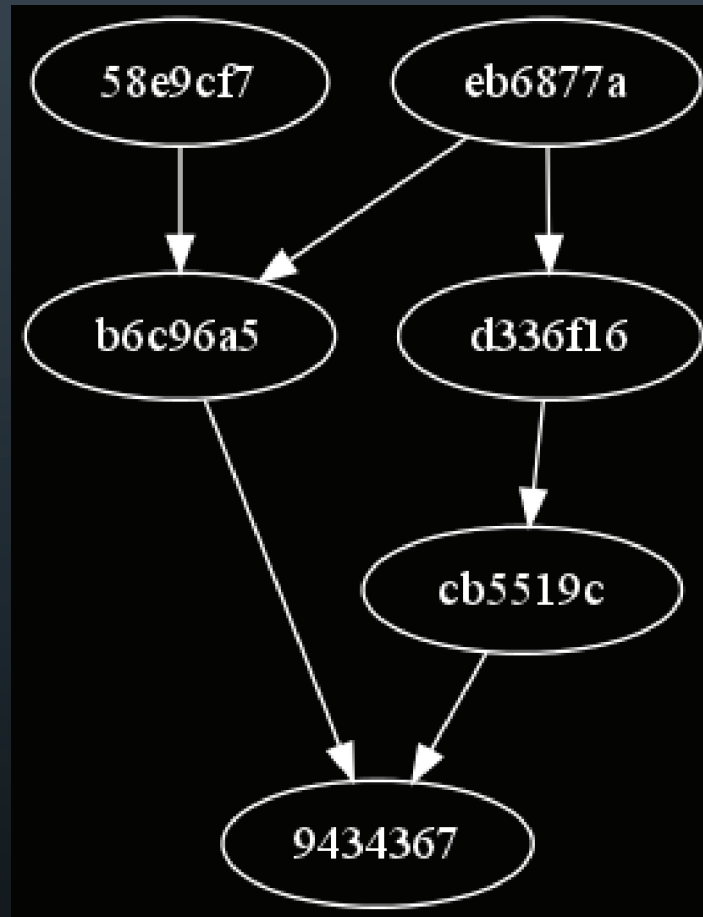


Fusing Git

Marco Fontani - Glasgow.pm - 2011-01-13

Git



```
~/.gitconfig  
[alias]
```

```
graphviz = "!f() { echo 'digraph git {' ; \  
    git log --pretty='format: %h -> { %p }' \  
    \"$@\" | \  
    sed 's/[0-9a-f][0-9a-f]*/\"&\"/g' ; \  
    echo '}' ; } ; f"
```

```
[okram@ferus test-repos]$ git log --pretty=oneline --decorate --graph --all
* 58e9cf7b98b46e04484bc01ed30d6bc6a245bf73 (tag: exe_changed, c_branch) Changed exe name
| * eb6877adc7c6e7258f4874cd3cc0e7e060edb6ee (all_merged) Merge branch 'pl_stuff' into all_merged
| | \
| / /
| * d336f168b4d5e6ed3bccbb560661c0c21978613c (pl_stuff) And a Perl file
| * cb5519cb67b02d9d577de6704a432cb4529685ec Some utf8 text
* | b6c96a565b63d53ec63cb3ee5919ea42b44e7936 (tag: c_branch_tag_1) hello world and compiled
| /
* 94343678fcda143b31053f4dad64dfede8d5f3cf (HEAD, master) Added stuff
[okram@ferus test-repos]$ import test-repos-text-black.png
```

Heads & Tags

HEADS

? parent

TAGS

-> commit

? signed

FUSE

Filesystem in Userspace

a fully functional

Filesystem

In _a Userspace

program, using the FUSE library

github.com

/dpavlin

/perl-fuse

github.com

/dpavlin

/perl-fuse

Writing a Filesystem in Perl

```
use Fuse;
```

```
Fuse::main(
```

```
    mountpoint
```

```
    => $mountpoint,
```

```
    # ...
```

```
);
```

```
sub statfs { ... }
```

```
sub getdir { ... }
```

...

getattr() , getdir() ,
open() , read() ,
close() , flush() ,
statfs() , write() ,
mknod() , mkdir() ,
unlink() , fsync() ,
truncate() , ...

$qx\{ \dots \}$

```
git add $thing
git add $thing/.keep
git branch
git cat-file -p $head:$file
git cat-file -s $head:$file
git cat-file -t $head:$file
git checkout $head
git checkout $orig_branch
git commit -m "updated $head file $thing" $thing
git commit $thing/.keep -m "Created directory $thing"
git commit $thing -m "Created empty file $thing"
git commit $thing -m "Removed $thing"
git ls-tree $head $file
git ls-tree $thing
git rm $thing
git tag
git tag -ln $head
git tag -v $head 2>&1
```

```
$ ls mnt/  
heads/  
tags/
```

```
$ ls mnt/heads/  
    master/  
    branch_a/  
    branch_b
```

```
$ ls \  
mnt/heads/master  
dir_a/  
file_a  
file_b
```

```
$ echo "hi" > \
mnt/heads/master
/new_file
```

```
Commit feed10beef  
"Created empty file  
  new_file"
```

```
Commit feed1decaf  
"Updated master file  
  new_file"
```

Entity	Thing	Read	Write	Create	Delete
Files	HEADs	Y	Y (a)	Y (b)	Y
Dirs	HEADs	Y	n/a	Y (c)	?
Files	TAGs	Y	N (d)	N (d)	N (d)
Dirs	TAGs	Y	N (d)	N (d)	N (d)

a) one file is committed at a time, on the tip of the HEAD

b) a new file is committed empty, then a) happens

c) a ".keep" file is created inside the new directory, and committed

d) Trees reachable from TAGs are read-only

are all **DEMO** es...

doomed to fail?

**Fruitful
train
journey**

github.com

/mfontani

/git-fuse-perl

github.com

/mfontani

/git-fuse-perl

patches

welcome

(snickers)

to: Marco Fontani <MFONTANI@cpan.org>
date: Sun, Jan 9, 2011 at 8:17 PM
subject: 2 patches, 2 feature requests

Mounting a bare Git repository read-only
[...]

your project makes this possible :)
... with the "following" 2 patches.

commit 4076ba7ee

Date: Sun Jan 9 21:17:15 2011 +0100

Allow running from any directory
of a bare or checked-out repository

It's not a bug
it's a FEATURE

It's not a bug
it's a FEATURE
REQUEST

Make symlinks work

```
$ ls -l RelNotes
```

```
/bin/ls: cannot read symbolic link RelNotes: Function  
not implemented
```

```
sub readlink { ... }
```

Mount a revision

read-only

no tags, no heads

on mountpoint/

Coming soon ...

Coming soonish...

github.com

/mfontani

/git-fuse-perl