boot/warhog9/mkinitrd-syslinux.git	Syslinux branch of Fedora's ...	John Hawley	4 months ago summary shotlog log tree git
cogito/cogito-bundle.git	Example Cogito addon - mail ...	Petr Baudis	2 years ago summary shotlog log tree git
cogito/cogito-doc.git	Petr's user-friendly GIT inter ...	Petr Baudis	2 years ago summary shotlog log tree git
cogito/cogito.git	Petr's user-friendly GIT interface	Petr Baudis	2 years ago summary shotlog log tree git
daemon/cld/cld.git	Coarse locking daemon	Jeff Garzik	15 hours ago summary shotlog log tree git
daemon/distsrv/chunkd.git	Data storage daemon.	Jeff Garzik	3 weeks ago summary shotlog log tree git
daemon/distsrv/storaged.git	Data storage daemon.	Jeff Garzik	3 weeks ago summary shotlog log tree git
daemon/distsrv/tailed.git	Distributed key/value map ...	Jeff Garzik	9 days ago summary shotlog log tree git
daemon/dns/dvdns.git	Authoritative-only DNS server.	Jeff Garzik	3 months ago summary shotlog log tree git

index : kernel/git/torvalds/linux.git master switch Linux Torvalds

summary refs **log** tree commit diff stats log msg search

Age	Commit message (Expand)	Author	Files	Lines
6 days	Linux 3.17-rc4 v3.17-rc4	Linus Torvalds	1	-1/+1
6 days	Documentation: new page link in SubmittingPatches	Sudip Mukherjee	1	-0/+1
6 days	Documentation: NFS/RDMA: Document separate Kconfig symbols	Paul Bolle	1	-7/+9
6 days	Documentation: misc-devices: Rename freefall.c from hpfall.c in lis2lv02d	Masanari Iida	1	-1/+1
6 days	Documentation: i2c: rename variable "register" to "reg"	Jose Manuel Alarcon Roldan	1	-5/+5
6 days	Documentation: seq_file: Document seq_open_private(), seq_release_private()	Rob Jones	1	-0/+33
6 days	Merge tag 'pm+acpi-3.17-rc4' of git://git.kernel.org/pub/scm/linux/kernel/git/rafael/linux-pm	Linus Torvalds	12	-34/+122
6 days	Merge branch 'for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/viro/vfs	Linus Torvalds	4	-14/+18
6 days	Merge branch 'core-urgent-for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/tip/tip	Linus Torvalds	2	-12/+12
6 days	Merge branch 'timers-urgent-for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/tip/tip	Linus Torvalds	4	-11/+39
6 days	ufs: fix deadlocks introduced by sb mutex merge	Alexey Khoroshilov	2	-13/+8
7 days	Merge tag 'for-linus' of git://git.kernel.org/pub/scm/virt/kvm/kvm	Linus Torvalds	8	-17/+28
7 days	Merge tag 'fixes-for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/arm/arm-soc	Linus Torvalds	12	-59/+90
7 days	Merge tag 'xfs-for-linus-3.17-rc3' of git://git.kernel.org/pub/scm/linux/kernel/git/dgc/linux-xfs	Linus Torvalds	4	-12/+114
7 days	Merge tag 'for-linus-20140905' of git://git.infradead.org/linux-mtd	Linus Torvalds	2	-1/+5
7 days	timekeeping: Update timekeeper before updating vsyscall and pvclock	Thomas Gleixner	1	-2/+3
7 days	compat: nanosleep: Clarify error handling	Thomas Gleixner	1	-3/+21
8 days	Merge branch 'i2c-for-current' of git://git.kernel.org/pub/scm/linux/kernel/git/wsa/linux	Linus Torvalds	4	-14/+62
8 days	Merge tag 'at91-fixes' of git://github.com/at91linux/linux-at91 into fixes	Kevin Hilman	3	-2/+12
8 days	bus: arm-ccn: Move event cleanup routine	Pawel Moll	1	-26/+25
8 days	Merge branch 'for-linus' of git://git.kernel.org/pub/scm/linux/kernel/git/geert/linux-m68k	Linus Torvalds	3	-1/+5
8 days	ARM: at91/dt: rm9200: fix usb clock definition	Alexandre Belloni	1	-1/+1
8 days	ARM: at91: rm9200: fix clock registration	Alexandre Belloni	1	-1/+10
8 days	ARM: at91/dt: sam9g20: set at91sam9g20 pll_b driver	Gael PORTAY	1	-0/+1
8 days	mm: memcontrol: revert use of root_mem_cgorg res_counter	Johannes Weiner	1	-25/+78

Figure : Cgit Kernel Org

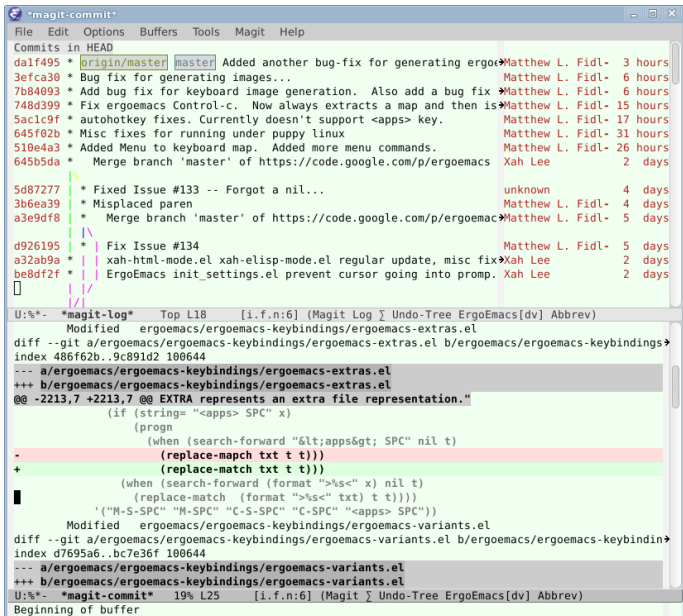
Ferramentas de Uso

- ▶ Tem alguma interface pra usar?

Ferramentas de Uso

- ▶ Tem alguma interface pra usar?
- ▶ e Linha de comando, não serve ??

magit + emacs



The screenshot shows the Magit commit window in Emacs. The window title is "magit-commit". The menu bar includes File, Edit, Options, Buffers, Tools, Magit, and Help. The main area is divided into two panes. The top pane, titled "Commits in HEAD", lists several commits with their hashes, authors, and commit messages. The bottom pane shows a diff view for the file "ergoemacs-keybindings/ergoemacs-extras.el".

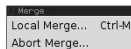
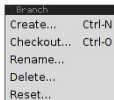
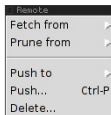
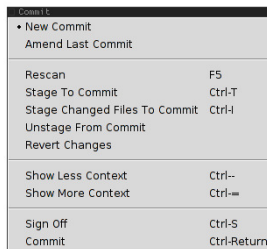
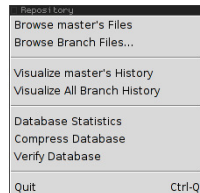
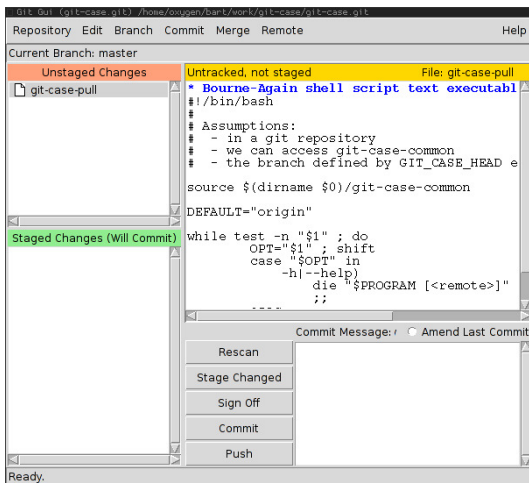
```
Commits in HEAD
dalf495 * origin/master master Added another bug-fix for generating ergoemacs-keybindings/ergoemacs-extras.el Matthew L. Fidl- 3 hours
3efca30 * Bug fix for generating images... Matthew L. Fidl- 6 hours
7b84093 * Add bug fix for keyboard image generation. Also add a bug fix for ergoemacs-keybindings/ergoemacs-extras.el Matthew L. Fidl- 6 hours
748d399 * Fix ergoemacs Control-c. Now always extracts a map and then is... Matthew L. Fidl- 15 hours
5ac1c9f * autohotkey fixes. Currently doesn't support <apps> key. Matthew L. Fidl- 17 hours
645f02b * Misc fixes for running under puppy linux Matthew L. Fidl- 31 hours
510e4a3 * Added Menu to keyboard map. Added more menu commands. Matthew L. Fidl- 26 hours
645b5da * Merge branch 'master' of https://code.google.com/p/ergoemacs Xah Lee 2 days

5d87277 * Fixed Issue #133 -- Forgot a nil... unknown 4 days
3b6ea39 * Misplaced paren Matthew L. Fidl- 4 days
a3e9df8 * Merge branch 'master' of https://code.google.com/p/ergoemacs Matthew L. Fidl- 5 days

d926195 * Fix Issue #134 Matthew L. Fidl- 5 days
a32ab9a * xah-html-mode.el xah-elisp-mode.el regular update, misc fix Xah Lee 2 days
be8df2f * ErgoEmacs init_settings.el prevent cursor going into promp. Xah Lee 2 days

U:%*-*magit-log* Top L18 [i.f.n:6] (Magit Log [Undo-Tree ErgoEmacs[dv] Abbrev)
Modified ergoemacs/ergoemacs-keybindings/ergoemacs-extras.el
diff --git a/ergoemacs/ergoemacs-keybindings/ergoemacs-extras.el b/ergoemacs/ergoemacs-keybindings/
index 486f62b..9c891d2 100644
--- a/ergoemacs/ergoemacs-keybindings/ergoemacs-extras.el
+++ b/ergoemacs/ergoemacs-keybindings/ergoemacs-extras.el
@@ -2213,7 +2213,7 @@ EXTRA represents an extra file representation."
  (if (string= "<apps> SPC" x)
    (progn
      (when (search-forward "<apps> SPC" nil t)
-       (replace-match txt t t))
+       (replace-match txt t t))
      (when (search-forward (format ">%s<" x) nil t)
        (replace-match (format ">%s<" txt) t t)))
    ("M-S-SPC" "M-SPC" "C-S-SPC" "C-SPC" "<apps> SPC"))
  Modified ergoemacs/ergoemacs-keybindings/ergoemacs-variants.el
diff --git a/ergoemacs/ergoemacs-keybindings/ergoemacs-variants.el b/ergoemacs/ergoemacs-keybindin
index d7695a6..bc7e36f 100644
--- a/ergoemacs/ergoemacs-keybindings/ergoemacs-variants.el
+++ b/ergoemacs/ergoemacs-keybindings/ergoemacs-variants.el
U:%*-*magit-commit* 19% L25 [i.f.n:6] (Magit [Undo-Tree ErgoEmacs[dv] Abbrev)
Beginning of buffer
```

git-gui



Ah tá, mas eu uso o Eclipse ...

tem a versão para eclipse também: Egit

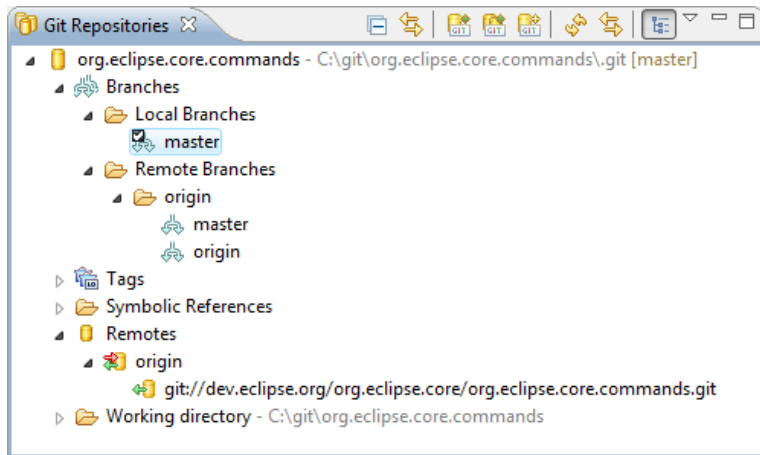


Figure : Egit - <http://www.eclipse.org/egit/>

E no Windows ?

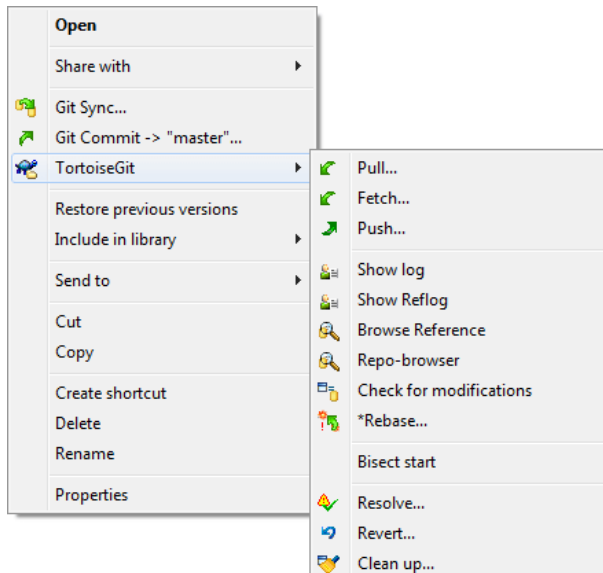
TortoiseGit



Figure : TortoiseGit - <https://code.google.com/p/tortoisegit/>

Ainda no Windows...

Menus do Tortoise Git integrados ao Explorer



Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

Manutenção de Configurações com GIT

Cenário: Modificações de configuração conforme Ordens de Serviço / Solicitações via E-mail.

Objetivo: rastreio do que foi mudado e porquê foi mudado

Local

Controle de arquivos de configuração que foram alterados.

Por exemplo, usando o pacote **etckeeper** no Debian/Ubuntu.

Distribuída

Manter uma base de controle de configurações centralizada.

Usa um repositório central e controle local com um **git** que faz push para o repositório remoto.

Trabalho em Conjunto = Colaboração

Ambiente

- ▶ Siemens - Curitiba
- ▶ FISL - Porto Alegre

Restrições de Comunicação

Em Curitiba só o e-mail funcionava/passava para fora.

Em Porto Alegre, conexão lenta.

Exemplo da conclusão de trabalho do Mestrado

Alan na Siemens, eu no FISL ... e a utilização de patches por e-mail. Eu sincronizava a árvore e publicava no gitorious de onde o Alan baixava.

Solução

- ▶ git format patch (Alan)
- ▶ git am (Leslie)

Desenvolvimento de Software

- ▶ Controle do histórico de modificações entre os desenvolvedores
- ▶ Controle de características novas a serem implementadas em branches

Exemplo: Manutenção de Versão Paralela

Necessidade

Testar e adaptar todo o código do sistema para rodar no IE6.

Problema

Todo dia o build (construção) do sistema tinha alterações e muitas vezes não compilava / rodava porque outro desenvolvedor havia comitado código quebrado.

Solução

Manter uma versão paralela do código do Sistema que estava no CVS em uma árvore GIT, que permitia atualizar meu código somente no momento em que fosse necessário. Permitia reaplicar minhas modificações e manter o código funcionando localmente.

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

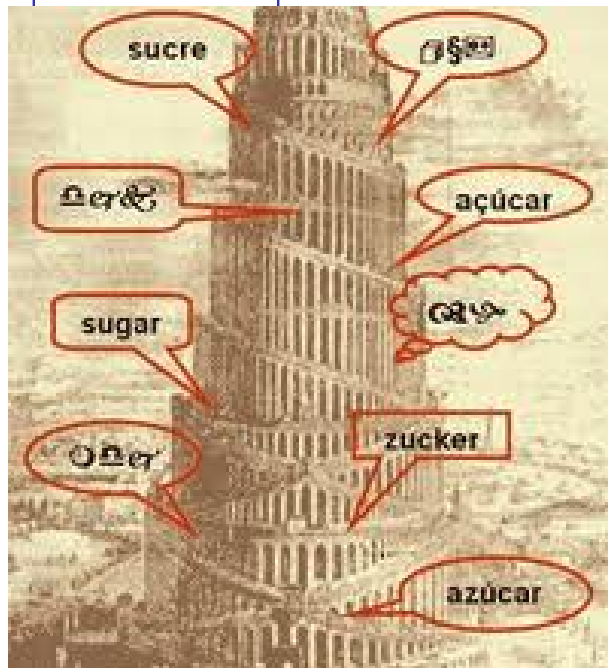
Revisão de Código

Aonde tem / Quem Oferece?

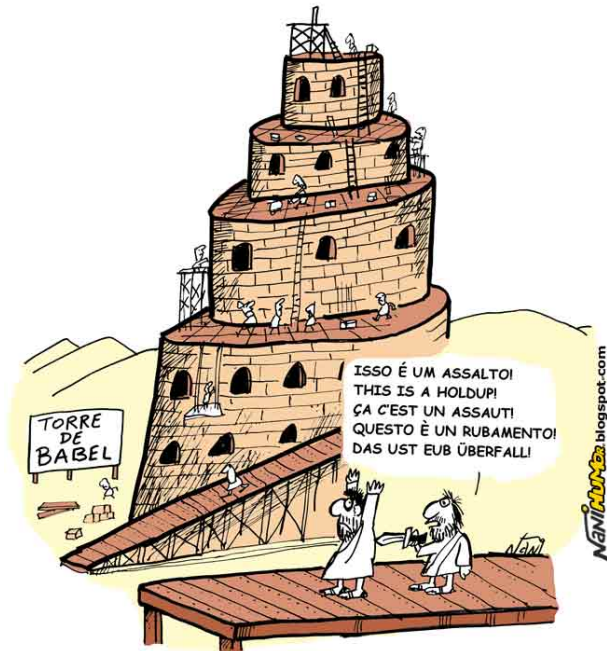
Extras

Conclusão

Por que definir boas práticas?



Por que definir boas práticas?



Melhores Práticas - Motivação

Não adianta de nada ter uma ferramenta poderosa e utilizar ela de forma errada.

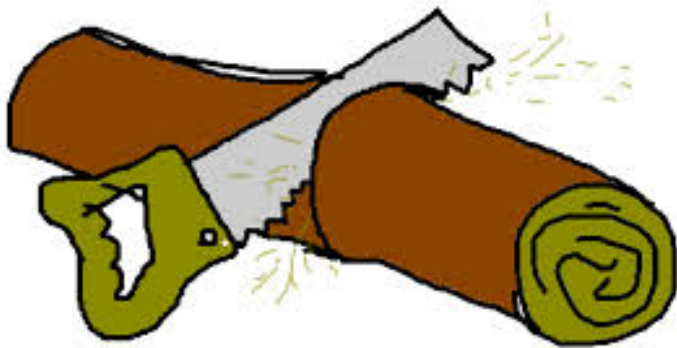
Mesmo usando uma ferramenta de controle de versão, existem as melhores práticas na hora de utilizá-la.

Imagine-se procurando um erro em um único commit de vários arquivos onde 9000 linhas foram modificadas e tudo foi feito de "uma vez"...

Não adianta uma ferramenta poderosa para a tarefa errada



Use a ferramenta certa para a tarefa



Melhores Práticas 1

- ▶ Comite Mudanças Relacionadas entre si
- ▶ Comite Frequentemente
- ▶ Não comite trabalho pela metade
- ▶ Teste Antes de Comitar
- ▶ Escreva Mensagens de Comit Relevantes e Descritivas
- ▶ O Controle de Versão não é um sistema de Backup
- ▶ Use Branches
- ▶ Concorde em um Fluxo de Trabalho e use-o

Melhores Práticas 2

- ▶ Comite antes e frequentemente
- ▶ Não entre em pânico
- ▶ Faça backups
- ▶ Não modifique o histórico publicado
- ▶ Escolha um fluxo de trabalho
- ▶ Divida o seu trabalho em repositórios
- ▶ Escreva mensagens de commit úteis
- ▶ Mantenha-se atualizado
- ▶ Faça manutenção periódica
- ▶ Reforce o uso de Padrões
- ▶ Utilize ferramentas úteis
- ▶ Integre com ferramentas externas
- ▶

Exemplo MP: Envio de Patches para Kernel do Linux

1. diff -up
2. Describe your changes.
3. Separate your changes.
4. Style check your changes.
5. Select e-mail destination.
6. Select your CC (e-mail carbon copy) list.
7. No MIME, no links, no compression, no attachments. Just plain text.
8. E-mail size.
9. Name your kernel version.
10. Don't get discouraged. Re-submit.
11. Include PATCH in the subject
12. Sign your work
13. ...
14. (ao todo são 812 linhas de texto)

Fonte: <http://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/tree/Documentation/SubmittingPatches>

Envio de Patches para Kernel do Linux

Em resumo:

- ▶ Envie as suas correções usando as regras pré-estabelecidas
- ▶ Todos ganham com essas regras pré-definidas.

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

Revisão de Código

Até os melhores erram . . .




Gerrit

All Projects Documentation

Open Merged Abandoned

status:open Search Register Sign In

 **eclipse**

Search for status:open

Subject	Status	Owner	Project	Branch	Updated	CR	V
▶ Bug 443826 - Resizing/moving of a combined fragment works incorrectly in ...		Anatoliy Tischenko	papyrus/org.eclipse.papyrus	master	10:06 AM		
[444045] Messenger does not support nested AnnotationMirrors		Stefan Ocke	jdt/eclipse.jdt.core	master	9:32 AM		
436929 - Changes in imported dependencyManagement poms do not cause refresh ...		Anton Tanasenko	m2e/m2e-core	master	9:18 AM		
Bug 438972 - VarDoc doesn't support array syntax		Michal Niewrzal	pdtd/org.eclipse.pdt	master	8:54 AM		✓
[443314] move cursor into substructure when moving down		Florian Thienel	mylyn/org.eclipse.mylyn.docs.vex	master	6:03 AM		✓
[421401] do not return an offset outside the box		Florian Thienel	mylyn/org.eclipse.mylyn.docs.vex	master	6:01 AM		✓
322716: [website] update bug count on contributors page		Frank Becker	www.eclipse.org/mylyn	staging	3:56 AM		
443871: show Gerrit topic in the task editor		Marc-Andre Laperle	mylyn/org.eclipse.mylyn.reviews	master	Sep 13		✓
Bug 361731 - Set size constraints on children of PartSashContainer		Steven Spungin	platform/eclipse.platform.ui	master	Sep 13		✓
Enable logging of all SWTBot tests with log4j.		Andrew Ferrazzutti	linuxtools/org.eclipse.linuxtools	master	Sep 13		✓
Introduce context menu "import changed projects"		Tobias Melcher	egit/egit	stable-3.4	Sep 13	✗	
Bug 441496 - Deprecated members styling doesn't work for traits static methods		Michal Niewrzal	pdtd/org.eclipse.pdt	master	Sep 13		✓

Figure : Gerrit - Exemplo

Gerrit

platform/eclipse.platform.ui	master	Sep 13		✗
linuxtools/org.eclipse.linuxtools	master	Sep 12		✓
linuxtools/org.eclipse.linuxtools	master (CTF performance)	Sep 12	+1	✓
linuxtools/org.eclipse.linuxtools	master	Sep 12		✓
linuxtools/org.eclipse.linuxtools	master	Sep 12	+1	✓
linuxtools/org.eclipse.linuxtools	master	Sep 12		✓
linuxtools/org.eclipse.linuxtools	master	Sep 12		✓
cdt/org.eclipse.cdt	master	Sep 12		✓
nebula/org.eclipse.nebula	master	Sep 12		✓
linuxtools/org.eclipse.linuxtools	master	Sep 12	-1	✓
cdt/org.eclipse.cdt	master	Sep 12	+1	✓

Code-Review
by Bernd Hufmann

Figure : Gerrit - Review

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

Aonde tem / Quem Oferece?

CVS

- ▶ sourceforge.net

SVN

- ▶ sourceforge
- ▶ code.google.com

Hospedagem Git

- ▶ `http://github.com`
- ▶ `http://gitorious.org`
- ▶ `http://repo.or.cz`
- ▶ `http://bitbucket.org`
- ▶ e muitos outros ...

Github

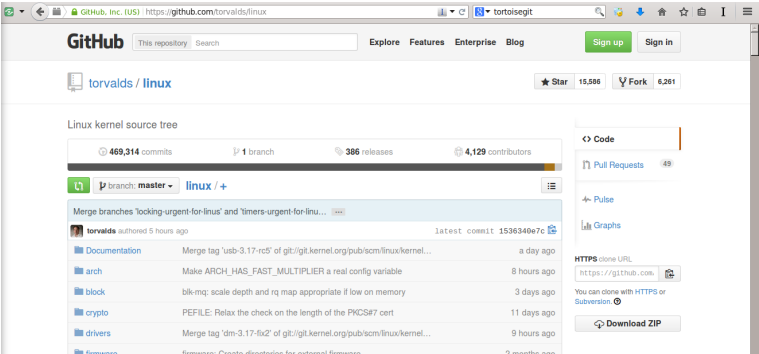


Figure : GitHub

Gitoriuos

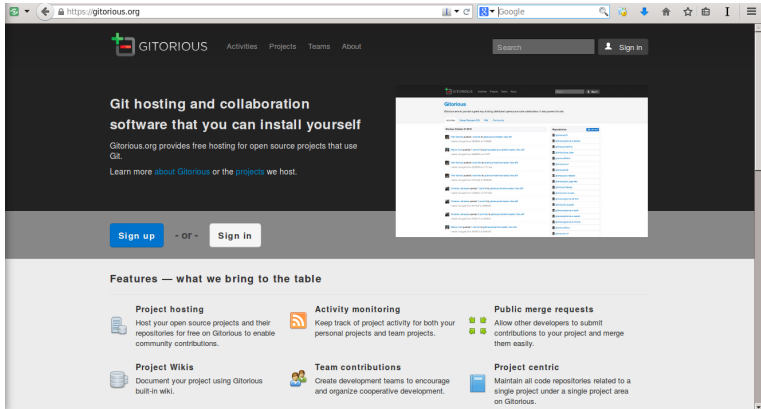
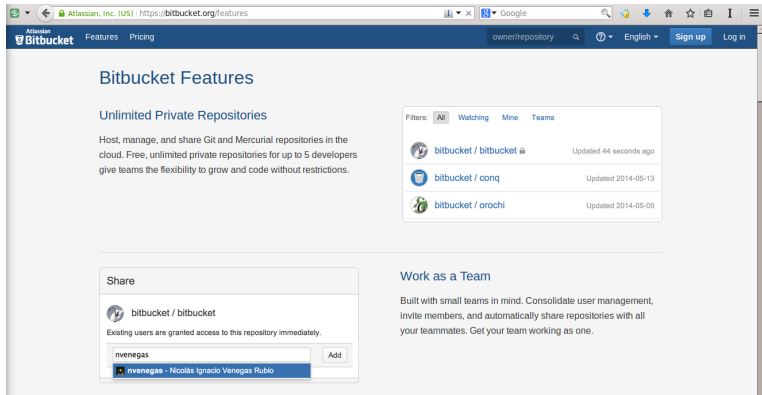


Figure : Gitorious

BitBucket



The screenshot displays the BitBucket website interface. At the top, the navigation bar includes the BitBucket logo, links for 'Features' and 'Pricing', a search bar with the text 'owner/repository', a language selector set to 'English', and buttons for 'Sign up' and 'Log in'. The main content area is titled 'Bitbucket Features'. Under the sub-header 'Unlimited Private Repositories', a paragraph states: 'Host, manage, and share Git and Mercurial repositories in the cloud. Free, unlimited private repositories for up to 5 developers give teams the flexibility to grow and code without restrictions.' To the right, a repository list is shown with filters for 'All', 'Watching', 'Mine', and 'Teams'. The list contains three entries: 'bitbucket / bitbucket' (updated 44 seconds ago), 'bitbucket / conq' (updated 2014-05-13), and 'bitbucket / orochi' (updated 2014-05-05). Below the main text, a 'Share' section shows a repository 'bitbucket / bitbucket' and a list of users with 'nvenegas - Nicolás Ignacio Venegas Rubio' selected. On the right side of the page, the section 'Work as a Team' is followed by the text: 'Built with small teams in mind. Consolidate user management, invite members, and automatically share repositories with all your teammates. Get your team working as one.'

Bitbucket Features

Unlimited Private Repositories

Host, manage, and share Git and Mercurial repositories in the cloud. Free, unlimited private repositories for up to 5 developers give teams the flexibility to grow and code without restrictions.

Filters: All Watching Mine Teams

- bitbucket / bitbucket Updated 44 seconds ago
- bitbucket / conq Updated 2014-05-13
- bitbucket / orochi Updated 2014-05-05

Share

bitbucket / bitbucket

Existing users are granted access to this repository immediately.

nvenegas Add

nvenegas - Nicolás Ignacio Venegas Rubio

Work as a Team

Built with small teams in mind. Consolidate user management, invite members, and automatically share repositories with all your teammates. Get your team working as one.

Figure : Bitbucket

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

Referências e Leituras

Vídeos Educativos sobre Git

- ▶ <https://training.github.com/resources/videos/>

Teste ao vivo com git

- ▶ <http://try.github.io>

Referências e Leituras

- ▶ ProGit - <http://git-scm.com/book>
- ▶ Git Tower - <http://www.git-tower.com/learn/ebook/command-line/introduction>
- ▶ Tutorial Magit - http://ergoemacs.org/emacs/emacs_magit-mode_tutorial.html
- ▶ <http://git.or.cz>
- ▶ <http://git-scm.com>
- ▶ <http://git.or.cz/gitwiki>
- ▶ <http://gitready.com>
- ▶ Intro to Git por Bart Jukie – PDF e Vídeo - <http://excess.org/article/2008/07/ogre-git-tutorial/>

Referências e Leituras

Melhores práticas

- ▶ <http://sethrobertson.github.io/GitBestPractices/>
- ▶ <http://pt.slideshare.net/TarinGamberini/commit-messages-goodpractices>
- ▶ <http://www.git-tower.com/learn/ebook/command-line/appendix/best-practices>
- ▶ <http://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/tree/Documentation/SubmittingPatches>
- ▶ <http://tbaggery.com/2008/04/19/a-note-about-git-commit-messages.html>

Referências e Leituras

Revisão de Código

- ▶ Gerrit - <https://code.google.com/p/gerrit/>
 - ▶ Gerrit no github - <http://gerrithub.io/>
 - ▶ Gerrit no Eclipse -
<https://git.eclipse.org/r/#/q/status:open,n,z>
 - ▶ Treinamento - <http://gerrit-training.scmforge.com/>

Topic

Introdução

Versionamento Manual

Conceitos

Sistemas de Controle de Versão

GIT

Casos de Utilização

Melhores Práticas em Controles de Versão

Revisão de Código

Aonde tem / Quem Oferece?

Extras

Conclusão

Coisas para Lembrar

- ▶ Use o controle de versão sempre que puder, uma hora ou outra ele vai salvar a sua pele
- ▶ Compartilhe informações, normalmente a informação que você compartilha hoje volta muito mais rica no futuro.
- ▶ ...

Até Breve



Contato

Leslie H. Watter

leslie em watter.net / watter no gmail.com

<http://github.com/watter>

<http://www.watter.net>