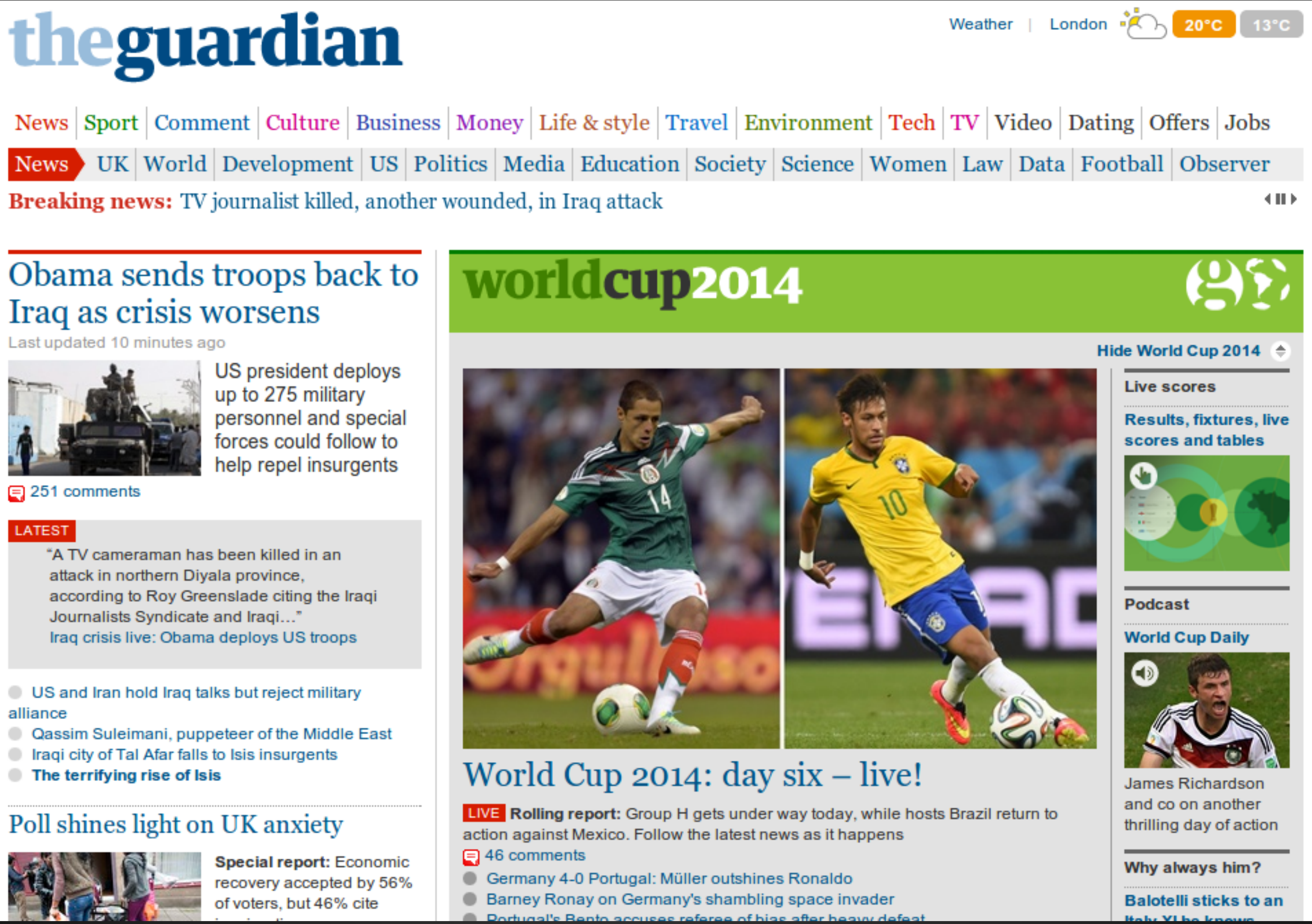
Git Going Faster… with Scala



theguardian

Weather | London 20°C 13°C

News | Sport | Comment | Culture | Business | Money | Life & style | Travel | Environment | Tech | TV | Video | Dating | Offers | Jobs

News | UK | World | Development | US | Politics | Media | Education | Society | Science | Women | Law | Data | Football | Observer

Breaking news: TV journalist killed, another wounded, in Iraq attack

Obama sends troops back to Iraq as crisis worsens

Last updated 10 minutes ago

US president deploys up to 275 military personnel and special forces could follow to help repel insurgents

251 comments

LATEST

"A TV cameraman has been killed in an attack in northern Diyala province, according to Roy Greenslade citing the Iraqi Journalists Syndicate and Iraqi…"

Iraq crisis live: Obama deploys US troops

- US and Iran hold Iraq talks but reject military alliance
- Qassim Suleimani, puppeteer of the Middle East
- Iraqi city of Tal Afar falls to Isis insurgents
- **The terrifying rise of Isis**

Poll shines light on UK anxiety

Special report: Economic recovery accepted by 56% of voters, but 46% cite

worldcup2014

Hide World Cup 2014

World Cup 2014: day six – live!

LIVE **Rolling report:** Group H gets under way today, while hosts Brazil return to action against Mexico. Follow the latest news as it happens

46 comments

- Germany 4-0 Portugal: Müller outshines Ronaldo
- Barney Ronay on Germany's shambling space invader
- Portugal's Bento accuses referee of bias after heavy defeat

Live scores

Results, fixtures, live scores and tables

Podcast

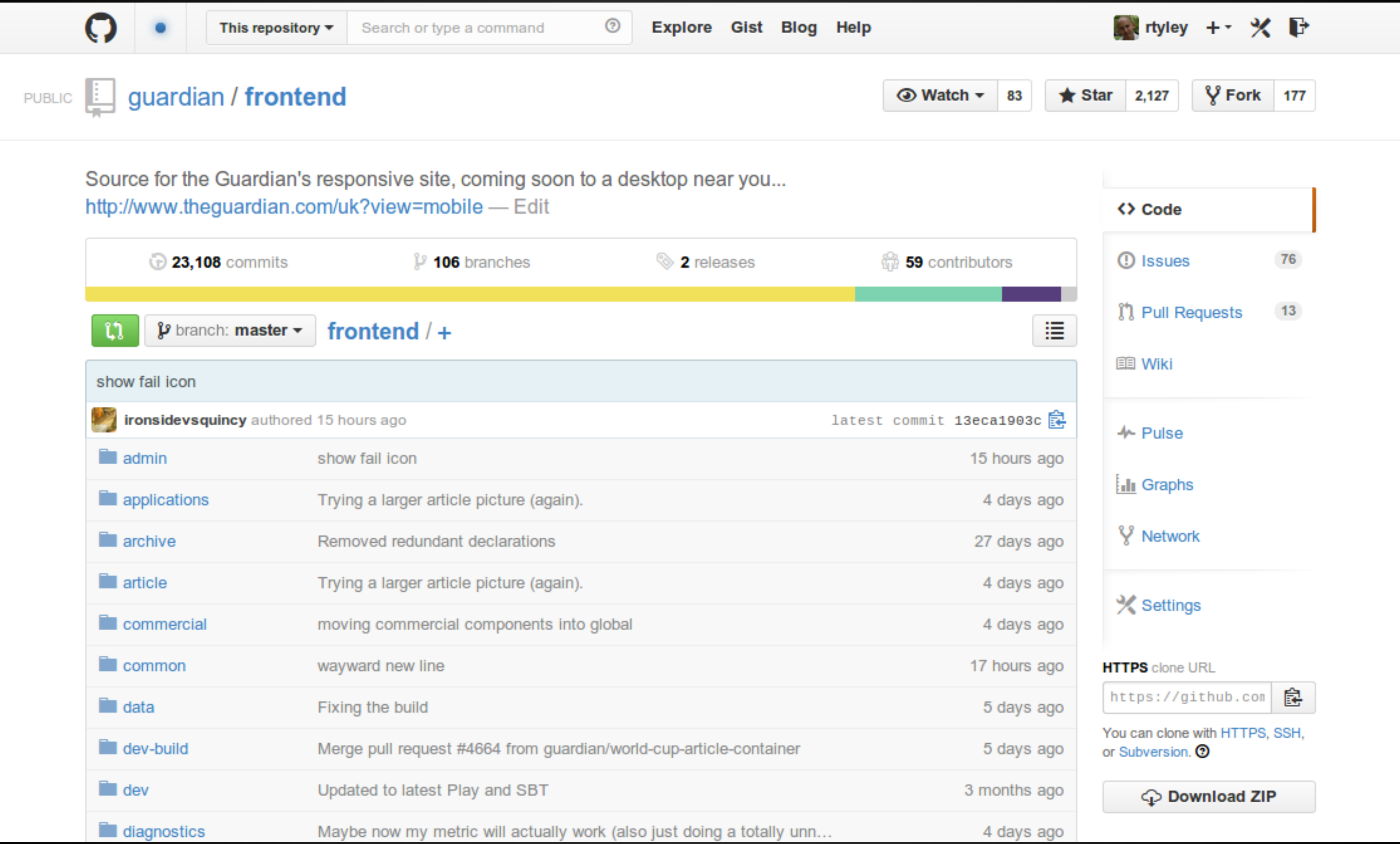
World Cup Daily

James Richardson and co on another thrilling day of action

Why always him?

Balotelli sticks to an Italy XI he knows

NSA somehow manages to exceed your most paranoid imaginings



guardian / frontend

Source for the Guardian's responsive site, coming soon to a desktop near you...
http://www.theguardian.com/uk?view=mobile — Edit

23,108 commits | 106 branches | 2 releases | 59 contributors

frontend / +

show fail icon

ironsidevsquincy authored 15 hours ago — latest commit 13eca1903c

admin	show fail icon	15 hours ago
applications	Trying a larger article picture (again).	4 days ago
archive	Removed redundant declarations	27 days ago
article	Trying a larger article picture (again).	4 days ago
commercial	moving commercial components into global	4 days ago
common	wayward new line	17 hours ago
data	Fixing the build	5 days ago
dev-build	Merge pull request #4664 from guardian/world-cup-article-container	5 days ago
dev	Updated to latest Play and SBT	3 months ago
diagnostics	Maybe now my metric will actually work (also just doing a totally unn…	4 days ago

HTTPS clone URL
https://github.com

You can clone with HTTPS, SSH, or Subversion.

Git

Git Going Faster… with Scala

How?

No wait, what?

Faster...

- Prototyping
- Performance

the problem...

Bad data in yer Git repo...

Crazy Big Files

&

Passwords, **Credentials** & other **Private data**

...lurking in your history

How big is BIG?

GitHub limits as of June 2013 :

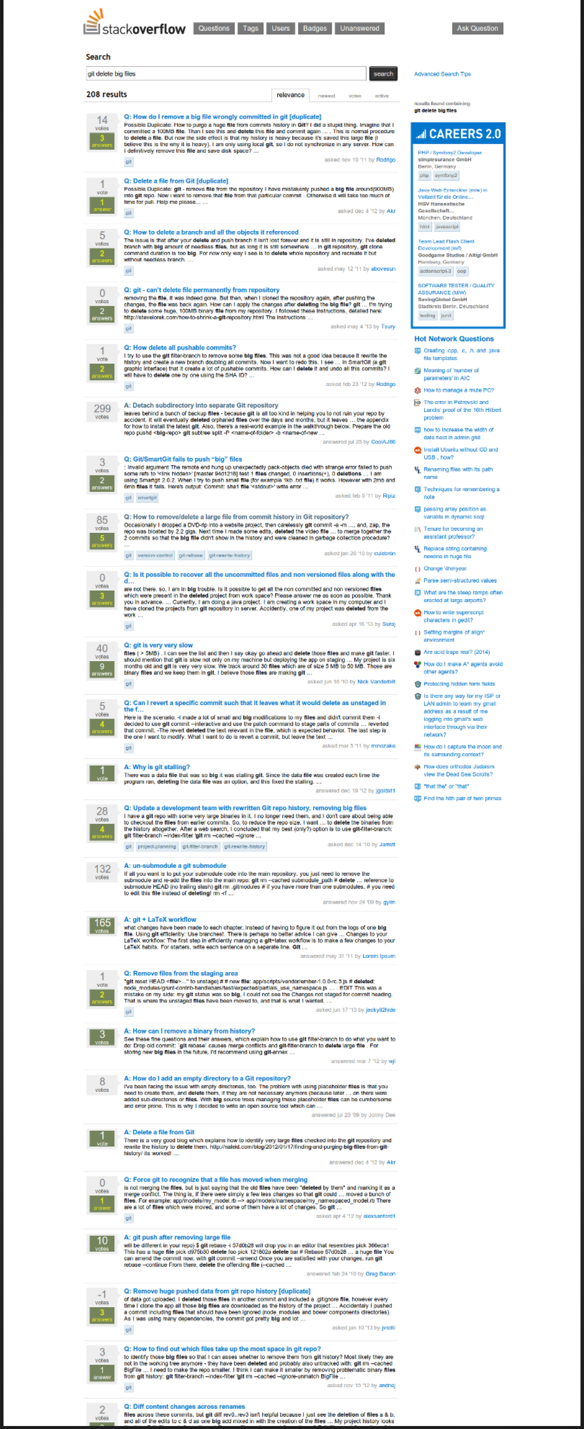
50MB - WARN

100MB - ERROR

How big is BIG?

5MB - BAD

10MB - BAAD



Search

git delete big files

search

Advanced Search Tips

208 results

relevance newest votes active

results found containing
git delete big files

14 votes 3 answers
Q: How do I remove a big file wrongly committed in git [duplicate]
Possible Duplicate: How to purge a huge file from commits history in Git? I did a stupid thing. Imagine that I committed a 100MB file. Than I see this and delete this file and commit again … . This is normal procedure to delete a file. But now the side effect is that my history is heavy because it's saved this large file (I believe this is the why it is heavy). I am only using local git, so I do not synchronize in any server. How can I definitively remove this file and save disk space? …
git
asked nov 10 '11 by Rodrigo

1 vote 1 answer
Q: Delete a file from Git [duplicate]
Possible Duplicate: git - remove file from the repository I have mistakenly pushed a big file around(900MB) into git repo. Now i want to remove that file from that particular commit . Otherwise it will take too much of time for pull. Help me please... …
git
asked dec 4 '12 by Akr

5 votes 2 answers
Q: How to delete a branch and all the objects it referenced
The issue is that after your delete and push branch it isn't lost forever and it is still in repository. I've deleted branch with big amount of needless files, but as long it is still somewhere … in git repository, git clone command duration is too big. For now only way I see is to delete whole repository and recreate it but without needless branch. …
git
asked may 12 '11 by abovesun

0 votes 2 answers
Q: git - can't delete file permanently from repository
removing the file. It was indeed gone. But then, when I cloned the repository again, after pushing the changes, the file was back again. How can I apply the changes after deleting the big file? git … I'm trying to delete some huge, 100MB binary file from my repository. I followed these instructions, detailed here: http://stevelorek.com/how-to-shrink-a-git-repository.html The instructions …
git
asked may 4 '13 by Tsury

1 vote 2 answers
Q: How delete all pushable commits?
I try to use the git filter-branch to remove some big files. This was not a good idea because it rewrite the history and create a new branch doubling all commits. Now I want to redo this. I see … In SmartGit (a git graphic interface) that it create a lot of pushable commits. How can I delete it and undo all this commits? I will have to delete one by one using the SHA ID? …
git
asked feb 23 '12 by Rodrigo

299 votes
A: Detach subdirectory into separate Git repository
leaves behind a bunch of backup files - because git is all too kind in helping you to not ruin your repo by accident. It will eventually deleted orphaned files over the days and months, but it leaves … the appendix for how to install the latest git. Also, there's a real-world example in the walkthrough below. Prepare the old repo pushd <big-repo> git subtree split -P <name-of-folder> -b <name-of-new …
answered jul 25 by CoolAJ86

3 votes 2 answers
Q: Git/SmartGit fails to push "big" files
: Invalid argument The remote end hung up unexpectedly pack-objects died with strange error failed to push some refs to '<link hidden>' [master 94d1216] test 1 files changed, 0 insertions(+), 0 deletions … I am using Smartgit 2.0.2. When I try to push small file (for example 1kb .txt file) it works. However with 2mb and 6mb files it fails. Here's output: Commit: sha1 file '<stdout>' write error …
git smartgit
asked feb 5 '11 by Ripiz

85 votes 5 answers
Q: How to remove/delete a large file from commit history in Git repository?
Occasionally I dropped a DVD-rip into a website project, then carelessly git commit -a -m ..., and, zap, the repo was bloated by 2.2 gigs. Next time I made some edits, deleted the video file … to merge together the 2 commits so that the big file didn't show in the history and were cleaned in garbage collection procedure? …
git version-control git-rebase git-rewrite-history
asked jan 20 '10 by culebrón

0 votes 3 answers
Q: Is it possible to recover all the uncommitted files and non versioned files along with the d…
are not there. so, I am in big trouble. Is it possible to get all the non committed and non versioned files which were present in the deleted project from work space? Please answer me as soon as possible. Thank you in advance. … Currently, I am doing a java project. I am creating a work space in my computer and I have cloned the projects from git repository in server. Accidently, one of my project was deleted from the work …
git
asked apr 16 '13 by Suraj

40 votes 9 answers
Q: git is very very slow
files (> 5MB) . I can see the list and then I say okay go ahead and delete those files and make git faster. I should mention that git is slow not only on my machine but deploying the app on staging … My project is six months old and git is very very slow. We track around 30 files which are of size 5 MB to 50 MB. Those are binary files and we keep them in git. I believe those files are making git …
git
asked jun 16 '10 by Nick Vanderbilt

5 votes 4 answers
Q: Can I revert a specific commit such that it leaves what it would delete as unstaged in the f…
Here is the scenario: -I made a lot of small and big modifications to my files and didn't commit them -I decided to use git commit --interactive and use the patch command to stage parts of commits … reverted that commit. -The revert deleted the text relevant in the file, which is expected behavior. The last step is the one I want to modify. What I want to do is revert a commit, but leave the text …
git
asked mar 5 '11 by minozake

1 vote
A: Why is git stalling?
There was a data file that was so big it was stalling git. Since the data file was created each time the program ran, deleting the data file was an option, and this fixed the stalling. …
answered dec 19 '12 by jgoldst1

28 votes 4 answers
Q: Update a development team with rewritten Git repo history, removing big files
I have a git repo with some very large binaries in it. I no longer need them, and I don't care about being able to checkout the files from earlier commits. So, to reduce the repo size, I want … to delete the binaries from the history altogether. After a web search, I concluded that my best (only?) option is to use git-filter-branch: git filter-branch --index-filter 'git rm --cached --ignore …
git project-planning git-filter-branch git-rewrite-history
asked dec 14 '10 by Jarrett

132 votes
A: un-submodule a git submodule
If all you want is to put your submodule code into the main repository, you just need to remove the submodule and re-add the files into the main repo: git rm --cached submodule_path # delete … reference to submodule HEAD (no trailing slash) git rm .gitmodules # if you have more than one submodules, # you need to edit this file instead of deleting! rm -rf …
answered nov 24 '09 by gyim

165 votes
A: git + LaTeX workflow
what changes have been made to each chapter, instead of having to figure it out from the logs of one big file. Using git efficiently: Use branches!. There is perhaps no better advice I can give … Changes to your LaTeX workflow: The first step in efficiently managing a git+latex workflow is to make a few changes to your LaTeX habits. For starters, write each sentence on a separate line. Git …
answered may 31 '11 by Lorem Ipsum

1 vote 2 answers
Q: Remove files from the staging area
"git reset HEAD <file>..." to unstage) # # new file: app/scripts/vendor/ember-1.0.0-rc.5.js # deleted: node_modules/grunt-contrib-handlebars/test/expected/partials_use_namespace.js … . EDIT This was a mistake on my side: my git status was so big, I could not see the Changes not staged for commit heading. That is where the unstaged files have been moved to, and that is what I wanted. …
git
asked jun 17 '13 by jeckyll2hide

3 votes
A: How can I remove a binary from history?
See these fine questions and their answers, which explain how to use git filter-branch to do what you want to do: Drop old commit: `git rebase` causes merge conflicts and git-filter-branch to delete large file . For storing new big files in the future, I'd recommend using git-annex …
answered mar 7 '12 by wjl

8 votes
A: How do I add an empty directory to a Git repository?
I've been facing the issue with empty directories, too. The problem with using placeholder files is that you need to create them, and delete them, if they are not necessary anymore (because later … on there were added sub-directories or files. With big source trees managing these placeholder files can be cumbersome and error prone. This is why I decided to write an open source tool which can …
answered jul 23 '09 by Jonny Dee

1 vote
A: Delete a file from Git
There is a very good blog which explains how to identify very large files checked into the git repository and rewrite the history to delete them. http://naleid.com/blog/2012/01/17/finding-and-purging-big-files-from-git-history/ its worked! …
answered dec 4 '12 by Akr

0 votes 1 answer
Q: Force git to recognize that a file has moved when merging
is not merging the files, but is just saying that the old files have been "deleted by them" and marking it as a merge conflict. The thing is, if there were simply a few less changes so that git could … moved a bunch of files. For example: app/models/my_model.rb --> app/models/namespace/my_namespaced_model.rb There are a lot of files which were moved, and some of them have a lot of changes. So git …
git
asked apr 4 '12 by alexsanford1

10 votes
A: git push after removing large file
will be different in your repo) $ git rebase -i 57d0b28 will drop you in an editor that resembles pick 366eca1 This has a huge file pick d975b30 delete foo pick 121802a delete bar # Rebase 57d0b28 … a huge file You can amend the commit now, with git commit --amend Once you are satisfied with your changes, run git rebase --continue From there, delete the offending file (--cached …
answered feb 24 '10 by Greg Bacon

-1 votes 3 answers
Q: Remove huge pushed data from git repo history [duplicate]
of data got uploaded. I deleted those files in another commit and included a .gitignore file, however every time I clone the app all those big files are downloaded as the history of the project … Accidentaly I pushed a commit including files that should have been ignored (node_modules and bower components directories). As I was using many dependencies, the commit got pretty big and lot …
git
asked jan 10 '13 by jviotti

3 votes 1 answer
Q: How to find out which files take up the most space in git repo?
to identify those big files so that I can asses whether to remove them from git history? Most likely they are not in the working tree anymore - they have been deleted and probably also untracked with: git rm --cached BigFile … I need to make the repo smaller. I think I can make it smaller by removing problematic binary files from git history: git filter-branch --index-filter 'git rm --cached --ignore-unmatch BigFile …
git
asked nov 15 '12 by andrej

2 votes
Q: Diff content changes across renames
files across these commits, but git diff rev0..rev3 isn't helpful because I just see the deletion of files a & b, and all of the edits to c & d as one big add mixed in with the creation of the files … My project history looks

CAREERS 2.0

PHP / Symfony2 Developer
simplesurance GmbH
Berlin, Germany
php symfony2

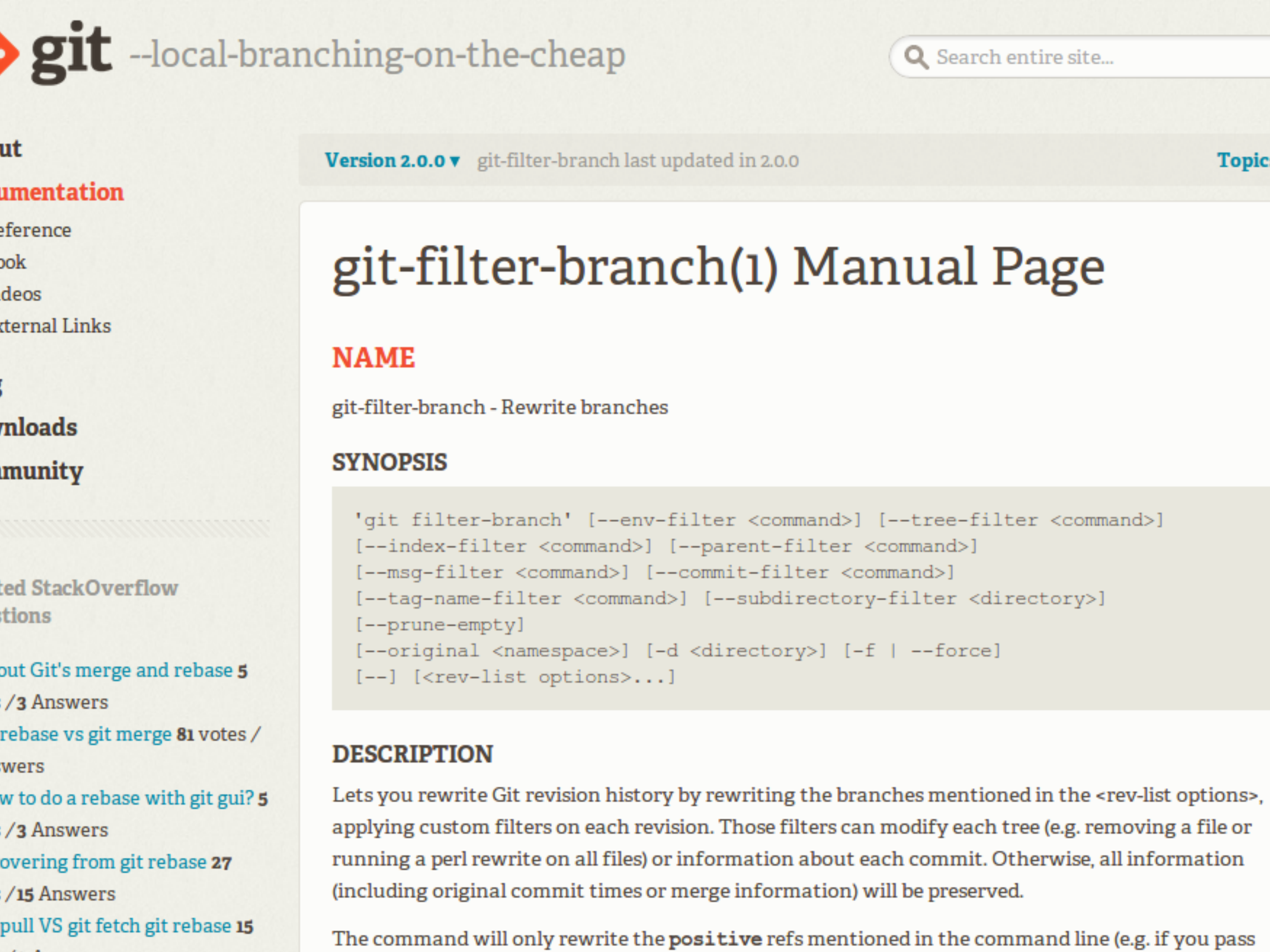
Java-Web-Entwickler (m/w) in Vollzeit für die Online…
HGV Hanseatische Gesellschaft…
München, Deutschland
html javascript

Team Lead Flash Client Development (m/f)
Goodgame Studios / Altigi GmbH
Hamburg, Germany
actionscript-3 oop

SOFTWARE TESTER / QUALITY ASSURANCE (M/W)
SavingGlobal GmbH
Stadtkreis Berlin, Deutschland
testing junit

Hot Network Questions

- Creating .cpp, .c, .h, and .java file templates
- Meaning of 'number of parameters' in AIC
- How to manage a mute PC?
- The error in Petrovski and Landis' proof of the 16th Hilbert problem
- how to increase the width of date field in admin grid
- Install Ubuntu without CD and USB , how?
- Renaming files with its path name
- Techniques for remembering a note
- passing array position as variable in dynamic soql
- Tenure for becoming an assistant professor?
- Replace string containing newline in huge file
- Change \the\year
- Parse semi-structured values
- What are the steep ramps often erected at large airports?
- How to write superscript characters in gedit?
- Setting margins of align* environment
- Are acid traps real? (2014)
- How do I make A* agents avoid other agents?
- Protecting hidden form fields
- Is there any way for my ISP or LAN admin to learn my gmail address as a result of me logging into gmail's web interface through via their network?
- How do I capture the moon and its surrounding context?
- How does orthodox Judaism view the Dead Sea Scrolls?
- "that the" or "that"
- Find the Nth pair of twin primes


git --local-branching-on-the-cheap
Search entire site...
ut
umentation
eference
ook
deos
xternal Links
nloads
munity
ted StackOverflow
tions
out Git's merge and rebase 5
/3 Answers
rebase vs git merge 81 votes /
wers
w to do a rebase with git gui? 5
/3 Answers
overing from git rebase 27
/15 Answers
pull VS git fetch git rebase 15
Version 2.0.0 ▾
git-filter-branch last updated in 2.0.0
Topic

git-filter-branch(1) Manual Page

NAME

git-filter-branch - Rewrite branches

SYNOPSIS

```
'git filter-branch' [--env-filter <command>] [--tree-filter <command>]
[--index-filter <command>] [--parent-filter <command>]
[--msg-filter <command>] [--commit-filter <command>]
[--tag-name-filter <command>] [--subdirectory-filter <directory>]
[--prune-empty]
[--original <namespace>] [-d <directory>] [-f | --force]
[--] [<rev-list options>...]
```

DESCRIPTION

Lets you rewrite Git revision history by rewriting the branches mentioned in the <rev-list options>, applying custom filters on each revision. Those filters can modify each tree (e.g. removing a file or running a perl rewrite on all files) or information about each commit. Otherwise, all information (including original commit times or merge information) will be preserved.

The command will only rewrite the `positive` refs mentioned in the command line (e.g. if you pass

As a user, the problems with git filter-branch:

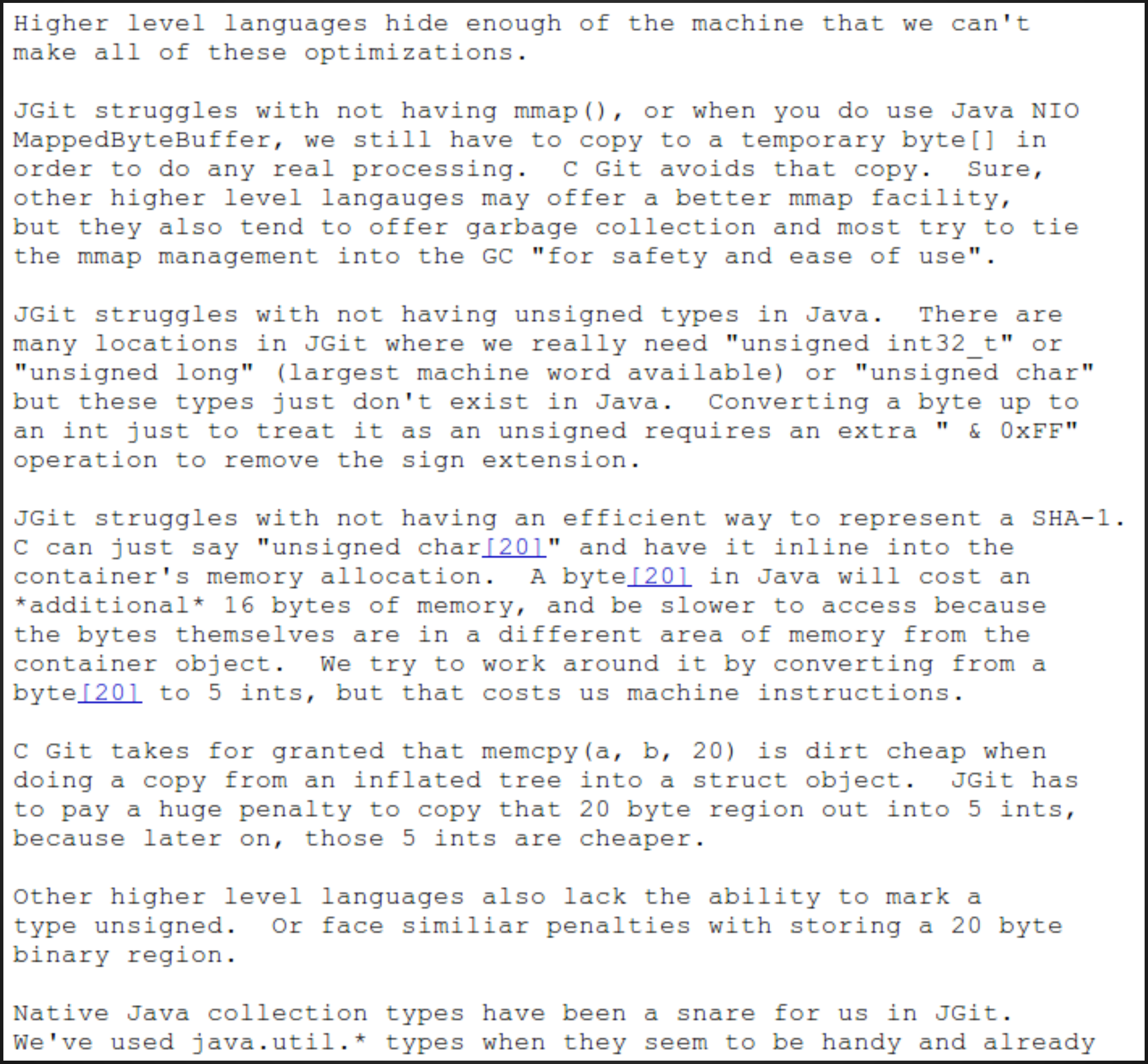
1. It's difficult to use
2. It's slow

a Scala version of git filter-branch?

JGit

(pure Java version of Git)

JGit in 2009...



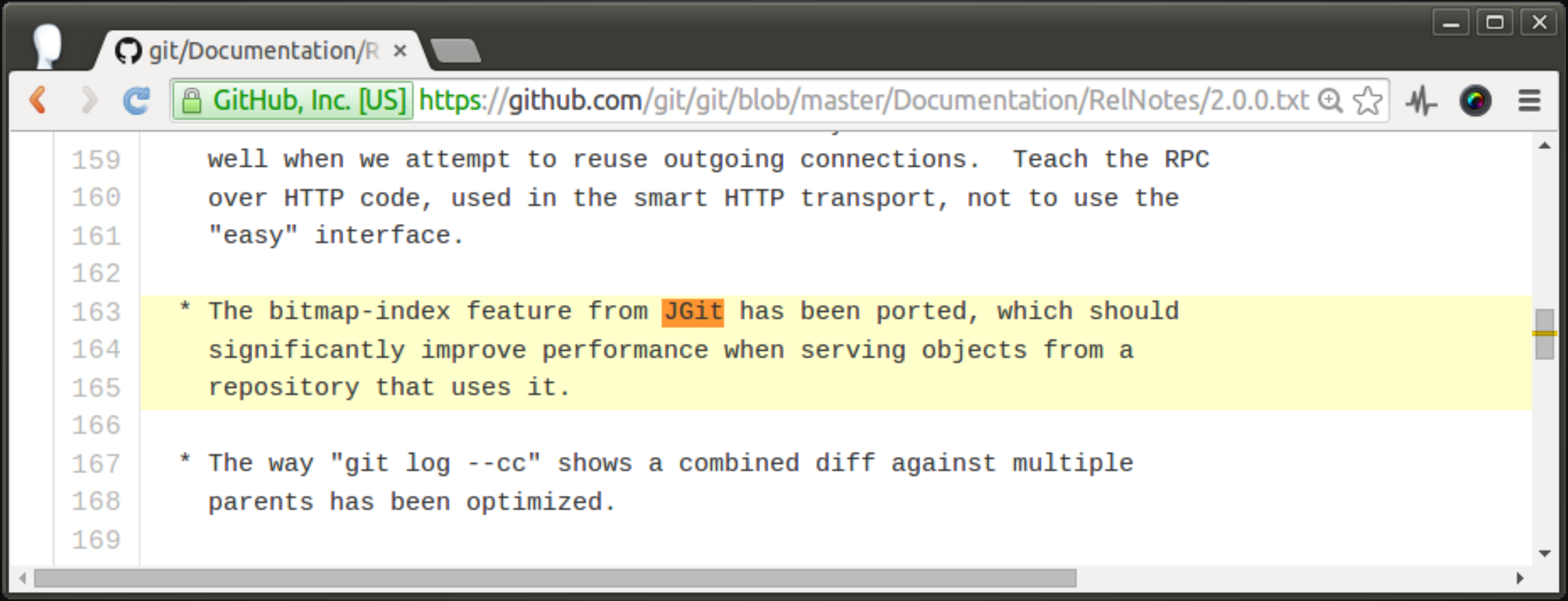
```
Higher level languages hide enough of the machine that we can't
make all of these optimizations.

JGit struggles with not having mmap(), or when you do use Java NIO
MappedByteBuffer, we still have to copy to a temporary byte[] in
order to do any real processing.  C Git avoids that copy.  Sure,
other higher level langauges may offer a better mmap facility,
but they also tend to offer garbage collection and most try to tie
the mmap management into the GC "for safety and ease of use".

JGit struggles with not having unsigned types in Java.  There are
many locations in JGit where we really need "unsigned int32_t" or
"unsigned long" (largest machine word available) or "unsigned char"
but these types just don't exist in Java.  Converting a byte up to
an int just to treat it as an unsigned requires an extra " & 0xFF"
operation to remove the sign extension.

JGit struggles with not having an efficient way to represent a SHA-1.
C can just say "unsigned char[20]" and have it inline into the
container's memory allocation.  A byte[20] in Java will cost an
*additional* 16 bytes of memory, and be slower to access because
the bytes themselves are in a different area of memory from the
container object.  We try to work around it by converting from a
byte[20] to 5 ints, but that costs us machine instructions.

C Git takes for granted that memcpy(a, b, 20) is dirt cheap when
doing a copy from an inflated tree into a struct object.  JGit has
to pay a huge penalty to copy that 20 byte region out into 5 ints,
because later on, those 5 ints are cheaper.

Other higher level languages also lack the ability to mark a
type unsigned.  Or face similiar penalties with storing a 20 byte
binary region.

Native Java collection types have been a snare for us in JGit.
We've used java.util.* types when they seem to be handy and already
```

http://marc.info/?l=git&m=124111702609723

JGit in 2011...

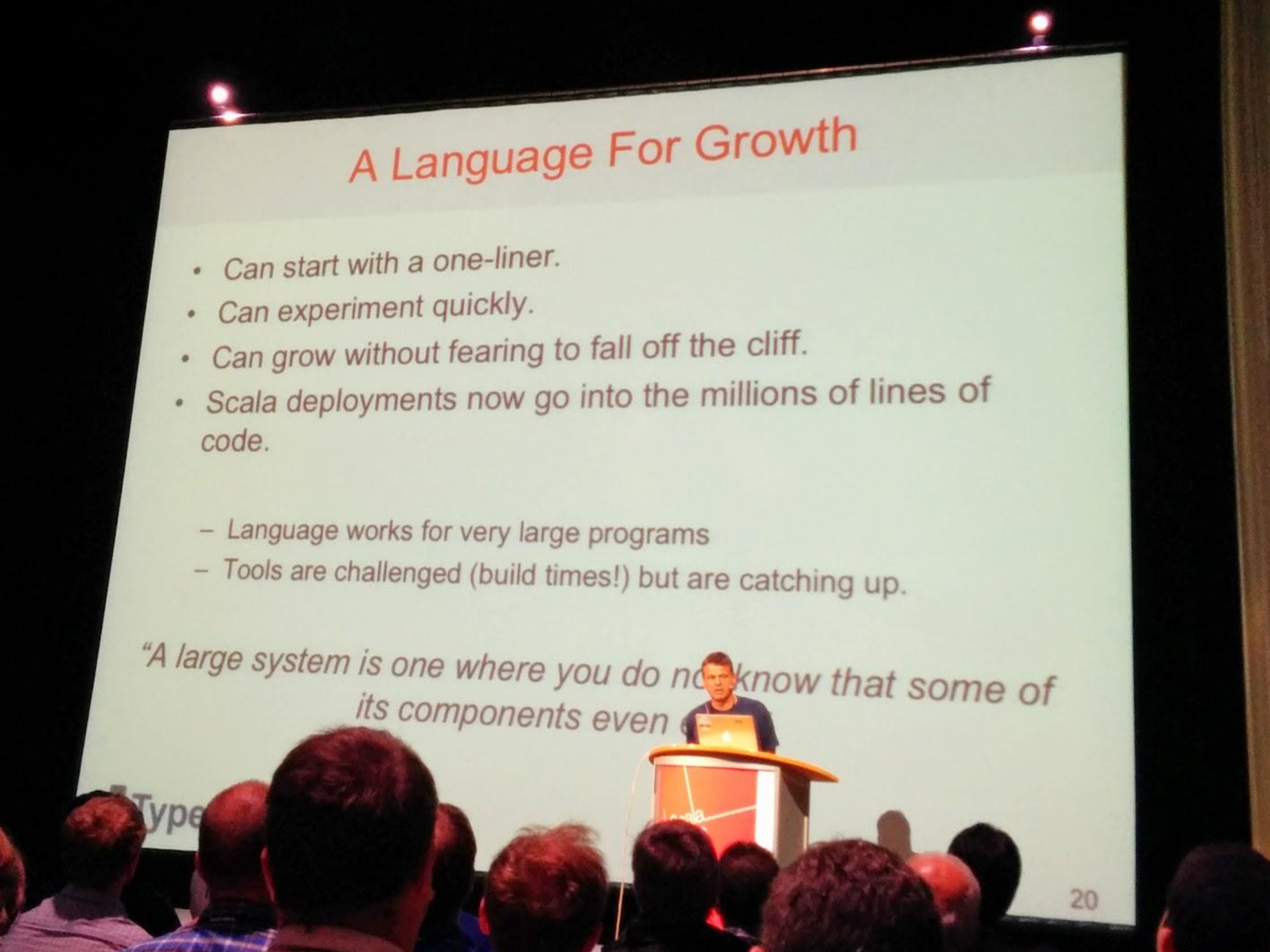
- Served 36TiB of Ice Cream Sandwich on https://android.googlesource.com
- Ran on mobile phones! (Agit)

JGit in 2014...


git/Documentation/R
GitHub, Inc. [US] https://github.com/git/git/blob/master/Documentation/RelNotes/2.0.0.txt

```
   well when we attempt to reuse outgoing connections.  Teach the RPC
   over HTTP code, used in the smart HTTP transport, not to use the
   "easy" interface.

 * The bitmap-index feature from JGit has been ported, which should
   significantly improve performance when serving objects from a
   repository that uses it.

 * The way "git log --cc" shows a combined diff against multiple
   parents has been optimized.
```

Cool! We can do this!



A Language For Growth

- Can start with a one-liner.
- Can experiment quickly.
- Can grow without fearing to fall off the cliff.
- Scala deployments now go into the millions of lines of code.

  - Language works for very large programs
  - Tools are challenged (build times!) but are catching up.

"A large system is one where you do no know that some of its components even

Type
Scala
20

As a user, the problems with git filter-branch:

1. It's difficult to use
2. It's slow

As a user, the problems with git filter-branch:

1. It's difficult to use
2. It's **slow**

1 West Ruislip
2 West Ruislip
nor delays due to ongoing signalling problem
Westbound platform 1
510
91081

git filter-branch lets you do…

anything…

to a Git repository

for us this is

too much power,

at too high a price

But 95% of the time...

…but git filter-branch is used for exactly one purpose:

removing unwanted data

(big files or passwords)

identical content in Git is stored exactly

ONCE

let’s clean it

ONCE

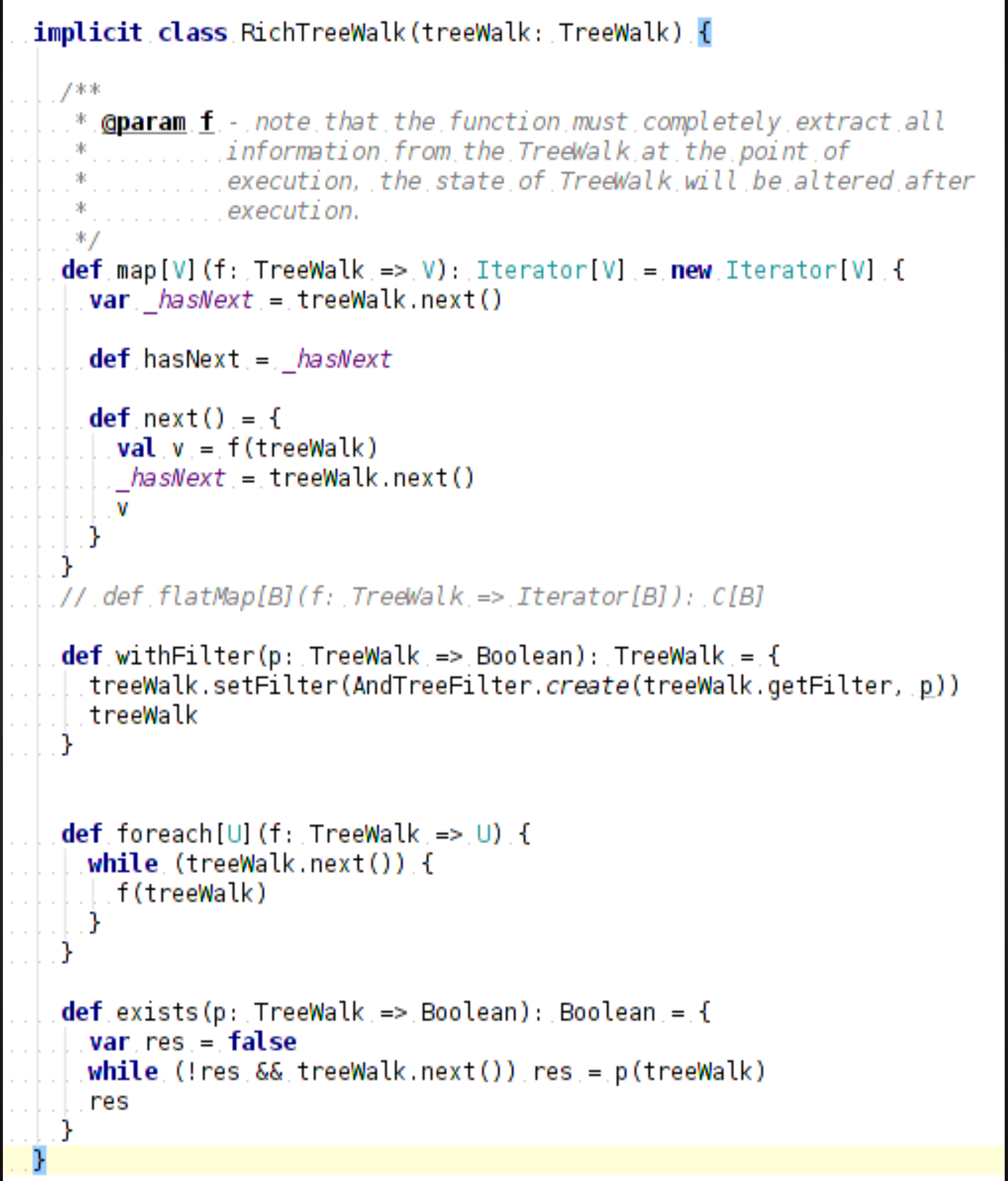
all we need is a map...

ObjectId -> ObjectId

(dirty) (clean)

growing with concurrent updates, but otherwise immutable

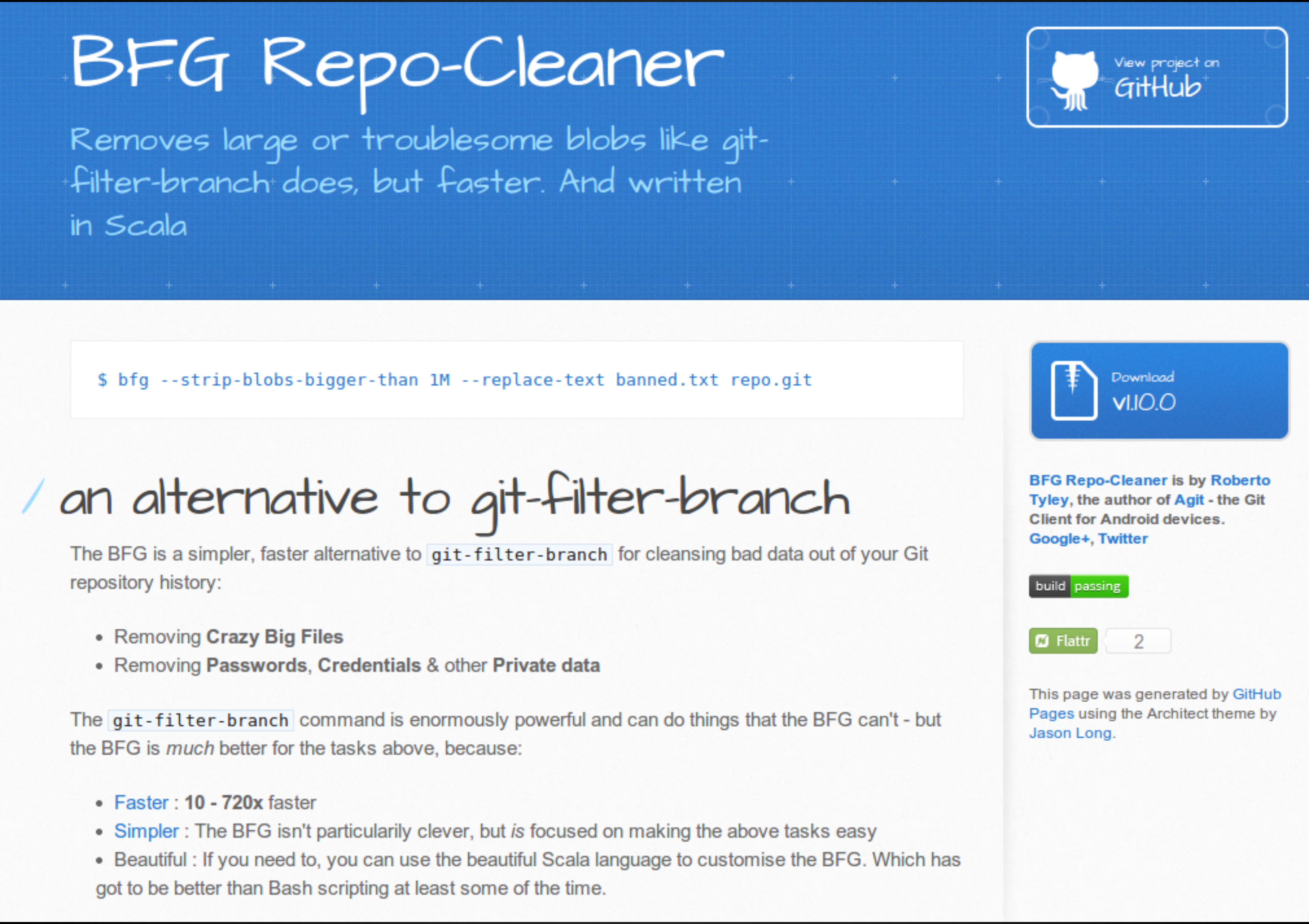
- Prototype with Scala, and *scale-up*
- Git runs pretty damn fast on the JVM



```scala
implicit class RichTreeWalk(treeWalk: TreeWalk) {

  /**
   * @param f - note that the function must completely extract all
   *          information from the TreeWalk at the point of
   *          execution, the state of TreeWalk will be altered after
   *          execution.
   */
  def map[V](f: TreeWalk => V): Iterator[V] = new Iterator[V] {
    var _hasNext = treeWalk.next()

    def hasNext = _hasNext

    def next() = {
      val v = f(treeWalk)
      _hasNext = treeWalk.next()
      v
    }
  }
  // def flatMap[B](f: TreeWalk => Iterator[B]): C[B]

  def withFilter(p: TreeWalk => Boolean): TreeWalk = {
    treeWalk.setFilter(AndTreeFilter.create(treeWalk.getFilter, p))
    treeWalk
  }


  def foreach[U](f: TreeWalk => U) {
    while (treeWalk.next()) {
      f(treeWalk)
    }
  }

  def exists(p: TreeWalk => Boolean): Boolean = {
    var res = false
    while (!res && treeWalk.next()) res = p(treeWalk)
    res
  }
}
```

The BFG

a simple, fast tool for cleansing *bad* data out of your Git repo history



BFG Repo-Cleaner

Removes large or troublesome blobs like git-filter-branch does, but faster. And written in Scala

View project on GitHub

```
$ bfg --strip-blobs-bigger-than 1M --replace-text banned.txt repo.git
```

Download v1.10.0

an alternative to git-filter-branch

The BFG is a simpler, faster alternative to `git-filter-branch` for cleansing bad data out of your Git repository history:

- Removing **Crazy Big Files**
- Removing **Passwords**, **Credentials** & other **Private data**

The `git-filter-branch` command is enormously powerful and can do things that the BFG can't - but the BFG is *much* better for the tasks above, because:

- Faster : **10 - 720x** faster
- Simpler : The BFG isn't particularily clever, but *is* focused on making the above tasks easy
- Beautiful : If you need to, you can use the beautiful Scala language to customise the BFG. Which has got to be better than Bash scripting at least some of the time.

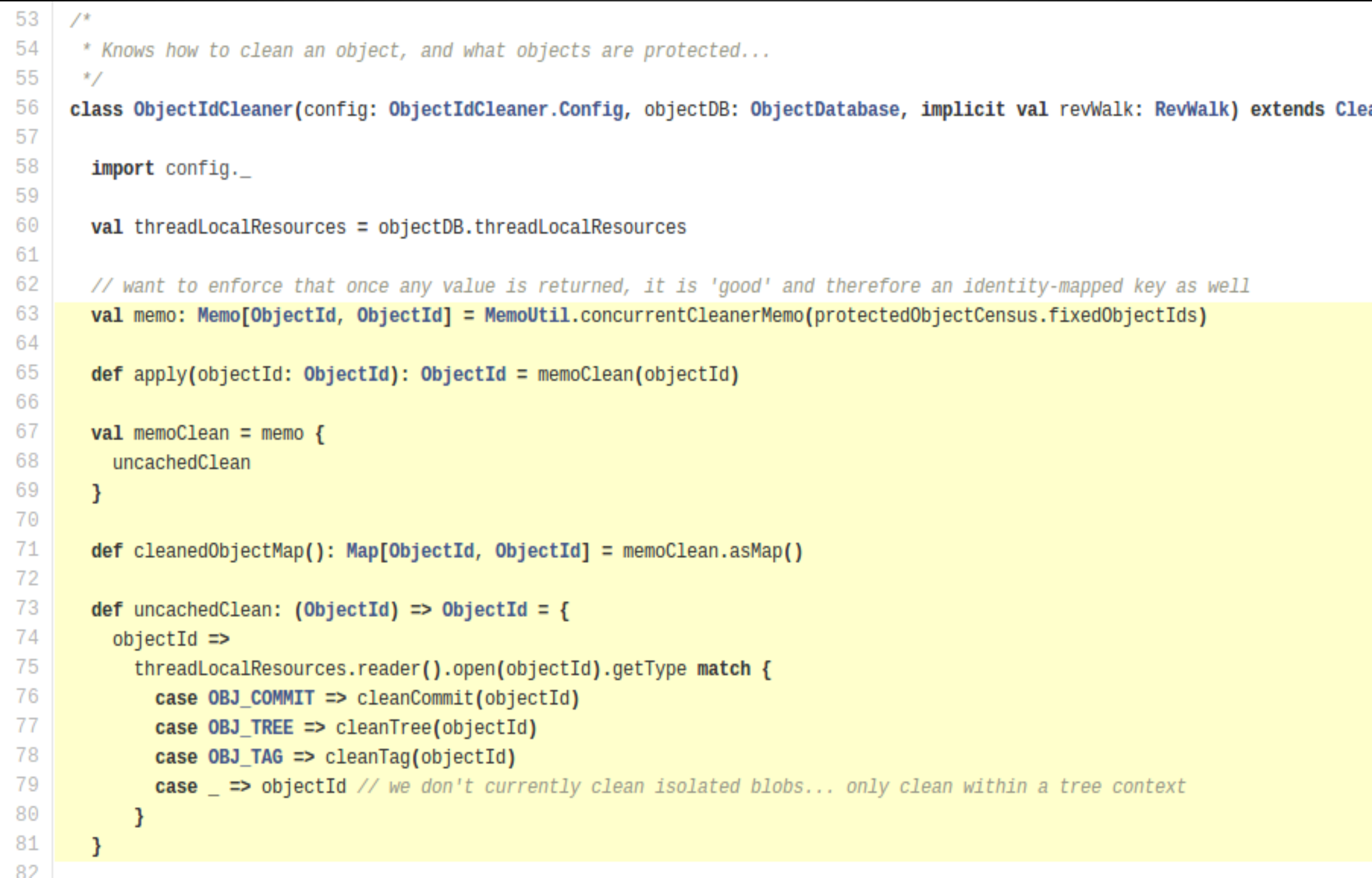
BFG Repo-Cleaner is by Roberto Tyley, the author of Agit - the Git Client for Android devices. Google+, Twitter

build passing

Flattr 2

This page was generated by GitHub Pages using the Architect theme by Jason Long.

This is The BFG



```
/*
 * Knows how to clean an object, and what objects are protected...
 */
class ObjectIdCleaner(config: ObjectIdCleaner.Config, objectDB: ObjectDatabase, implicit val revWalk: RevWalk) extends Clea

  import config._

  val threadLocalResources = objectDB.threadLocalResources

  // want to enforce that once any value is returned, it is 'good' and therefore an identity-mapped key as well
  val memo: Memo[ObjectId, ObjectId] = MemoUtil.concurrentCleanerMemo(protectedObjectCensus.fixedObjectIds)

  def apply(objectId: ObjectId): ObjectId = memoClean(objectId)

  val memoClean = memo {
    uncachedClean
  }

  def cleanedObjectMap(): Map[ObjectId, ObjectId] = memoClean.asMap()

  def uncachedClean: (ObjectId) => ObjectId = {
    objectId =>
      threadLocalResources.reader().open(objectId).getType match {
        case OBJ_COMMIT => cleanCommit(objectId)
        case OBJ_TREE => cleanTree(objectId)
        case OBJ_TAG => cleanTag(objectId)
        case _ => objectId // we don't currently clean isolated blobs... only clean within a tree context
      }
  }

```

Simpler to use...

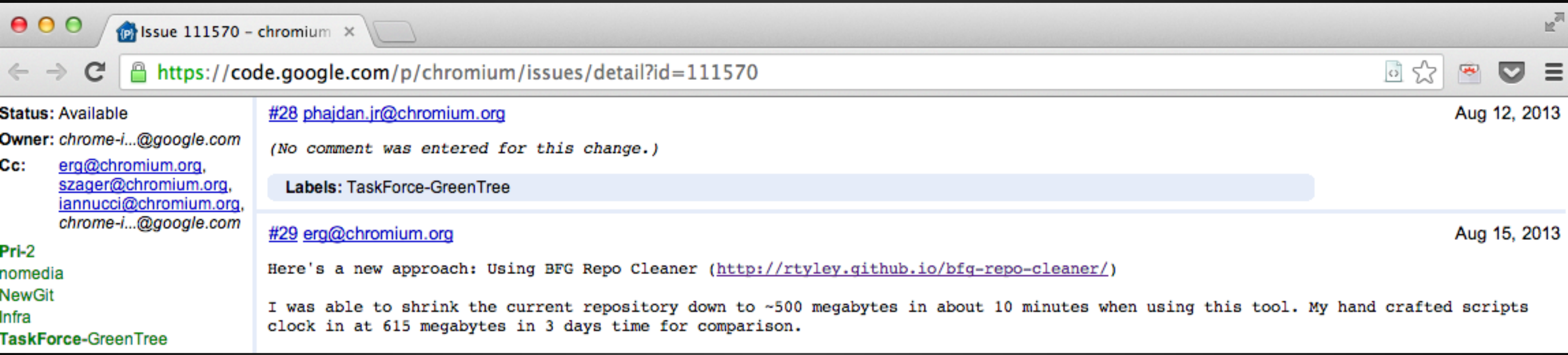
git-filter-branch:

```
$ git filter-branch --index-filter 'git rm
--cached --ignore-unmatch Rakefile' --tag-
name-filter cat -- --all
```

The BFG:

```
$ bfg -D Rakefile
```

Google Chrome



Status: Available
Owner: chrome-i...@google.com
Cc: erg@chromium.org, szager@chromium.org, iannucci@chromium.org, chrome-i...@google.com
Pri-2
nomedia
NewGit
Infra
TaskForce-GreenTree

#28 phajdan.jr@chromium.org Aug 12, 2013

(No comment was entered for this change.)

Labels: TaskForce-GreenTree

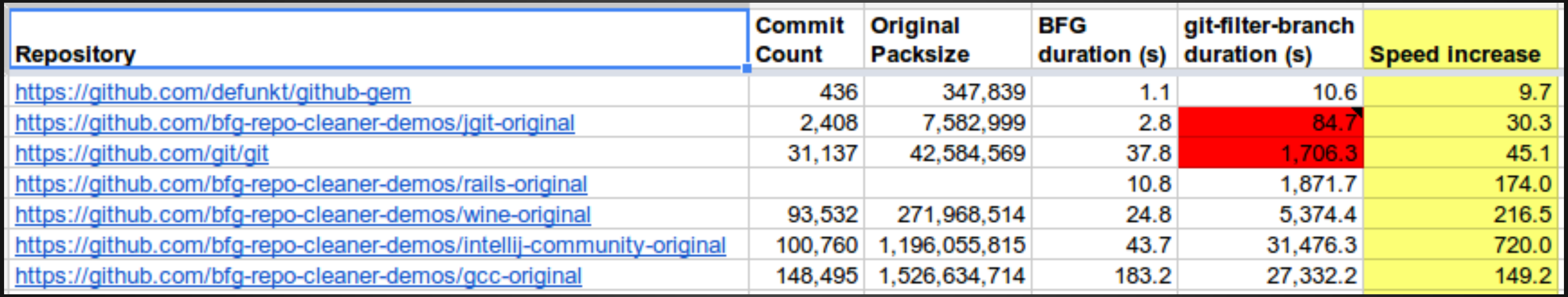
#29 erg@chromium.org Aug 15, 2013

```
Here's a new approach: Using BFG Repo Cleaner (http://rtyley.github.io/bfg-repo-cleaner/)

I was able to shrink the current repository down to ~500 megabytes in about 10 minutes when using this tool. My hand crafted scripts
clock in at 615 megabytes in 3 days time for comparison.
```

"...I was able to shrink the current repository down to ~500 megabytes in about **10 minutes** using this tool. My hand crafted scripts clock in at 615 megabytes in **3 days**"

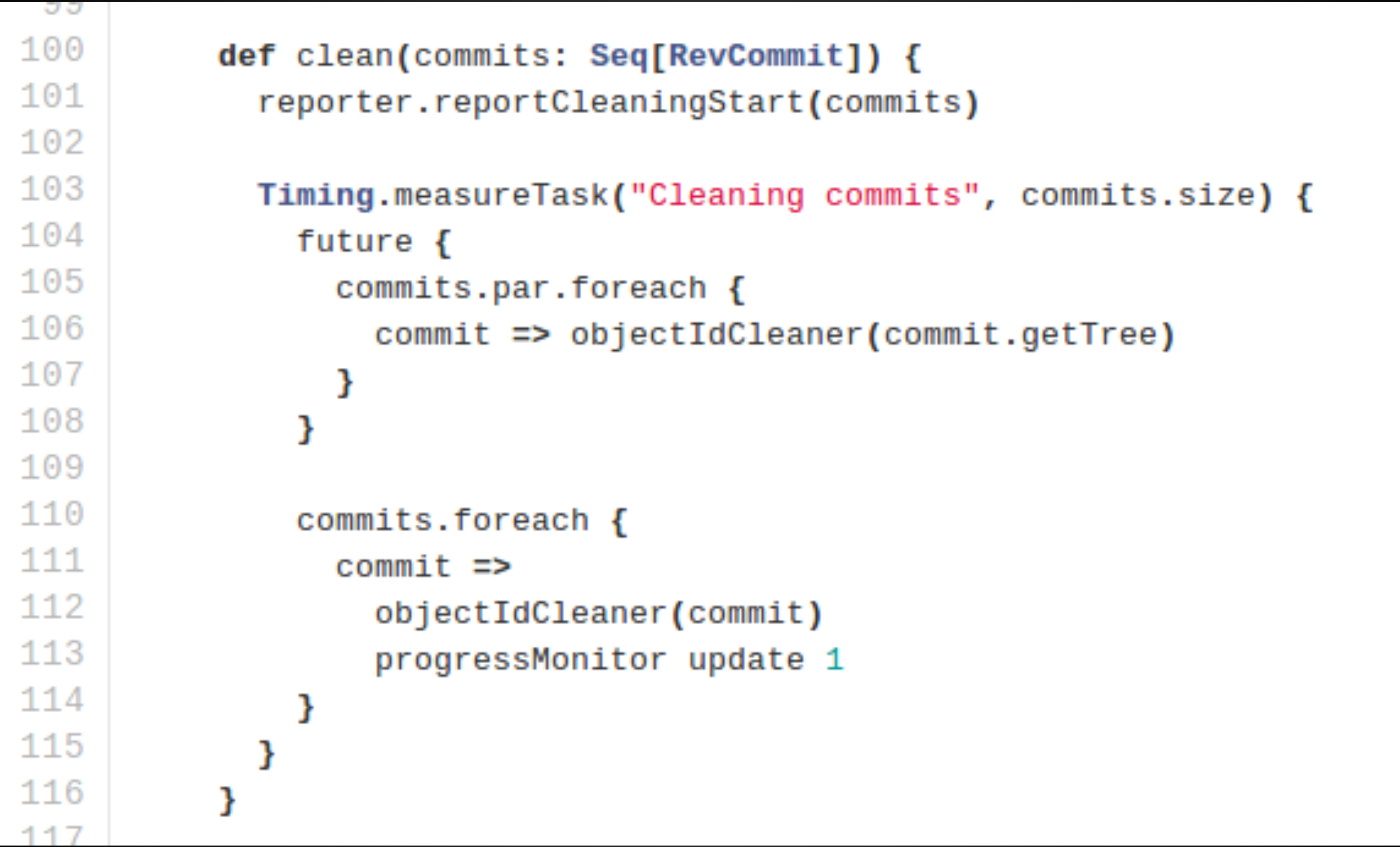
- Elliot Glaysher, Google Engineer

...faster...



Repository	Commit Count	Original Packsize	BFG duration (s)	git-filter-branch duration (s)	Speed increase
https://github.com/defunkt/github-gem	436	347,839	1.1	10.6	9.7
https://github.com/bfg-repo-cleaner-demos/jgit-original	2,408	7,582,999	2.8	84.7	30.3
https://github.com/git/git	31,137	42,584,569	37.8	1,706.3	45.1
https://github.com/bfg-repo-cleaner-demos/rails-original			10.8	1,871.7	174.0
https://github.com/bfg-repo-cleaner-demos/wine-original	93,532	271,968,514	24.8	5,374.4	216.5
https://github.com/bfg-repo-cleaner-demos/intellij-community-original	100,760	1,196,055,815	43.7	31,476.3	720.0
https://github.com/bfg-repo-cleaner-demos/gcc-original	148,495	1,526,634,714	183.2	27,332.2	149.2

...does *amazing* parallelism...



```
    def clean(commits: Seq[RevCommit]) {
      reporter.reportCleaningStart(commits)

      Timing.measureTask("Cleaning commits", commits.size) {
        future {
          commits.par.foreach {
            commit => objectIdCleaner(commit.getTree)
          }
        }

        commits.foreach {
          commit =>
            objectIdCleaner(commit)
            progressMonitor update 1
        }
      }
    }
```

Treats you like a reformed alcoholic...

...won't modify the latest version of your files, unless you really *really* want it to.

Updates the commit ids in your commit messages...

Before :

```
This fixes the bug that was introduced with
the refactor in 35c48634.
```

After :

```
This fixes the bug that was introduced with
the refactor in 54ed32ab [formerly
35c48634].
```

```
c4370fac0077 (817/4
b065a471d4a5 (818/4
f10439575fdf (819/4
5df024973eed (820/4
d27c32812f17 (821/4
e53d46910b06 (822/4
d32a155b93cd (823/4
```

```
 * commit 0246712b (protected
 by 'HEAD')

Cleaning
--------
```

rtyley.github.io/bfg-repo-cleaner

@rtyley