Git Sucks

How git sucks and why you'll use it anyway

Intro

- David Whittington
- Self Employed
  - xForty
  - ParentProxy

Goals

- Convince you to at least try git
- Give you enough information to get started

Experience?

- What VCS experience do you have?
  - RCS?
  - CVS/SVN?
  - Perforce?
  - Git/hg/bazaar?

Version Control History

- In the beginning … RCS
  - Lock – modify – unlock
  - Single file
  - Makes collaboration difficult

Version Control History

- CVS/SVN
  - Copy – modify – merge
  - Central server
  - Makes forking + experimentation difficult

Version Control History

- Distributed version control
  - Clone – modify – commit – fetch – merge ...
  - Lots of workflow options
  - Examples: Git, Mercurial, Bazaar-ng, Monotone

What is Git?

- Distributed VCS system
- Created in 2005 by Torvalds following Bitkeeper fallout
- DAG based (as opposed to diff based)

Git Sucks

- DVCS = complexity
- Unintuitive
- Documentation also distributed
- Code can be obtuse

Why You'll Use it Anyway

- Distributed VCS is Good
  - Encourages contributions
  - Encourages experimentation (branches are easy)
  - Offline access rocks

Why You'll Use it Anyway

from http://blog.orebokech.com/2008/06/updated-debian-vcs-statistics.html
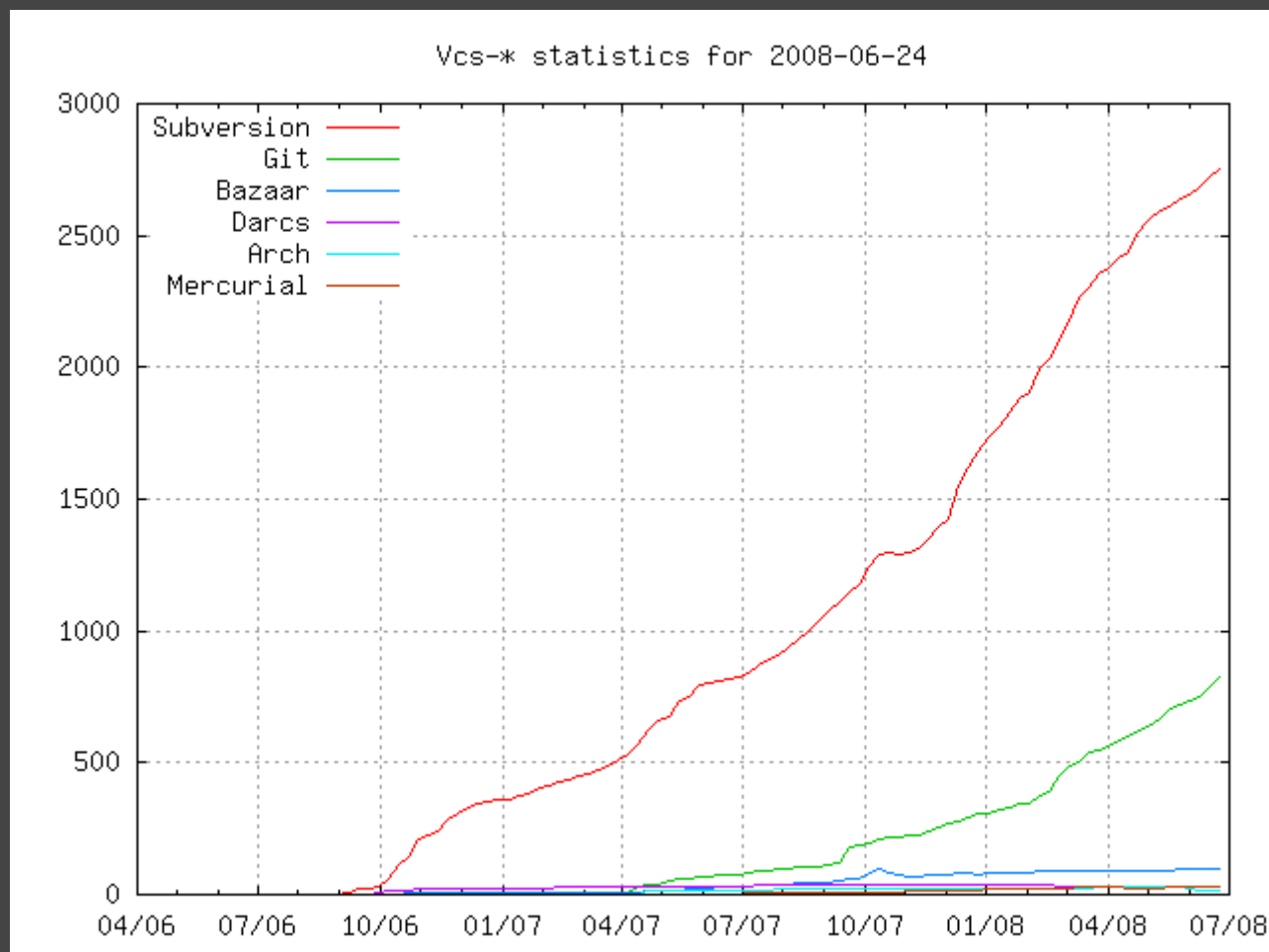
Why You'll Use it Anyway

- These projects are using it:
  - Linux Kernel
  - X.org
  - Samba
  - Wine
  - Ruby on Rails
  - ... more at http://git.or.cz/gitwiki/GitProjects

Why You'll Use it Anyway

- It really is powerful
  - Unixy
  - Fast
  - SVN bridge

Why You'll Use it Anyway

Github rocks!

Why You'll Use it Anyway

Hypnotoad Commands It!

Using Git - Configuration

```
$ git config --global user.name "David ..."
$ git config --global user.email "djwhitt@..."
$ git config --list
user.name="David Whittington"
user.email="djwhitt@gmail.com"
$ cat ~/.gitconfig
[user]
    name = David Whittington
    email = "djwhitt@gmail.com"
```

Using Git – Cloning a Repo

```
$ git clone git://github.com/rails/rails.git
$ cd rails
```

Using Git – Creating a Repo

```
$ mkdir deathray

$ cd deathray

$ git init
```

Using Git – Adding a File

```
$ echo "deathray - a ray that kills" > README
$ git add README
$ git commit -m "initial commit"
```

Using Git – Modifying a File

```
$ echo "most awesome deathray ever" >> README

$ git add README

$ git commit -m "added detail to README"
```


or …


```
$ echo "most awesome deathray ever" >> README

$ git commit -a -m "added detail to README"
```

Using Git – The Index

Object Dir   Index   Working Dir

Commits

Trees   ←   Blobs   ←   Files

Blobs
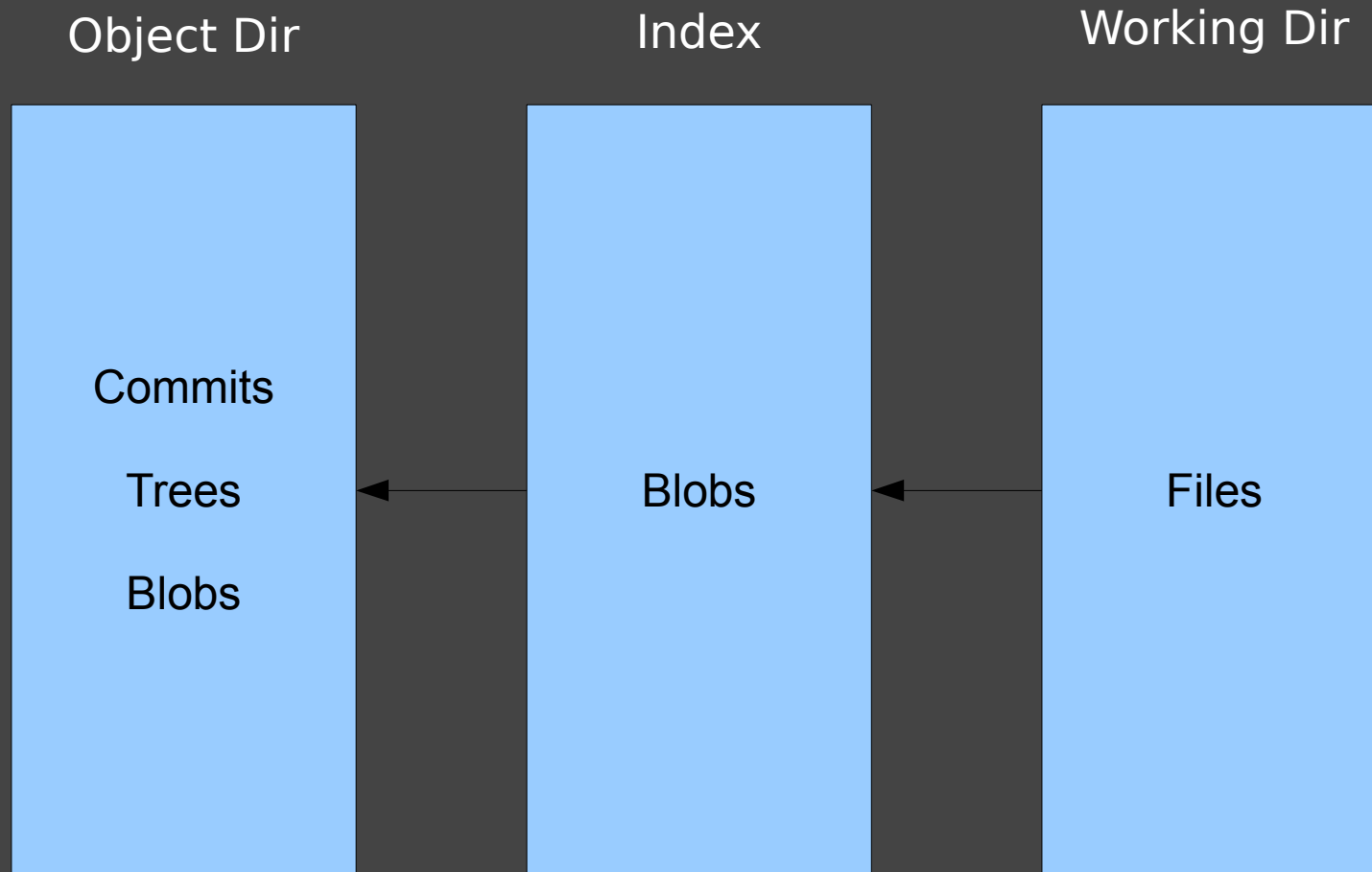
Using Git – Branches

```
$ git branch

* master

$ git branch fusion

$ git branch

  fusion

* master

$ git checkout fusion

$ git branch

* fusion

  master
```

Using Git – Branches

Using Git – Branches

```
$ echo "reactor core" > reactor
$ git add reactor
$ git commit -m "adding reactor"
$ ls
README reactor
$ git checkout master
$ ls
README
```

Using Git – Branches

Using Git - Merging

```
$ git branch
  fusion
* master
$ echo "focussing lens" > lens
$ git add lens
$ git commit -m "adding lens"
```

Using Git - Merging

Using Git - Merging

```
$ git merge fusion
```

Using Git - Merging

Using Git - Rebasing

```
$ git branch
  fusion
* master
$ echo "focussing lens" > lens
$ git add lens
$ git commit -m "adding lens"
```

Using Git - Rebasing

Using Git - Rebasing

```
$ git rebase fusion
```

Using Git - Rebasing

Using Git – Git Status

```
$ git status
# Changed but not updated:
#        modified:    lens
# Untracked files:
#        amplifier
```

Using Git – Git Log

```
$ git log
commit c23d31c9d34c7bf540a8e57f615f735f9ecdc2f2
Author: David Whittington <djwhitt@gmail.com>
Date:    Tue Oct 14 14:25:55 2008 -0400

        adding lens
...
```

Using Git - .gitignore

```
$ cat .gitignore
# swap files
*.swp
# temp dir
/tmp
```

Using Git - Aliases

```
$ git config --global alias.co checkout
$ git co fusion
```

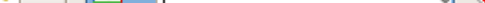# Using Git – Awesome Stuff

```
$ git add --interactive

$ git add --patch

$ git stash
```
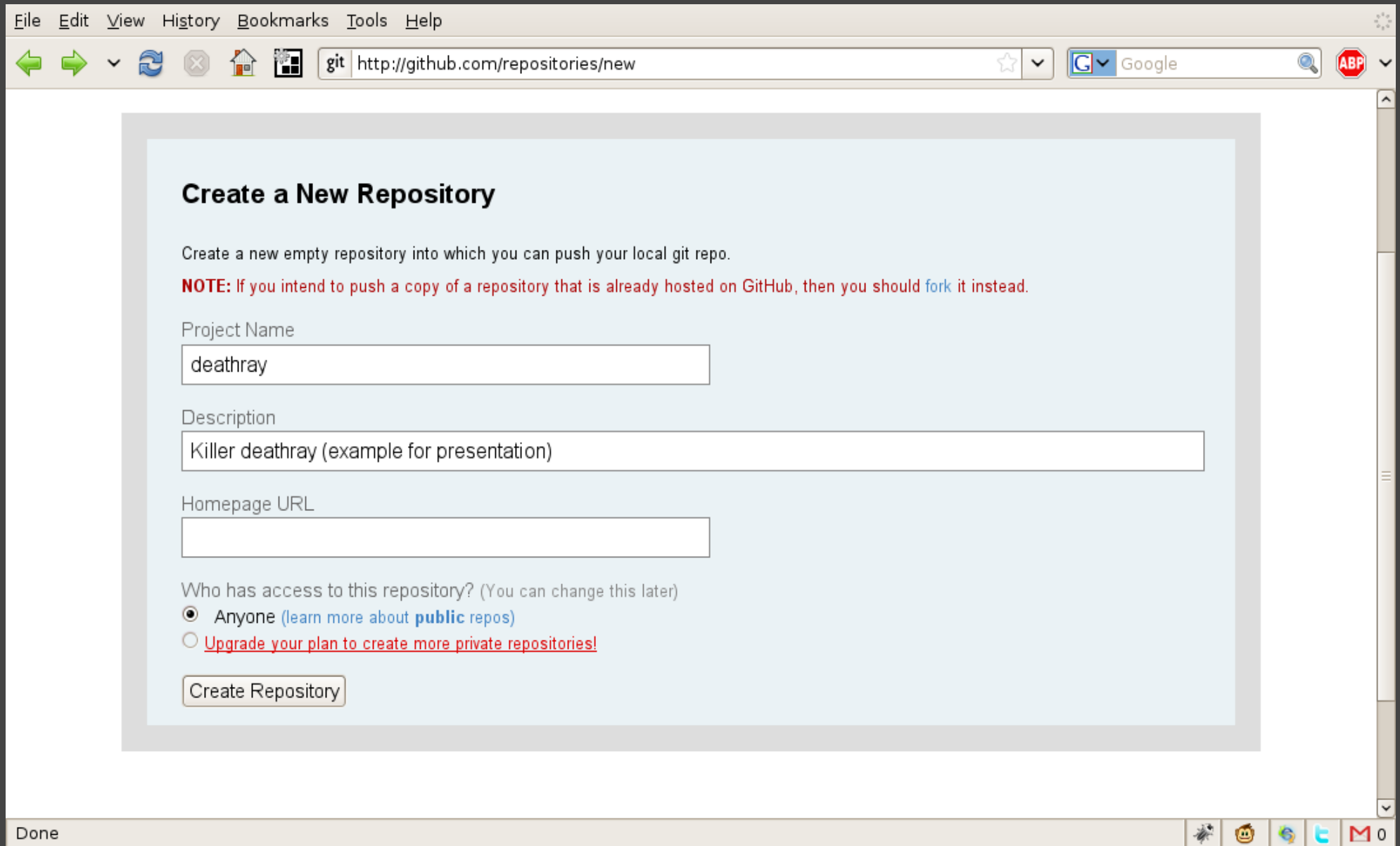
Using Git - Github

Using Git - Github

Using Git - Github

git http://github.com/djwhitt/deathray/tree/master   Google

Global setup:
```
Download and install Git
git config --global user.email djwhitt@gmail.com
```

Next steps:
```
mkdir deathray
cd deathray
git init
touch README
git add README
git commit -m 'first commit'
git remote add origin git@github.com:djwhitt/deathray.git
git push origin master
```

Existing Git Repo?
```
cd existing_git_repo
git remote add origin git@github.com:djwhitt/deathray.git
git push origin master
```

Importing a SVN Repo?

Done

Using Git – Remote + Push

```
$ git remote add github \
> git@github.com:djwhitt/deathray.git
$ git push github master
```

Using Git – Fetch, Merge, Pull

```
$ git fetch github
$ git merge github/master
```

or...

```
$ git pull github master
```

Using Git - Tracking

```
$ git config branch.master.remote github

$ git config branch.master.merge github/master
```

Stuff I Didn't Cover

- Workflow

- Conflicts

- Many more git commands

- Internals

- Other hosting options

Resources

- http://git.or.cz/index.html
- http://git-scm.com/
- http://www.gitcasts.com/
- http://peepcode.com/
  - screencast and pdf book
  - not free

Questions?