
Working with GIT and PostgreSQL

Heikki Linnakangas

Distributed Version Control Systems

- All work is done on a local repository
- You can create branches, commit, merge and more, all in your local repository
- The “master” repository is not any different from your local one
 - Except that people regard it as the master

Getting started

- Initialize the repository

```
~$ git-clone http://git.postgresql.org/git/postgresql.git  
pgsql-temp
```

```
Initialized empty Git repository in /home/hlinnaka/pgsql-temp/.git/
```

```
Checking out files: 100% (3303/3303), done.
```

```
~/pgsql-temp$ ll
```

```
total 831  
-rw-r--r--  1 hlinnaka hlinnaka   387 2008-10-13 16:44 aclocal.m4  
drwxr-xr-x  2 hlinnaka hlinnaka   608 2008-10-13 16:44 config  
-rwxr-xr-x  1 hlinnaka hlinnaka 758332 2008-10-13 16:44 configure  
-rw-r--r--  1 hlinnaka hlinnaka 58535 2008-10-13 16:44 configure.in  
drwxr-xr-x 38 hlinnaka hlinnaka  1120 2008-10-13 16:44 contrib  
-rw-r--r--  1 hlinnaka hlinnaka  1192 2008-10-13 16:44 COPYRIGHT  
drwxr-xr-x  3 hlinnaka hlinnaka   904 2008-10-13 16:44 doc  
-rw-r--r--  1 hlinnaka hlinnaka  3710 2008-10-13 16:44 GNUmakefile.in  
-rw-r--r--  1 hlinnaka hlinnaka  1423 2008-10-13 16:44 Makefile  
-rw-r--r--  1 hlinnaka hlinnaka  1287 2008-10-13 16:44 README  
-rw-r--r--  1 hlinnaka hlinnaka   767 2008-10-13 16:44 README.CVS  
drwxr-xr-x 14 hlinnaka hlinnaka   576 2008-10-13 16:44 src
```

Branches

- There's two kinds of branches
 - Local
 - Remote
- A remote branch is a copy of a branch in a remote repository
- All your changes go to local branches

Branches

```
hlinnaka@heikkilaptop:~/git-sandbox/pgsql$ git-branch -a
dsm
fork2                origin/HEAD
fork3                origin/MANUAL_DIST
fsm-addons          origin/PG95-DIST
fsm-flat            origin/PG95_DIST
fsm-foo             origin/REL2_0B
fsm-nowal           origin/REL6_4
fsm-nowal-old       origin/REL6_5_PATCHES
fsm-rewrite         origin/REL7_0_PATCHES
fsm-rewrite-old     origin/REL7_1_STABLE
fsm-rewrite2        origin/REL7_2_STABLE
fsm-rewrite3        origin/REL7_3_STABLE
* fsm-updateonreplay origin/REL7_4_STABLE
master              origin/REL8_0_0
master-wtf          origin/REL8_0_STABLE
multicolgin         origin/REL8_1_STABLE
pgstandby           origin/REL8_2_STABLE
settablespace       origin/REL8_3_STABLE
windowfunc          origin/Release_1_0_3
windowfuncv6        origin/WIN32_DEV
xlogrefactor        origin/ecpg_big_bison
xlogrefactor3       origin/master
imola/fsm-rewrite3  windowfunc/master
```

Creating your own branch

- `git-branch <branch name> [starting point]`

```
$ git-branch --track mycoolfeature origin/master
Branch mycoolfeature set up to track remote branch
  refs/remotes/origin/master.
$ git-checkout mycoolfeature
Switched to branch "mycoolfeature"
```

The Index

- The “Index” is a staging area before committing
- You must `git-add` all new **and modified** files before committing
 - or use `git-add -u`, `git-commit -a`
- Use `git-diff --cached` to show currently staged changes
- `git-diff` shows differences between working copy and the index!

Committing

- `git-commit`
- Asks for a log message and creates a new commit

```
$ git-commit -m "Add XLogRecordPageWithFreeSpace function"
```

```
Created commit 9bf287e: Add XLogRecordPageWithFreeSpace function
```

```
1 files changed, 32 insertions(+), 0 deletions(-)
```

```
$ git-log -n 2
```

```
commit 9bf287e43b1d792f2f72fd8e6c1b539cf09067e5
```

```
Author: Heikki Linnakangas <heikki@enterprisedb.com>
```

```
Date: Tue Oct 14 14:13:25 2008 +0300
```

```
Add XLogRecordPageWithFreeSpace function
```

```
commit 87c17c798387c47f44ca769bf3d6ffacfcce384d
```

```
Author: Heikki Linnakangas <heikki@enterprisedb.com>
```

```
Date: Tue Oct 14 08:06:39 2008 +0000
```

```
Fix oversight in the relation forks patch: forgot to copy fork number to  
fsync requests. This should fix the installcheck failure of the buildfarm  
member "kudu".
```


Oops...

- Forgot something?
- Don't worry, you can *amend* a commit!

```
$ emacs src/include/storage/freespace.h
$ make -j -s
$ git-add src/include/storage/freespace.h
$ git-commit --amend
Created commit 5bb5d7c: Add XLogRecordPageWithFreeSpace function
 2 files changed, 34 insertions(+), 0 deletions(-)
$ git-log -n 1 --stat
commit 5bb5d7c115c0bae1d619d67796adc0013f244aa5
Author: Heikki Linnakangas <heikki@enterprisedb.com>
Date: Tue Oct 14 14:13:25 2008 +0300

    Add XLogRecordPageWithFreeSpace function

src/backend/storage/freespace/freespace.c | 32 ++++++
src/include/storage/freespace.h           |  2 +
2 files changed, 34 insertions(+), 0 deletions(-)
```

- Changes history!

Viewing history

- `git-log`
 - p for viewing patches
 - stat for summary of changes per file
- `git-diff`
 - generates a patch
 - no context diffs, sorry (use `filterdiff`)
- Can use between any two commits:
 - `git-diff origin/REL8_2_STABLE..origin/REL8_3_STABLE`
 - `git-log de1986b..87c17c7`
 - `git-diff origin/master..`

Pulling

- To update your branch with changes in remote repository:

```
$ git-pull
```

```
Updating 87c17c7..5f9d02e
```

```
Fast forward
```

```
src/interfaces/ecpg/ChangeLog | 4 ++
src/interfaces/ecpg/preproc/Makefile | 1 -
src/interfaces/ecpg/preproc/preproc.y | 9 ++--
src/interfaces/ecpg/test/expected/sql-fetch.c | 56 ++++++++
++++---
src/interfaces/ecpg/test/expected/sql-fetch.stderr | 41 ++++++++
+--
src/interfaces/ecpg/test/expected/sql-fetch.stdout | 1 +
src/interfaces/ecpg/test/sql/fetch.pgc | 11 ++++-
7 files changed, 102 insertions(+), 21 deletions(-)
```

Pulling

- `git-pull` is a combination of two commands
 - `git-fetch`
 - `git-merge`
- `git-fetch` connects to the remote repository and updates remote branches in local repository
- `git-merge` then merges the changes into the local branch you're working on

Pushing

- To update a branch in a remote repository:

```
$ git-push --dry-run imola
```

```
To ssh://XXX/pgsql.git/.git
```

```
6e907e0..87c17c7 origin/HEAD -> PostgreSQL/HEAD
```

```
6e907e0..87c17c7 origin/master -> PostgreSQL/master
```

```
+ a7135a2...0833d0c dsm -> imola/dsm (forced update)
```

```
+ 5ecc732...940d522 fsm-nowal -> imola/fsm-nowal (forced update)
```

```
de1986b..87c17c7 master -> imola/master
```

```
* [new branch] fsm-addons -> imola/fsm-addons
```

```
* [new branch] fsm-foo -> imola/fsm-foo
```

```
* [new branch] fsm-updateonreplay -> imola/fsm-updateonreplay
```

```
* [new branch] settablespace -> imola/settablespace
```

```
* [new branch] windowfuncv6 -> imola/windowfuncv6
```

GIT config file

- .. is in .git/config
- Edit with a text editor

```
[core]
repositoryformatversion = 0
filemode = true
bare = false
logallrefupdates = true
[remote "origin"]
#url = git://git.postgresql.org/git/postgresql.git
url = http://git.postgresql.org/git/postgresql.git
fetch = +refs/heads/*:refs/remotes/origin/*

[branch "fsm-rewrite"]
remote = origin
merge = refs/heads/master

[remote "gsoc08-tsoc"]
url = http://git.postgresql.org/git/~wulczer/gsoc08-tss.git
fetch = +refs/heads/*:refs/remotes/gsoc08-tsoc/*

[remote "imola"]
url = ssh://perftester@imola/home/perftester/pgsql.git/.git
push = +refs/heads/*:refs/remotes/imola/*
push = +refs/remotes/origin/*:refs/remotes/PostgreSQL/*

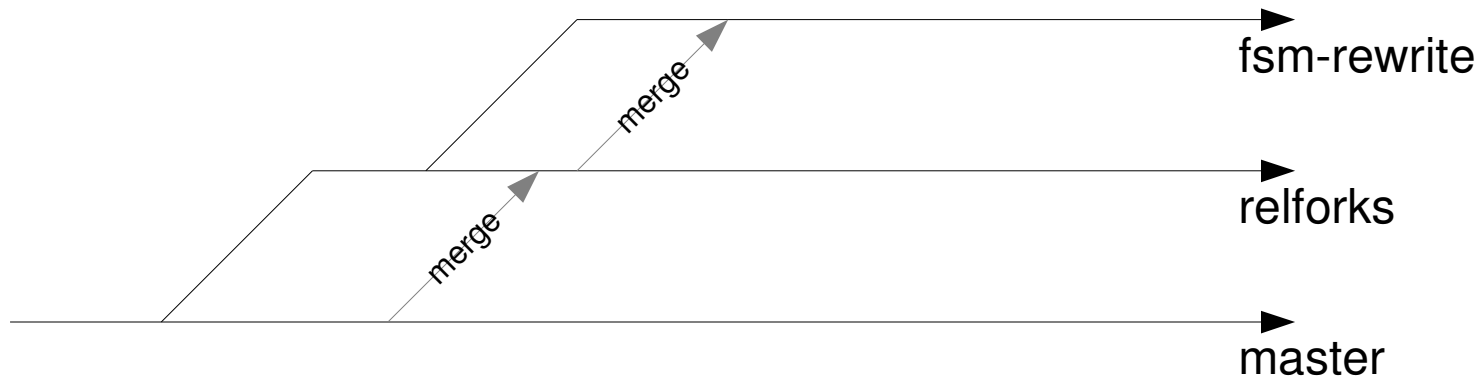
[remote "windowfunc"]
url = http://git.postgresql.org/git/~davidfetter/window_functions/.git/
fetch = +refs/heads/*:refs/remotes/windowfunc/*

[branch "windowfunc"]
remote = windowfunc
merge = refs/heads/master

[gui]
geometry = 1026x456+0+0 201 187
```

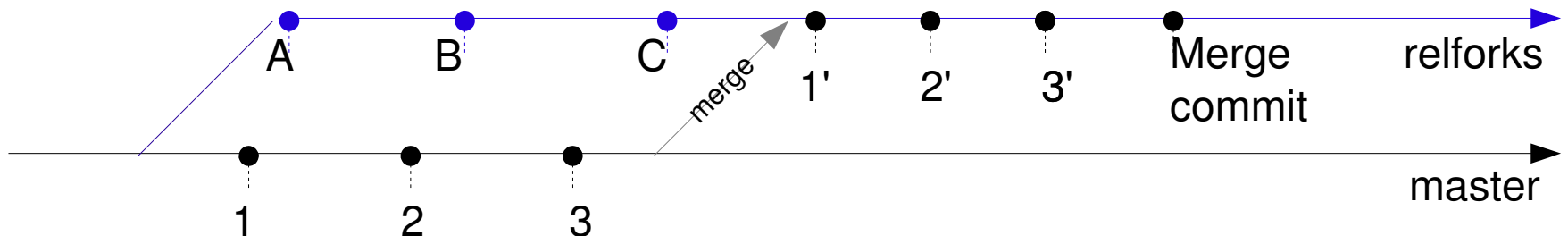
Maintaining a patch

- I started working on relation forks and FSM rewrite in spring
- Converted to GIT in summer
- Maintained a “fork of a fork” for some time



Merging

- `git-merge master`
- `git-merge` applies the commits from the other branch, and creates a merge commit



Merging

- `git-checkout relforks`
- `git-merge master`
- Fix conflicts manually with text editor
- Use `git-add` to stage the manual changes

```
commit 457423a248deebd64c411ffe04fc27208e542cba
Merge: 9bae34f... 483572b...
Author: Heikki Linnakangas <hlinnaka@heikkilaptop.(none)>
Date: Tue Jun 24 22:22:27 2008 +0300
```

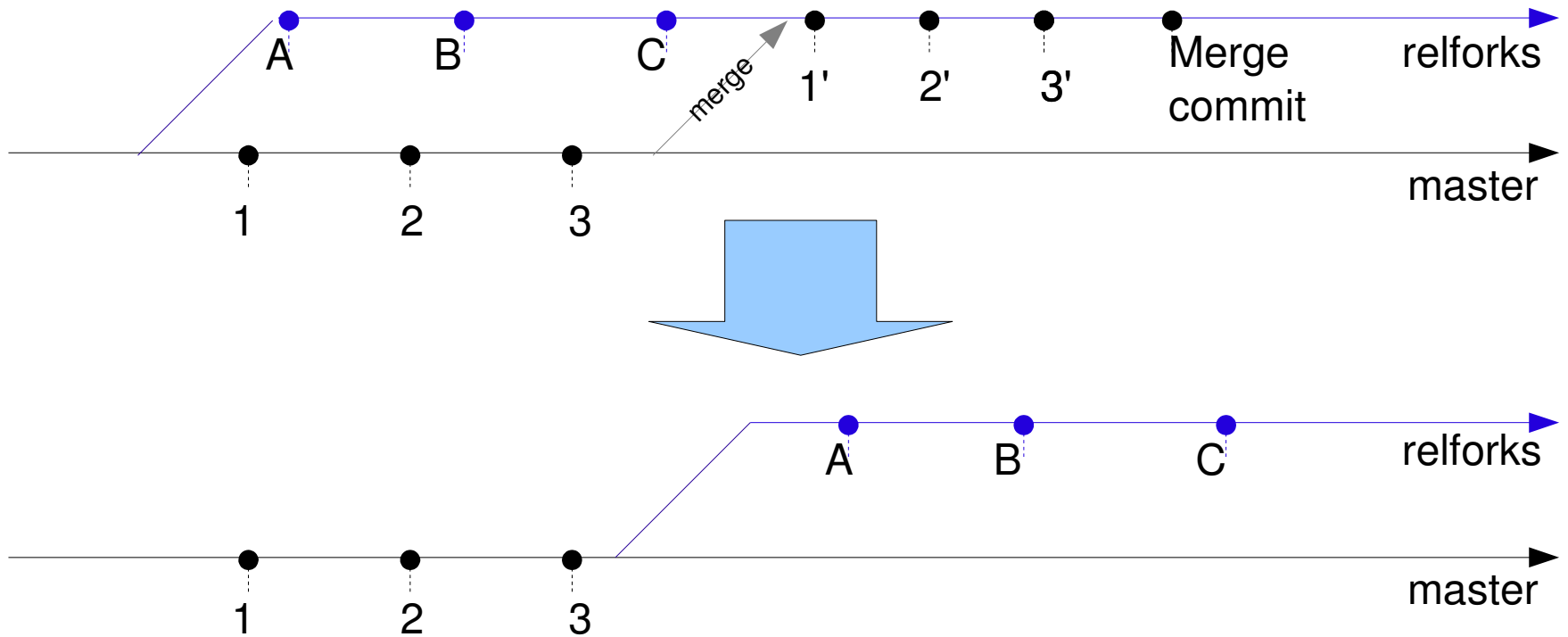
```
Merge branch 'master' of http://git.postgresql.org/git/postgresql into fsm-rewrite
```

```
Conflicts:
```

```
contrib/pg_freespacemap/pg_freespacemap.c
src/backend/access/gin/gininsert.c
src/backend/access/gin/ginutil.c
src/backend/access/gin/ginvacuum.c
src/backend/access/gist/gist.c
src/backend/access/gist/gistutil.c
```

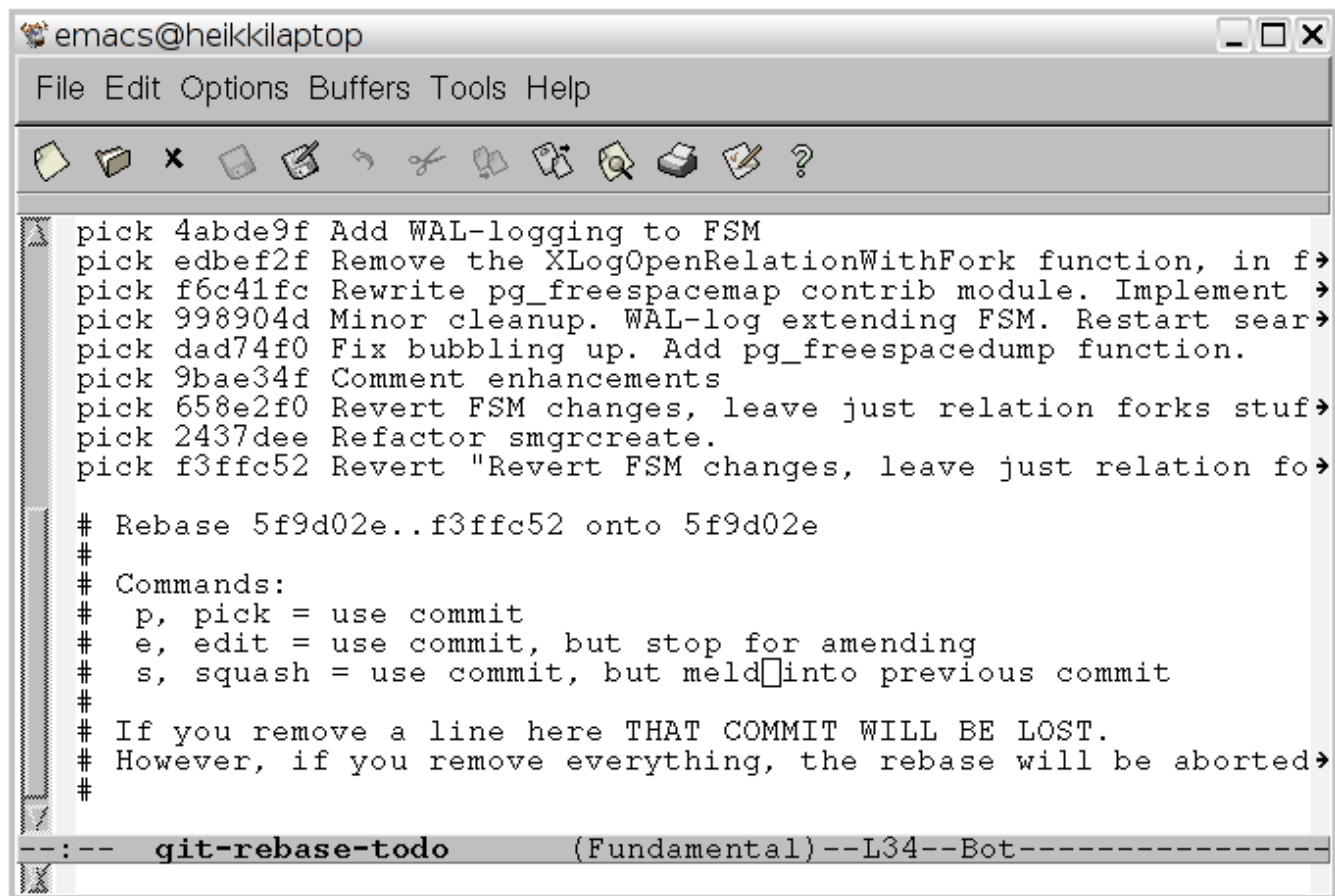
git-rebase

- Powerful command
- Modifies history, use only on a private branch!



Interactive rebase

- Git-rebase -i
- You can skip, add, squash, edit commits



```
emacs@heikkilaptop
File Edit Options Buffers Tools Help

pick 4abde9f Add WAL-logging to FSM
pick edbef2f Remove the XLogOpenRelationWithFork function, in f
pick f6c41fc Rewrite pg_freemap contrib module. Implement
pick 998904d Minor cleanup. WAL-log extending FSM. Restart sear
pick dad74f0 Fix bubbling up. Add pg_freemapdump function.
pick 9bae34f Comment enhancements
pick 658e2f0 Revert FSM changes, leave just relation forks stuf
pick 2437dee Refactor smgrcreate.
pick f3ffc52 Revert "Revert FSM changes, leave just relation fo

# Rebase 5f9d02e..f3ffc52 onto 5f9d02e
#
# Commands:
#   p, pick = use commit
#   e, edit = use commit, but stop for amending
#   s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted
#

--:-- git-rebase-todo (Fundamental)--L34--Bot-----
```

EnterpriseDB CVS->GIT migration

- We had to bring forward our Advanced Server product to PostgreSQL 8.3 release
- Used CVS before
- Decided to try out GIT, because of powerful branching and merging capabilities
- Used `git-rebase`
- Rebased over 2000 patches, fixing merge conflicts manually

EnterpriseDB CVS->GIT migration

- First, I converted our CVS repository to GIT
 - Used a bunch of custom scripts.
 - should've used `git-cvimport`
- Then, rebased that over a snapshot of PostgreSQL GIT master (= CVS HEAD), at the point just before branching REL8_3_STABLE:
- `git-rebase -i --onto PostgreSQL/REL8_3_STABLE PostgreSQL/REL8_2_STABLE EDBAS83r2-try5`
- we had to pick out backported features etc.

git-gui

The screenshot shows the git-gui application window titled "gitk: pgsq". The interface is divided into three main sections: a commit graph on the left, a commit message in the center, and a commit log on the right.

Commit Graph: A vertical line of colored circles (yellow, blue, green, red) represents the commit history. A yellow circle at the top is labeled "fsm-rewrite2".

Commit Message: The message for the selected commit is: "Merge branch 'relforks' into fsm-rewrite2". Below this, it lists several changes, including "Merge commit 'origin' into relforks", "Fix corner-case bug introduced with HOT: if REINDEX TABLE pg_class...", "Install checks in executor startup...", "Improve INTERSECT/EXCEPT hashing...", "Support hashing for duplicate-elimination...", "Teach the system how to use hashing for UNION...", "Do not allow Unique nodes to be scanned backwards...", "Department of second thoughts: fix newly-added code in planner.c...", "Add an ORDER BY to one more SELECT DISTINCT test case...", "In ReadOrZeroBuffer (and related entry points), don't bother to call Move pgstat.tmp...", "Merge branch 'relforks' into fsm-rewrite2", "Merge commit 'origin' into relforks", "Fix some message style guideline violations in pg_regress...", "Improve SELECT DISTINCT to consider hash aggregation...", "Improve CREATE/DROP/RENAME DATABASE so that when failing because the source Make GROUP BY work properly...", "Tighten up the sanity checks in TypeCreate(): pass-by-value types must have", "Fix copy-and-pasteo that's causing pg_regress to lie about which file it can't", "Rearrange the querytree representation of ORDER BY/GROUP BY/DISTINCT items", "Cope with Tcl versions that do not create a tclsh symlink to the version-", "Add a few more DTrace probes to the backend.", "Rearrange the code in auth.c so that all functions for a single authentication", "Merge branch 'relforks' into fsm-rewrite2", "Remove comma at the end of last enum value", "Whitespace changes, and revert two unrelated .po changes.", "Merge branch 'relforks' into fsm-rewrite2", "Cosmetic changes. Put back a FSM call, accidentally removed.", "Merge commit 'origin' into relforks", "Move ident authentication code into auth.c along with the other authentication", "Fix parser so that we don't modify the user-written ORDER BY list in order".

Commit Log: A list of commits with author names and timestamps. The authors include Heikki Linnakangas, Tom Lane, and Magnus Hagander. The timestamps range from 2008-08-01 to 2008-08-11.

Bottom Panel: A search bar with "SHA1 ID:" followed by "5002cea677bec30fe3fb92a8b3a8d70d1dd98b62". It includes navigation buttons for "next", "prev", "commit", and "containing:". The "Row" indicator shows "1 / 27261". There are also buttons for "Exact" and "All fields".

More commands: git-apply

- Like patch
- Use `--index` to stage the changes immediately

```
$ git-apply lwlock-measure-1.patch
error: patch failed: src/backend/storage/lmgr/lwlock.c:30
error: src/backend/storage/lmgr/lwlock.c: patch does not apply
$ git-apply --index --reject lwlock-measure-1.patch
Checking patch src/backend/storage/lmgr/lwlock.c...

...

error: patch failed: src/backend/storage/lmgr/lwlock.c:445
Checking patch src/include/nodes/nodes.h...
Applying patch src/backend/storage/lmgr/lwlock.c with 2 rejects...
Hunk #1 applied cleanly.
Rejected hunk #2.
Hunk #3 applied cleanly.
Hunk #4 applied cleanly.
Hunk #5 applied cleanly.
Rejected hunk #6.
Applied patch src/include/nodes/nodes.h cleanly.
```

More commands: git-cherry-pick

- Creates a patch of a given commit, and applies it to another branch

More commands: git-stash

- Creates a diff of your current changes, and stores it away
- Allows you to temporarily switch to another branch for a quick bug fix etc.

More commands: git-archive

- git-archive creates a tarball from the source tree

```
git-archive --format=tar --prefix=pgsql/ master | gzip > pgsql-snapshot.tar.gz
```

git.postgresql.org

- Free hosting site for PostgreSQL-related projects
 - Including a mirror of the PostgreSQL CVS repository
- Write to gitadmin@git.postgresql.org for an account
- Volunteers needed!