DVCS - Under the Bonnet

What's going on in there?

Jim Hague

LAIC AG

ACCU Conference 2011

Aims of the Session

- Look at the workings of two popular DVCS.
- Provide insights into their possible strengths and weaknesses.
- Compare engineering choices made in two applications developed in parallel and addressing the same problem.

Structure of the Session

- A bit of history. . .
- The approach.
- Overview of use.
- Files and data structures.
- Communication.
- Exit, pursued by a bear. . .

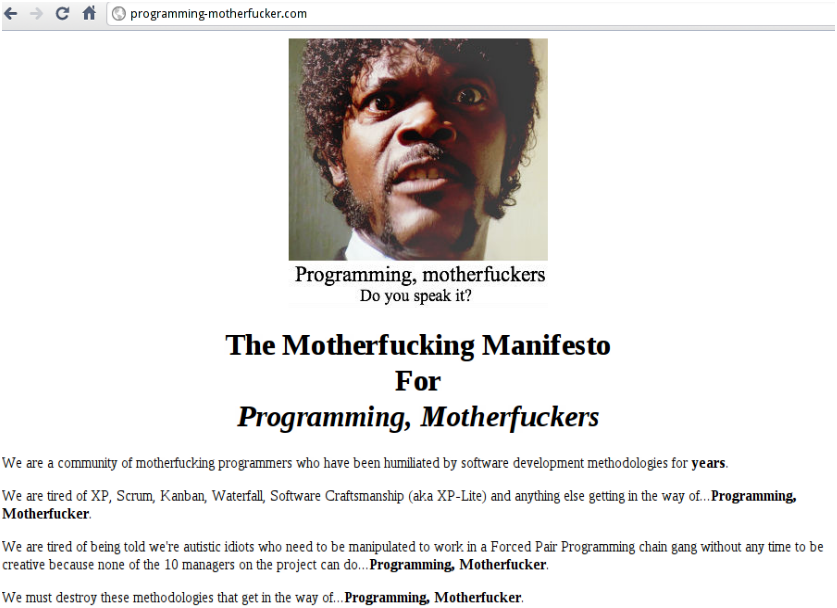
A bit of history. . .

A bit of history

Why do Mercurial and Git exist?

- The task of managing the Linux kernel source.
- And lo! There was Bitkeeper.
- Larry McEvoy spits the dummy.
- Monotone. So near, yet so far.

The response. . .



programming-motherfucker.com

Programming, motherfuckers
Do you speak it?

The Motherfucking Manifesto For *Programming, Motherfuckers*

We are a community of motherfucking programmers who have been humiliated by software development methodologies for **years**.

We are tired of XP, Scrum, Kanban, Waterfall, Software Craftsmanship (aka XP-Lite) and anything else getting in the way of...**Programming, Motherfucker**.

We are tired of being told we're autistic idiots who need to be manipulated to work in a Forced Pair Programming chain gang without any time to be creative because none of the 10 managers on the project can do...**Programming, Motherfucker**.

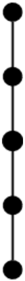
We must destroy these methodologies that get in the way of...**Programming, Motherfucker**.

Timeline

- 6/4/2005. BitMover Inc. announces withdrawal of open source licence.
- 7/4/2005. Linus mentions he is working on something. . .
- 7/4/2005. . . . and he's done a first commit (mostly by hand).
- 18/4/2005. First multi-branch merge in Git.
- 19/4/2005. Mercurial 0.1 announced.
  - Clone, pull, merge working.
  - Self-hosting.

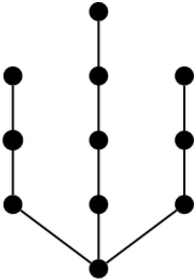
Traditional Central VCS

Central Version Control



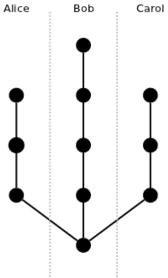
Traditional Central VCS with branch

Central Version Control
with branch



Distributed VCS

Distributed Version Control



A quick introduction to Git

An introduction to Git

- Written in: C, Shell, Perl.
- Runs on: Posix, Windows (Cygwin, msysGit), OS X, Solaris
- Who uses it? Linux kernel, Perl, Eclipse, KDE, Qt, Android, PostgreSQL. Oh, and GitHub.

An introduction to Git

Create a repository

```
$ git init
Initialized empty Git repository in /home/jim/doc/accu/ACCU2011/git
    -intro/project/.git/
```

An introduction to Git

Put something in it

```
$ echo 'There was a gibbon one morning' > pome.txt
$ git add pome.txt
$ git status
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   (use "git rm --cached <file>..." to unstage)
#
#       new file:   pome.txt
#
```

An introduction to Git

Commit

```
$ git commit -m "My Pome" --author="Jim Hague <jim.hague@acm.org>"
[master (root-commit) 2b20848] My Pome
 1 files changed, 1 insertions(+), 0 deletions(-)
 create mode 100644 pome.txt
```

Save author default

```
$ git config user.name "Jim Hague"
$ git config user.email "jim.hague@acm.org"
```

An introduction to Git

A further change

```
$ git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
    directory)
#
#       modified:   pome.txt
#
no changes added to commit (use "git add" and/or "git commit -a")
```

An introduction to Git

Commit it

```
$ git diff
diff --git a/pome.txt b/pome.txt
index b1e27b3..2ef4682 100644
--- a/pome.txt
+++ b/pome.txt
@@ -1 +1,2 @@
 There was a gibbon one morning
+said "I think I will fly to the moon".
$ git add pome.txt
$ git commit -m "A great second line"
[master 4e1042f] A great second line
 1 files changed, 1 insertions(+), 0 deletions(-)
```

An introduction to Git

A look at what we've done

```
$ git log
commit 4e1042f5f2b371a397ead4e4d49cb0c85f49f61f
Author: Jim Hague <jim.hague@acm.org>
Date:   Wed Mar 30 14:28:07 2011 +0100

    A great second line

commit 2b20848748b4e8ae788bc21597cf27bb752e3fb9
Author: Jim Hague <jim.hague@acm.org>
Date:   Wed Mar 30 14:28:04 2011 +0100

    My Pome
```

An introduction to Git

Make a clone

```
$ git clone . ../project2
Cloning into ../project2...
done.
2$ git log --oneline
4e1042f A great second line
2b20848 My Pome
```

A remote repository

```
2$ git branch
* master
2$ git remote
origin
```

An introduction to Git

Make a change in the clone

```
2$ git diff
diff --git a/pome.txt b/pome.txt
index 2ef4682..4cb32da 100644
--- a/pome.txt
+++ b/pome.txt
@@ -1,2 +1,2 @@
-There was a gibbon one morning
-said "I think I will fly to the moon".
+There was a baboon, who one afternoon,
+said "I think I will fly to the sun."
2$ git add pome.txt
2$ git commit -m "A better beginning."
[master e080605] A better beginning.
 1 files changed, 2 insertions(+), 2 deletions(-)
```

An introduction to Git

Make another change in the clone

```
2$ git status
# On branch master
# Your branch is ahead of ’origin/master’ by 1 commit.
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
    directory)
#
#       modified:   pome.txt
#
no changes added to commit (use "git add" and/or "git commit -a")
2$ git add pome.txt
2$ git commit -m "Finished first verse."
[master b8bed1b] Finished first verse.
 1 files changed, 2 insertions(+), 0 deletions(-)
```

An introduction to Git

A change back in the original

```
$ git mv pome.txt poem.txt
$ git commit -m "Rename my file."
[master 5b8ac50] Rename my file.
 1 files changed, 0 insertions(+), 0 deletions(-)
 rename pome.txt => poem.txt (100%)
```

An introduction to Git

Apply updates from the original

```
2$ git fetch
From /home/jim/doc/accu/ACCU2011/git-intro/project/.
   4e1042f..5b8ac50  master     -> origin/master
2$ git merge origin
Merge made by recursive.
 pome.txt => poem.txt |    0
 1 files changed, 0 insertions(+), 0 deletions(-)
 rename pome.txt => poem.txt (100%)
```

An introduction to Git

What do we have now?

```
2$ git log -3
commit 08a7086984139320447789 32a47a35004ded96f4
Merge: b8bed1b 5b8ac50
Author: Jim Hague <jim.hague@acm.org>
Date:   Wed Mar 30 14:28:14 2011 +0100

    Merge remote branch 'origin'

commit 5b8ac501b9bd20fb8709516b874bfb72632037f9
Author: Jim Hague <jim.hague@acm.org>
Date:   Wed Mar 30 14:28:13 2011 +0100

    Rename my file.

commit b8bed1b5097c5d5c52ba026b0748d9b96031f027
Author: Jim Hague <jim.hague@acm.org>
Date:   Wed Mar 30 14:28:12 2011 +0100

    Finished first verse.
```

An introduction to Git

Retrieve the changes

```
$ git pull ../project2
From ../project2
 * branch            HEAD       -> FETCH_HEAD
Updating 5b8ac50..08a7086
Fast-forward
 poem.txt |    6 ++++--
 1 files changed, 4 insertions(+), 2 deletions(-)
```

An introduction to Mercurial

A quick introduction to Mercurial

An introduction to Mercurial

- Written in: Python, C.
- Runs on: Posix, Windows, OS X, Solaris, AIX.
- Who uses it? Mozilla, Symbian, OpenOffice, OpenJDK, Google Go. Bitbucket.

An introduction to Mercurial

Create a repository

```
$ hg init
```

Put something in it

```
$ echo ’There was a gibbon one morning’ > pome.txt
$ hg add pome.txt
$ hg status
A pome.txt
```

An introduction to Mercurial

Commit

```
$ hg commit -m "My Pome" -u "Jim Hague <jim.hague@acm.org>"
```

Save author default

```
$ cat .hg/hgrc
[ui]
username = Jim Hague <jim.hague@acm.org>
```

An introduction to Mercurial

A further change

```
$ hg status
M pome.txt
$ hg diff --git
diff --git a/pome.txt b/pome.txt
--- a/pome.txt
+++ b/pome.txt
@@ -1,1 +1,2 @@
 There was a gibbon one morning
+said "I think I will fly to the moon".
$ hg commit -m "A great second line"
```

An introduction to Mercurial

A look at what we've done

```
$ hg log
changeset:   1:afa2c313db3b
tag:         tip
user:        Jim Hague <jim.hague@acm.org>
date:        Mon Apr 11 21:36:26 2011 +0100
summary:     A great second line

changeset:   0:e851bc079e45
user:        Jim Hague <jim.hague@acm.org>
date:        Mon Apr 11 21:36:22 2011 +0100
summary:     My Pome
```

An introduction to Mercurial

Make a clone

```
$ hg clone . ../project2
updating to branch default
1 files updated, 0 files merged, 0 files removed, 0 files
    unresolved
2$ hg log
changeset:   1:afa2c313db3b
tag:         tip
user:        Jim Hague <jim.hague@acm.org>
date:        Mon Apr 11 21:36:26 2011 +0100
summary:     A great second line

changeset:   0:e851bc079e45
user:        Jim Hague <jim.hague@acm.org>
date:        Mon Apr 11 21:36:22 2011 +0100
summary:     My Pome
```

An introduction to Mercurial

A remote repository

```
2$ cat .hg/hgrc
[paths]
default = /home/jim/doc/accu/ACCU2011/hg-intro/project
```

Make a change in the clone

```
2$ hg diff --git
diff --git a/pome.txt b/pome.txt
--- a/pome.txt
+++ b/pome.txt
@@ -1,2 +1,2 @@
-There was a gibbon one morning
-said "I think I will fly to the moon".
+There was a baboon, who one afternoon,
+said "I think I will fly to the sun."
2$ hg commit -m "A better beginning."
```

An introduction to Mercurial

Make another change in the clone

```
2$ hg status
M pome.txt
2$ hg commit -m "Finished first verse."
```

And a change back in the original

```
$ hg mv pome.txt poem.txt
$ hg commit -m "Rename my file."
```

An introduction to Mercurial

Apply updates from original

```
2$ hg pull
pulling from /home/jim/doc/accu/ACCU2011/hg-intro/project
searching for changes
adding changesets
adding manifests
adding file changes
added 1 changesets with 1 changes to 1 files (+1 heads)
(run 'hg heads' to see heads, 'hg merge' to merge)
2$ hg merge
merging pome.txt and poem.txt to poem.txt
0 files updated, 1 files merged, 0 files removed, 0 files
    unresolved
(branch merge, don't forget to commit)
```

An introduction to Mercurial

Commit the merge

```
2$ hg status
M poem.txt
R pome.txt
2$ hg commit -m "Merge branch"
```

Push back to original

```
2$ hg push
pushing to /home/jim/doc/accu/ACCU2011/hg-intro/project
searching for changes
adding changesets
adding manifests
adding file changes
added 3 changesets with 3 changes to 2 files
```

An introduction to Mercurial

Update original

```
$ hg update
1 files updated, 0 files merged, 0 files removed, 0 files
    unresolved
$ ls
poem.txt
$ cat poem.txt
There was a baboon, who one afternoon,
said "I think I will fly to the sun."
So with two great palms strapped to his arms,
He started his takeoff run.
```

Inside Git

Heading inside Git

Inside Git

Create a repository

```
$ git init
Initialized empty Git repository in /home/jim/doc/accu/ACCU2011/git
    -inside/project/.git/
```

Inside Git

Empty .git

```
$ ls -CF .git .git/objects .git/refs
.git:
branches/  config  description  HEAD  hooks/  info/  objects/  refs
    /

.git/objects:
info/  pack/

.git/refs:
heads/  tags/
$ cat .git/HEAD
ref: refs/heads/master
```

Inside Git

Setting the name

```
$ git config user.name "Jim Hague"
$ git config user.email "jim.hague@acm.org"
$ cat .git/config
[core]
        repositoryformatversion = 0
        filemode = true
        bare = false
        logallrefupdates = true
[user]
        name = Jim Hague
        email = jim.hague@acm.org
```

Inside Git

First commit

```
$ git add pome.txt
$ git commit -m "My Pome"
[master (root-commit) c1a813d] My Pome
 1 files changed, 1 insertions(+), 0 deletions(-)
 create mode 100644 pome.txt
```

Inside Git

.git after first commit

```
$ git log
commit c1a813d6a016c57dbf6e3ae4d7a9aa85cf238303
Author: Jim Hague <jim.hague@acm.org>
Date:   Tue Mar 29 16:59:20 2011 +0100

    My Pome
$ ls -CF .git
branches/       config       HEAD    index  logs/     refs/
COMMIT_EDITMSG  description  hooks/  info/  objects/
```

Inside Git

.git/objects after first commit

```
$ ls -CFR .git/objects
.git/objects:
7f/   b1/   c1/   info/   pack/

.git/objects/7f:
65b1313ea8d348d16e9a7358f31b874c9e9924

.git/objects/b1:
e27b3e56a80f28149507601e37f0cd3894667d

.git/objects/c1:
a813d6a016c57dbf6e3ae4d7a9aa85cf238303

.git/objects/info:

.git/objects/pack:
```

Inside Git

.git/objects after second commit

```
$ git commit -m "A great second line"
[master 4955bd9] A great second line
 1 files changed, 1 insertions(+), 0 deletions(-)
$ find .git/objects -type f
.git/objects/49/55bd9750586819f78058cd53f7edbd231c571e
.git/objects/74/c419bd9665f44781e67f5a1ea01efcbc4ebd79
.git/objects/7f/65b1313ea8d348d16e9a7358f31b874c9e9924
.git/objects/c1/a813d6a016c57dbf6e3ae4d7a9aa85cf238303
.git/objects/b1/e27b3e56a80f28149507601e37f0cd3894667d
.git/objects/2e/f468255526e6dd9cddd4cfe25521d1b0ea4c10
```

Inside Git

- Git object types
  - Blob. The contents of a file you're tracking.
  - Tree. A directory, listing a collection of objects. Objects may be either blobs or subtrees.
  - Commit. Information about a commit.
  - Tag. Details of an annotated tag.

Inside Git

A commit object

```
$ git log --oneline
4955bd9 A great second line
c1a813d My Pome
$ git cat-file -p 4955bd9
tree 74c419bd9665f44781e67f5a1ea01efcbc4ebd79
parent c1a813d6a016c57dbf6e3ae4d7a9aa85cf238303
author Jim Hague <jim.hague@acm.org> 1301414363 +0100
committer Jim Hague <jim.hague@acm.org> 1301414363 +0100

A great second line
```

Inside Git

The tree object

```
$ git cat-file -p 74c419bd9665f44781e67f5a1ea01efcbc4ebd79
100644 blob 2ef468255526e6dd9cddd4cfe25521d1b0ea4c10    pome.txt
$ git cat-file -p 2ef468255526e6dd9cddd4cfe25521d1b0ea4c10
There was a gibbon one morning
said "I think I will fly to the moon".
```

Inside Git

.git/refs

```
$ ls -CFR .git/refs
.git/refs:
heads/  tags/

.git/refs/heads:
master

.git/refs/tags:
$ cat .git/refs/heads/master
4955bd9750586819f78058cd53f7edbd231c571e
$ cat .git/HEAD
ref: refs/heads/master
```

Inside Git

- Pack files
  - Consolidate objects into a single data + index file pair.
  - Delta-compress similar objects.
  - Reduce storage requirement.
- Create by `git gc`.
- Run automatically:
  - On push to remote repository.
  - If thresholds exceeded (7000 loose objects, 50 packfiles).
- `git gc` also prunes unreachable nodes.

Inside Git

Packing objects

```
$ git gc
$ ls -CFR .git/objects
.git/objects:
info/  pack/

.git/objects/info:
packs

.git/objects/pack:
pack-d60c1fb218d2b98f2275f82a8b85e6f925b41785.idx
pack-d60c1fb218d2b98f2275f82a8b85e6f925b41785.pack
```

Inside Git

Packing objects

```
$ git verify-pack -v .git/objects/pack/pack-
    d60c1fb218d2b98f2275f82a8b85e6f925b41785.idx
2ef468255526e6dd9cddd4cfe25521d1b0ea4c10 blob   70 76 338
4955bd9750586819f78058cd53f7edbd231c571e commit 226 157 12
74c419bd9665f44781e67f5a1ea01efcbc4ebd79 tree   36 47 291
7f65b1313ea8d348d16e9a7358f31b874c9e9924 tree   36 47 414
b1e27b3e56a80f28149507601e37f0cd3894667d blob   31 41 461
c1a813d6a016c57dbf6e3ae4d7a9aa85cf238303 commit 166 122 169
non delta: 6 objects
.git/objects/pack/pack-d60c1fb218d2b98f2275f82a8b85e6f925b41785.
    pack: ok
```

Inside Git

Packing objects and refs

```
$ ls -CFR .git/refs
.git/refs:
heads/   tags/

.git/refs/heads:

.git/refs/tags:
$ cat .git/packed-refs
# pack-refs with: peeled
4955bd9750586819f78058cd53f7edbd231c571e refs/heads/master
```

Heading inside Mercurial

Inside Mercurial

Create a repository

```
$ hg init
$ ls -CFR .hg
.hg:
00changelog.i  requires  store/

.hg/store:
$ cat .hg/requires
revlogv1
store
fncache
dotencode
```

Inside Mercurial

First commit

```
$ echo ’There was a gibbon one morning’ > pome.txt
$ hg add pome.txt
$ hg commit -m "My Pome"
```

Inside Mercurial

.hg after first commit

```
$ ls -CFR .hg
.hg:
00changelog.i  last-message.txt  store/       undo.desc
dirstate       requires          undo.branch  undo.dirstate

.hg/store:
00changelog.i  00manifest.i  data/  fncache  undo

.hg/store/data:
pome.txt.i
```

Inside Mercurial

.hg after second commit

```
$ hg commit -m "A great second line"
$ ls -CFR .hg
.hg:
00changelog.i  dirstate          requires  undo.branch  undo.
    dirstate
cache/         last-message.txt  store/    undo.desc

.hg/cache:
branchheads

.hg/store:
00changelog.i  00manifest.i  data/  fncache  undo

.hg/store/data:
pome.txt.i
```

Inside Mercurial

- The Mercurial Revlog
  - A file or two (data + index) with versioned content.
  - Delta storage, with snapshots.
  - Append-only, for O(1) adding of new items.
  - Atomic transactions.

Inside Mercurial

File revlog

```
$ hg debugindex .hg/store/data/pome.txt.i
   rev    offset  length   base linkrev nodeid       p1
       p2
     0         0      32      0       0 6bbbd5d6cc53 000000000000
         000000000000
     1        32      51      0       1 83d266583303 6bbbd5d6cc53
         000000000000
$ hg debugdata .hg/store/data/pome.txt.i 0
There was a gibbon one morning
$ hg debugdata .hg/store/data/pome.txt.i 1
There was a gibbon one morning
said "I think I will fly to the moon".
```

Inside Mercurial

- The Manifest
  - List of files plus their revisions.
  - State of repository at a point in time.
- The Changelog
  - List of changes made in the repository.

Inside Mercurial

Manifest revlog

```
$ hg debugindex .hg/store/00manifest.i
   rev    offset  length   base linkrev nodeid       p1
       p2
     0         0      51      0       0 69daa5a9e08b 000000000000
         000000000000
     1        51      51      1       1 82a86d0ca4de 69daa5a9e08b
         000000000000
$ hg debugdata .hg/store/00manifest.i 0 | tr [\\0] _
pome.txt_6bbbd5d6cc53def4e9c8b03094de2e0f00b9d629
$ hg debugdata .hg/store/00manifest.i 1 | tr [\\0] _
pome.txt_83d266583303ee7adaf8d09cad0d020418189d6f
```

Inside Mercurial

Changelog revlog

```
$ hg debugindex .hg/store/00changelog.i
   rev    offset  length   base linkrev nodeid       p1
       p2
     0         0     106      0       0 6f455b280e17 000000000000
         000000000000
     1       106     103      0       1 d4b6d176ac48 6f455b280e17
         000000000000
$ hg debugdata .hg/store/00changelog.i 0
69daa5a9e08b3e1c51be5729d8e167cd4bd812a9
Jim Hague <jim.hague@acm.org>
1302087278 -3600
pome.txt

My Pome$ hg debugdata .hg/store/00changelog.i 1
82a86d0ca4de142b1a29140a13a1c3147970cd64
Jim Hague <jim.hague@acm.org>
1302087281 -3600
pome.txt

A great second line
```

Inside Mercurial

Copy/rename files

```
$ mkdir subdir
$ echo "#include <stdio.h>" > subdir/bar.c
$ hg add subdir/bar.c
$ hg commit -m "Add file in subdirectory"
$ hg debugdata .hg/store/00manifest.i 2 | tr [\\0] _
pome.txt_83d266583303ee7adaf8d09cad0d020418189d6f
subdir/bar.c_46dc8a668cb291a52a813a9ba3990a4acbf1b169
```

Inside Mercurial

Copy/rename files: file metadata

```
$ hg rename subdir/bar.c foo.c
$ hg commit -m "Move bar.c out of subdir, rename it foo.c"
$ hg debugdata .hg/store/00manifest.i 3 | tr [\\0] _
foo.c_a15d70387d7423c1e1cb22222194603 88df10efc
pome.txt_83d266583303ee7adaf8d09cad0d020418189d6f
$ hg debugdata .hg/store/data/foo.c.i 0 | tr [\\1] -
-
copy: subdir/bar.c
copyrev: 46dc8a668cb291a52a813a9ba3990a4acbf1b169
-
#include <stdio.h>
$ hg debugdata .hg/store/00changelog.i 3
de50c08af190f0d2e13b4d323aafc18c15139325
Jim Hague <jim.hague@acm.org>
1302087293 -3600
foo.c
subdir/bar.c

Move bar.c out of subdir, rename it foo.c
```

Remote Communication

Communicating with remote repositories

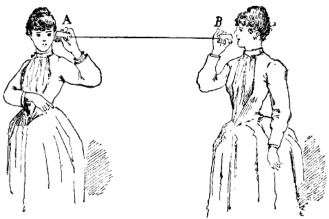


Fig. 76. Trådtelefon.

Getting remote changes in Git

1. `git fetch-pack` run locally, runs `git upload-pack` remotely.
2. Remote reports on its current branches and their heads.
3. Client requests wanted changes.
4. Server sends a pack.

Getting remote changes in Git

Client-server traffic

```
003fgit-upload-pack /home/jim/tmp/git-test2^@host=192.168.129.2^@
009b44c14a4a2314425ffba92943be144c96673b0727 HEAD^@multi_ack thin-
    pack side-band side-band-64k ofs-delta shallow no-progress
    include-tag multi_ack_detailed
003f44c14a4a2314425ffba92943be144c96673b0727 refs/heads/master
004608a70869841393204478932a47a35004ded96f4 refs/remotes/origin/
    HEAD
004808a70869841393204478932a47a35004ded96f4 refs/remotes/origin/
    master
00000073want 44c14a4a2314425ffba92943be144c96673b0727
    multi_ack_detailed side-band-64k thin-pack include-tag ofs-
    delta
00000032have 08a70869841393204478932a47a35004ded96f4
0009done
0038ACK 08a70869841393204478932a47a35004ded96f4 common
0031ACK 08a70869841393204478932a47a35004ded96f4
....
0000
```

Getting remote changes in Mercurial

1. Request remote heads.
2. For each head:
    1. Request last base (merge change) and parents.
    2. If base not known, add base parents to heads.
3. For each head, base:
    1. Request linear changes between head and base.
    2. Remote reports change 1,2,4,8,16. . . .
    3. Bisect to find first unknown change.
4. Request set of changes starting at all first unknown changes to remote heads.

Getting remote changes in Mercurial

HTTP Client-server traffic - getting heads

```
GET /?cmd=heads HTTP/1.1
Accept-Encoding: identity
host: localhost:8000
accept: application/mercurial-0.1
user-agent: mercurial/proto-1.0

HTTP/1.1 200 Script output follows
Server: BaseHTTP/0.3 Python/2.6.6
Date: Mon, 11 Apr 2011 21:00:19 GMT
Content-Type: application/mercurial-0.1
Content-Length: 41

cac9ac14985f51aaeb6d8587da416d7176f5c176
```

Getting remote changes in Mercurial

HTTP Client-server traffic - discover new branches

```
GET /?nodes=cac9ac14985f51aaeb6d8587da416d7176f5c176&cmd=branches
    HTTP/1.1

cac9ac14985f51aaeb6d8587da416d7176f5c176 39
    f4a9f96f98155f97caa6244d4bb3266f061292
    ef50509cd0ac6e78ad47e16c9d2c1bcadf45a8a5 30
    e996375e82af0321a15b81df0621cd29f81598

GET /?nodes=ef50509cd0ac6e78ad47e16c9d2c1bcadf45a8a5+30
    e996375e82af0321a15b81df0621cd29f81598&cmd=branches HTTP/1.1

ef50509cd0ac6e78ad47e16c9d2c1bcadf45a8a5
    b18ce11b76c8a0bc6edc90f3051328c3e5cc56f5
    b1fad46d57810cacc73c82fdf433dd7aa95356d5 1
    dd1fea47d594a2df7b90fac8fae4bcd60563ae8
30e996375e82af0321a15b81df0621cd29f81598 0818
    aa72d972fa3adef5d0e34529b58afc205a79 5
    e99d79e47f015076d6834b1d8860eb88af8301a 0122
    a8278387299f07961e208aa22a709a1093df
```

Getting remote changes in Mercurial

HTTP Client-server traffic - find range

```
GET /?pairs=ef50509cd0ac6e78ad47e16c9d2c1bcadf45a8a5-
    b18ce11b76c8a0bc6edc90f3051328c3e5cc56f5+5
    e99d79e47f015076d6834b1d8860eb88af8301a-
    b18ce11b76c8a0bc6edc90f3051328c3e5cc56f5+0122
    a8278387299f07961e208aa22a709a1093df-5251
    c4e4fb00d4833c670c0f85f00db14ac99023&cmd=between HTTP/1.1

848a2214bf0d01a9317bb171321251afe57dcc49
    fcfb308ae40fee1edd5b2a3049ee9a7034300e4e 73024
    e1c5d3d67a688d13214ae66c388716fa2dc
    b0396488d1c2d0720a3cc272269445b906bf9512 067
    d0dbcfdad89595bb1adbad7079c2b8c4c271f
...

GET /?pairs=fcfb308ae40fee1edd5b2a3049ee9a7034300e4e-73024
    e1c5d3d67a688d13214ae66c388716fa2dc+
    fcfb308ae40fee1edd5b2a3049ee9a7034300e4e-73024
    e1c5d3d67a688d13214ae66c388716fa2dc&cmd=between HTTP/1.1

32838dca4961eeeeea7dd7fca191e8a626c934e1
32838dca4961eeeeea7dd7fca191e8a626c934e1
```

Getting remote changes in Mercurial

HTTP Client-server traffic - get data

```
GET /?cmd=capabilities HTTP/1.1

lookup changegroupsubset branchmap pushkey stream unbundle=HG10GZ,
    HG10BZ,HG10UN

GET /?bases=32838dca4961eeeeea7dd7fca191e8a626c934e1+0122
    a8278387299f07961e208aa22a709a1093df&cmd=changegroupsubset&
    heads=cac9ac14985f51aaeb6d8587da416d7176f5c176 HTTP/1.1

Bundle data...
```

Nearly there, now. . .

- Neither are great for large binary files.
  - Mercurial possibly worse?
- Git better now if doing lots of renaming.
  - Mercurial catching up.
- Be careful when putting repos on network shares.
- User interface and environment issues dominate choice.

Exit. . .