Your Connection to ICT Research

Git: développez sans contraintes

Jeudis du Libre, UMons

FEDER

UNION EUROPEENNE

Wallonie

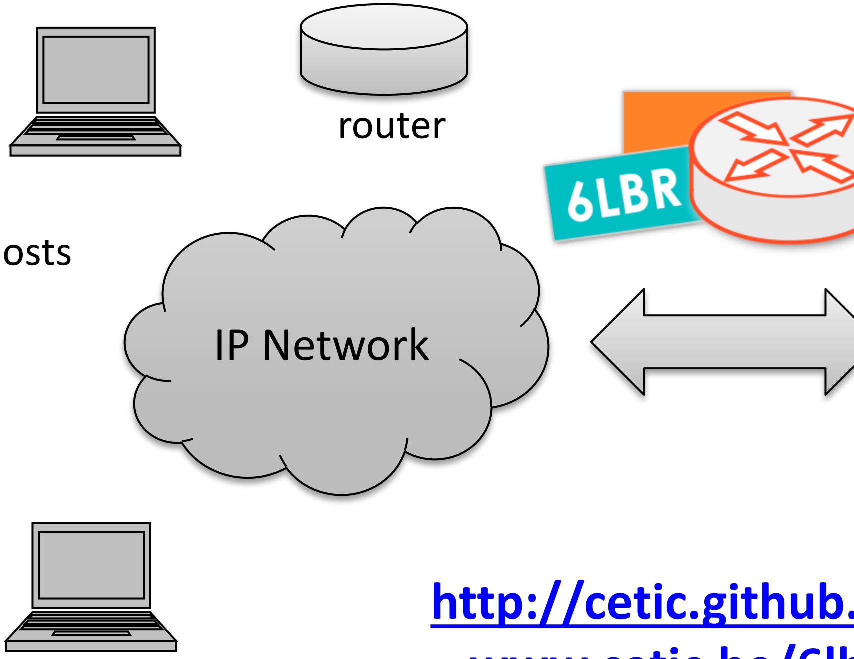
LE FONDS EUROPEEN DE DEVELOPPEMENT REGIONAL
ET LA WALLONIE INVESTISSENT DANS VOTRE AVENIR.

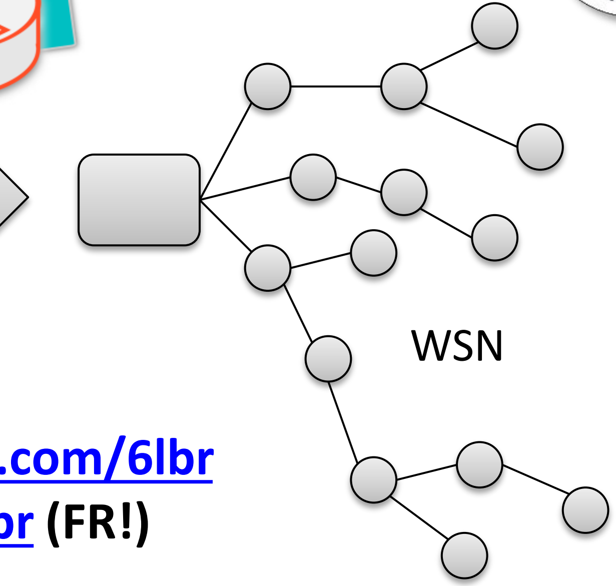
Sébastien DAWANS

18/04/2013

L’Internet des Objets

- 24 avril 2013 à Charleroi
- GDD CETIC, 06/2013
- Jeudi du libre 10/2013 (?)





http://cetic.github.com/6lbr

www.cetic.be/6lbr **(FR!)**

L'Internet des Objets

- 24 avril 2013 à Charleroi
- GDD CETIC, 06/2013
- Jeudi du libre 10/2013 (?)

Powered by Git !

http://cetic.github.com/6lbr

www.cetic.be/6lbr **(FR!)**

github
SOCIAL CODING

bitbucket
by ATLASSIAN

GITORIOUS

git

Git: Développez sans contraintes

~~Git: Développez sans contraintes~~

Git: Développez avec vos propres contraintes

~~Git: Développez sans contraintes~~

~~Git: Développez avec vos propres contraintes~~

Git: Décentralisez vos sources

~~Git: Développez sans contraintes~~

~~Git: Développez avec vos propres contraintes~~

~~Git: Décentralisez vos sources~~

Augmentez votre productivité avec Git

How GitHub will save the World Economy

How Git and **GitHub will** help developers **save** time and help businesses be efficient so they can turn **the World Economy** around

~~Git: Développez sans con~~ ~~intes~~

~~Git: Développez a~~ ~~contraintes~~

~~Git: Décentralis~~

~~Augmentez votre~~ ~~vité avec Git~~

It’s a bird… it’s a plane… no! It’s Git!

… and it will save humanity

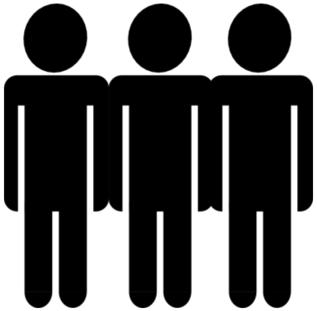
Source Code Management

« Software Configuration Management »

Source Code Management

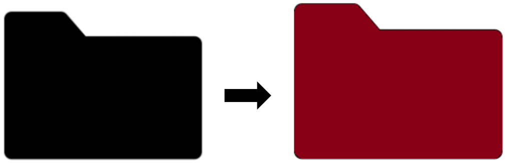
« Software Configuration Management »

vs

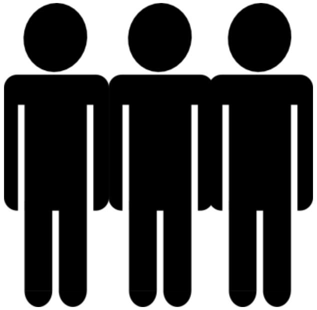


Source Code Management

« Software Configuration Management »



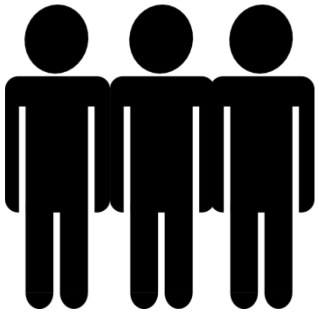
vs

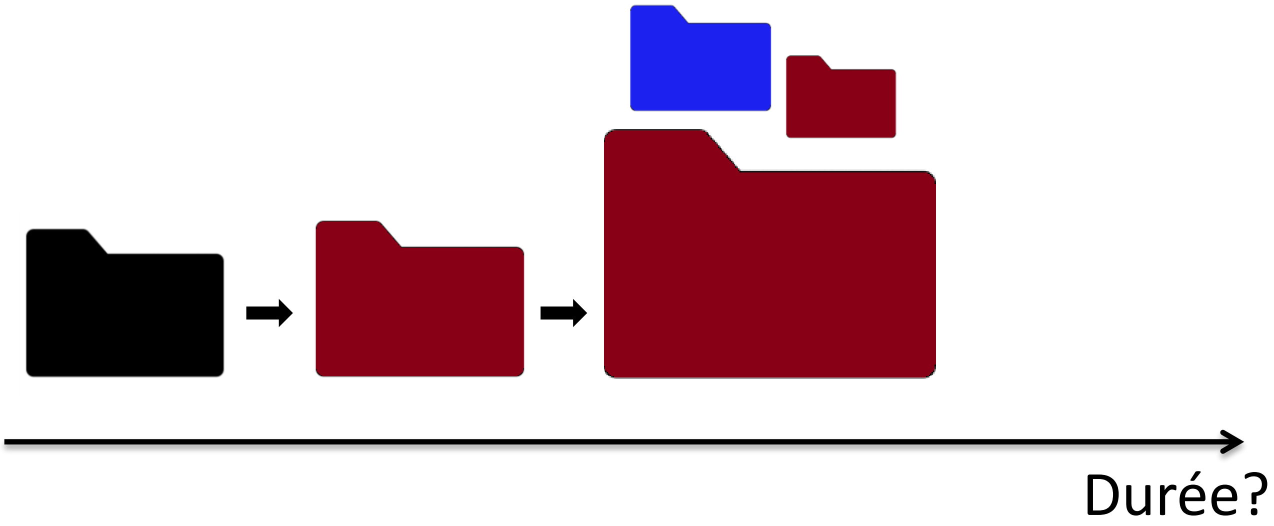


Source Code Management

« Software Configuration Management »

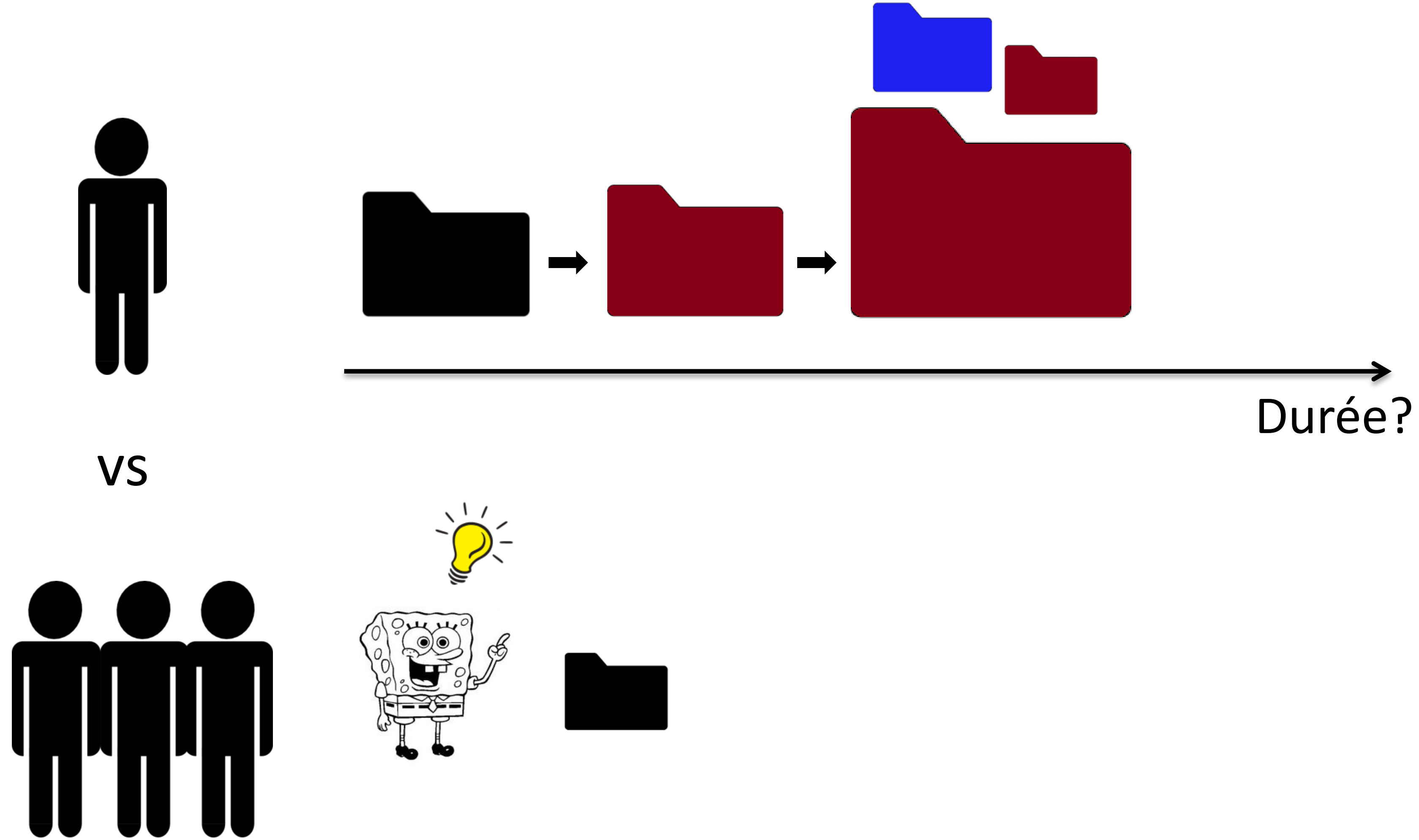
vs





Source Code Management

« Software Configuration Management »

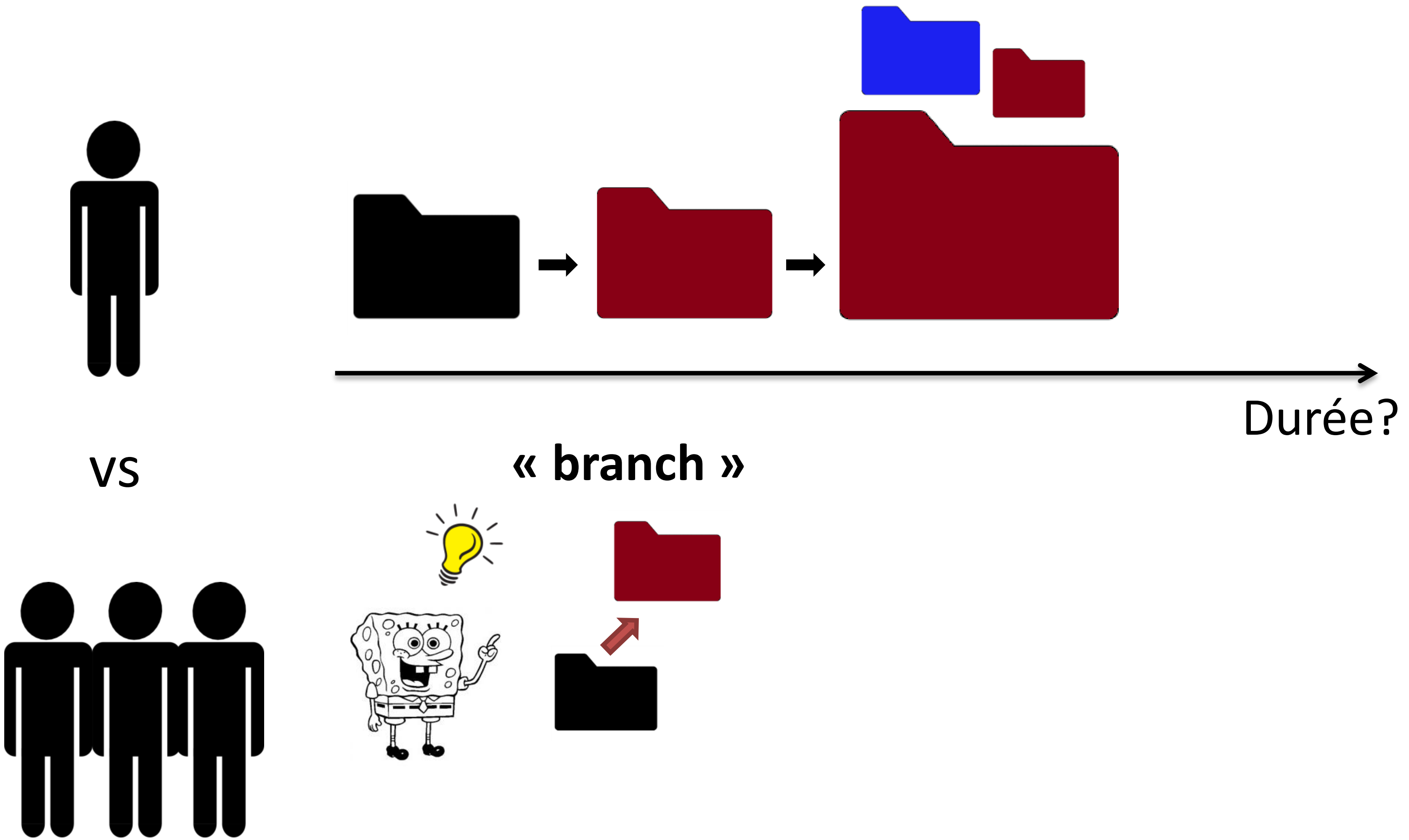


Durée?

vs

Source Code Management

« Software Configuration Management »



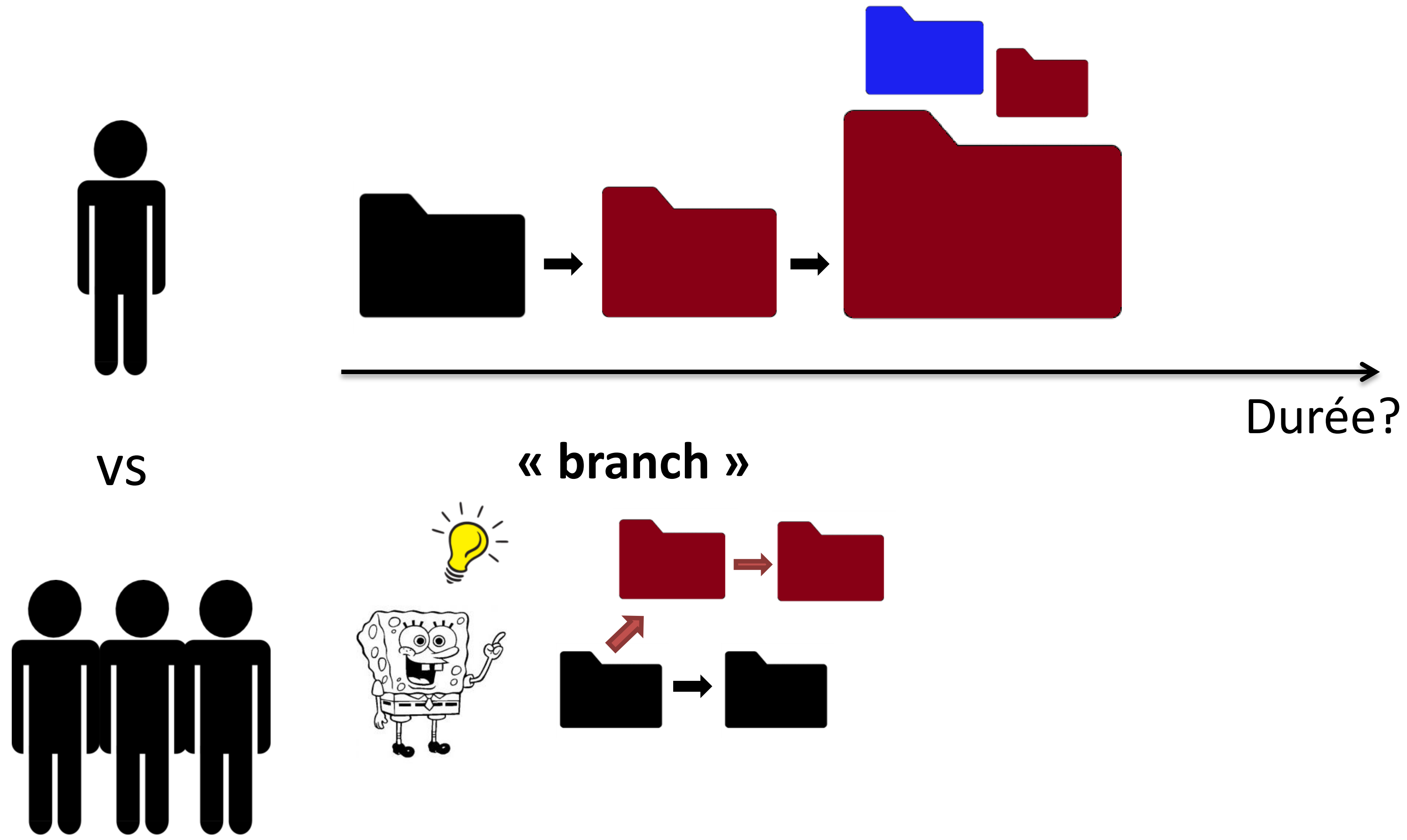
Durée?

vs

« branch »

Source Code Management

« Software Configuration Management »



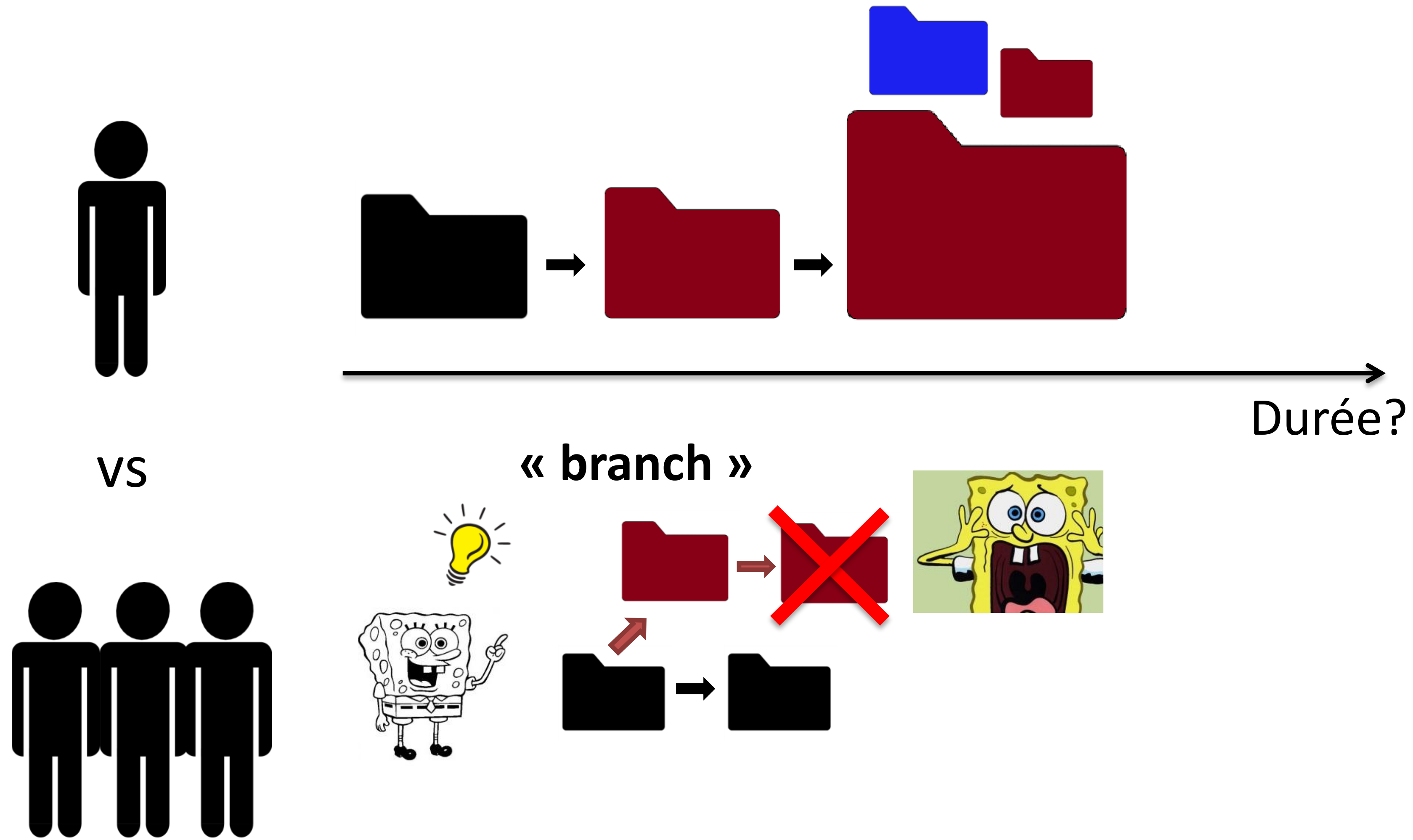
Durée?

vs

« branch »

Source Code Management

« Software Configuration Management »



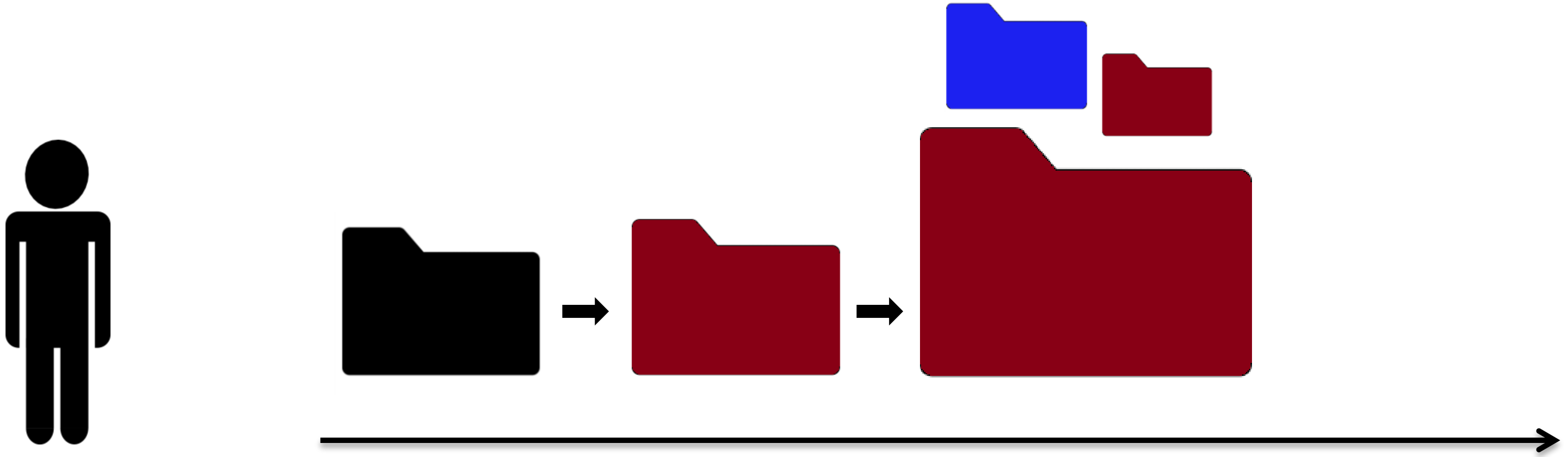
Durée?

vs

« branch »

Source Code Management

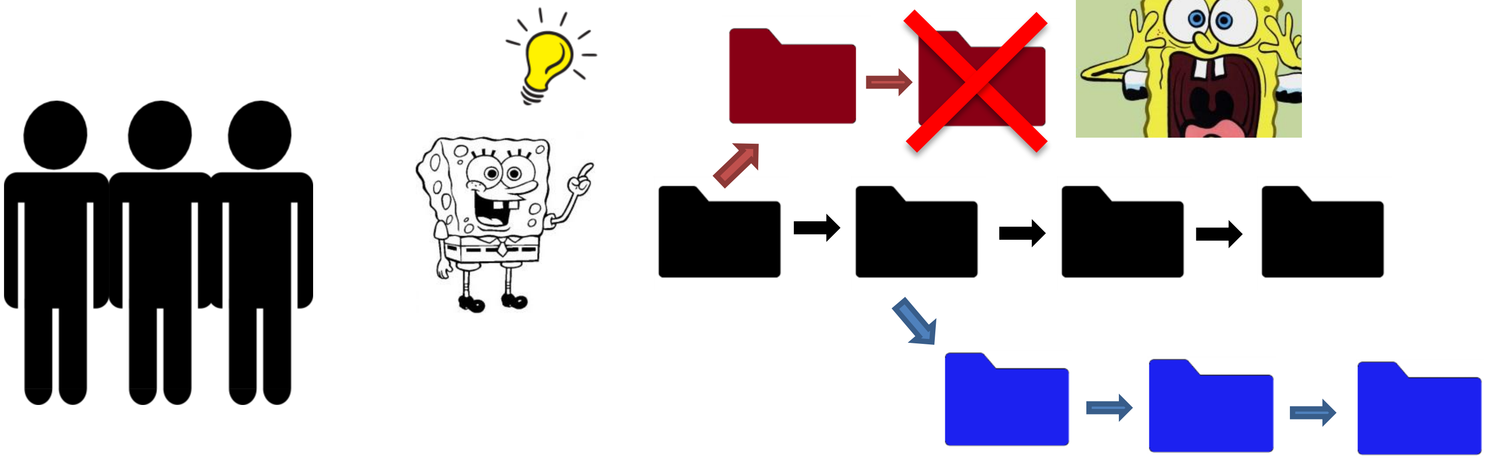
« Software Configuration Management »



Durée?

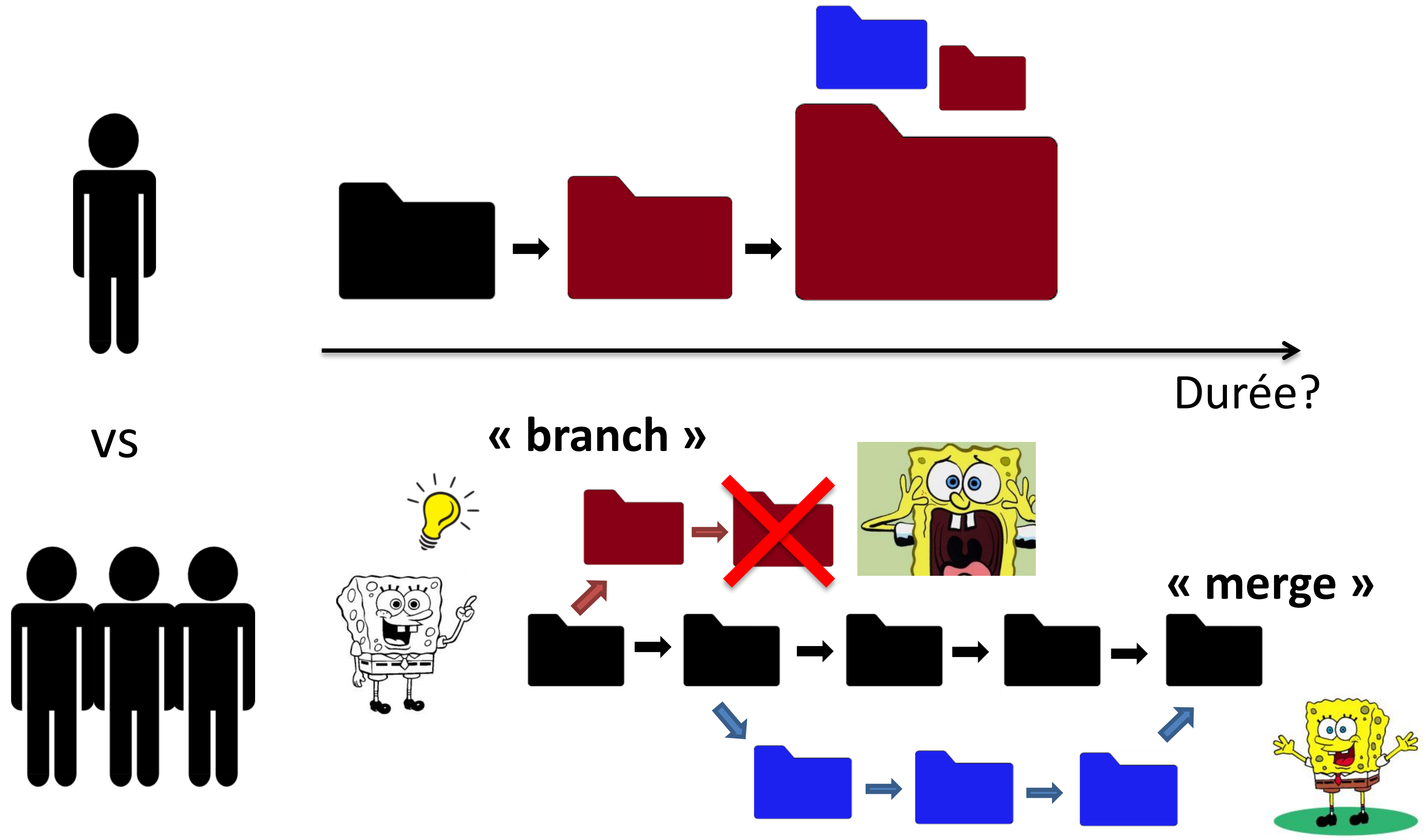
vs

« branch »



Source Code Management

« Software Configuration Management »



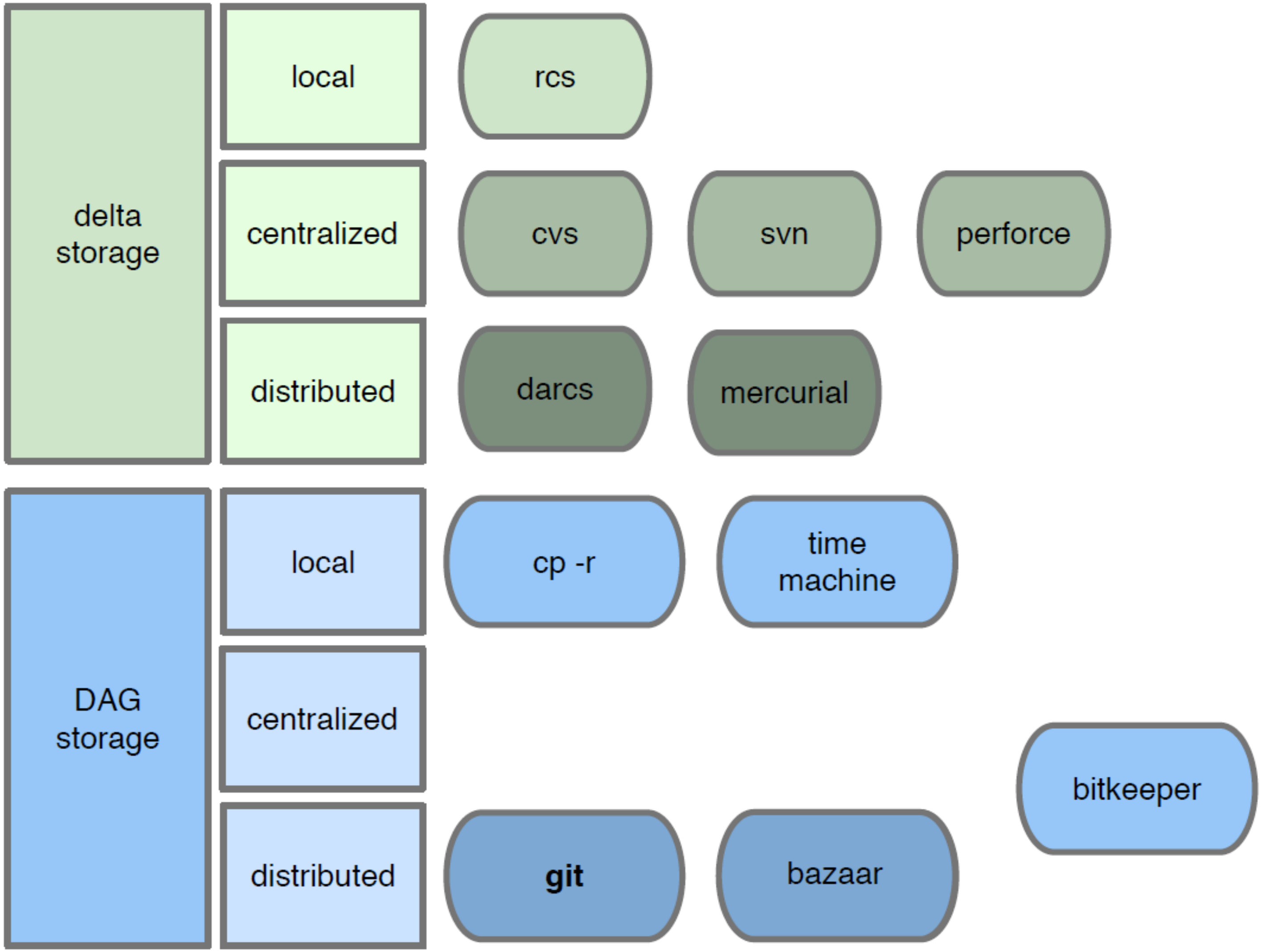
vs

Durée?

« branch »

« merge »

Panorama des SCM



delta storage

local

rcs

centralized

cvs

svn

perforce

distributed

darcs

mercurial

DAG storage

local

cp -r

time machine

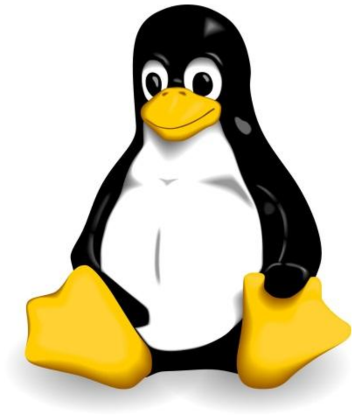
centralized

bitkeeper

distributed

git

bazaar



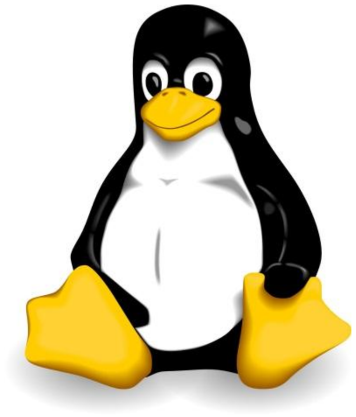
Un peu d’histoire…

1991

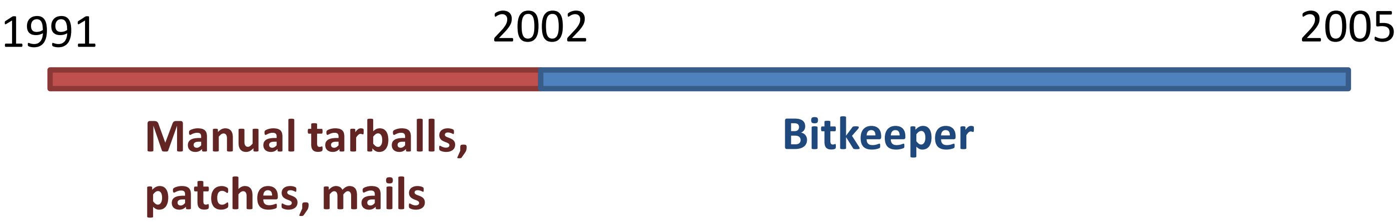
2002

Manual tarballs, patches, mails

« Much superior source control management system than CVS is »



Un peu d’histoire…



1991

2002

2005

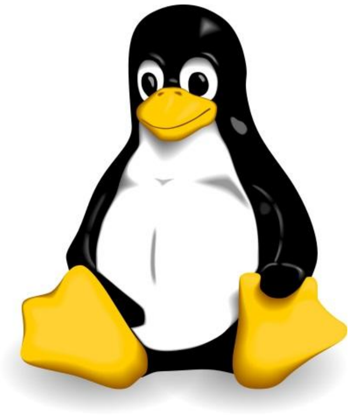
Manual tarballs, patches, mails

Bitkeeper

« Much superior source control management system than CVS is »

Controversé car commercial
Does the job right!
Conditions:

- No reverse-engineering
- No development of competing solution



Un peu d'histoire…

1991 — 2002 — 2005 — Aujourd'hui

Manual tarballs, patches, mails

Bitkeeper

Git

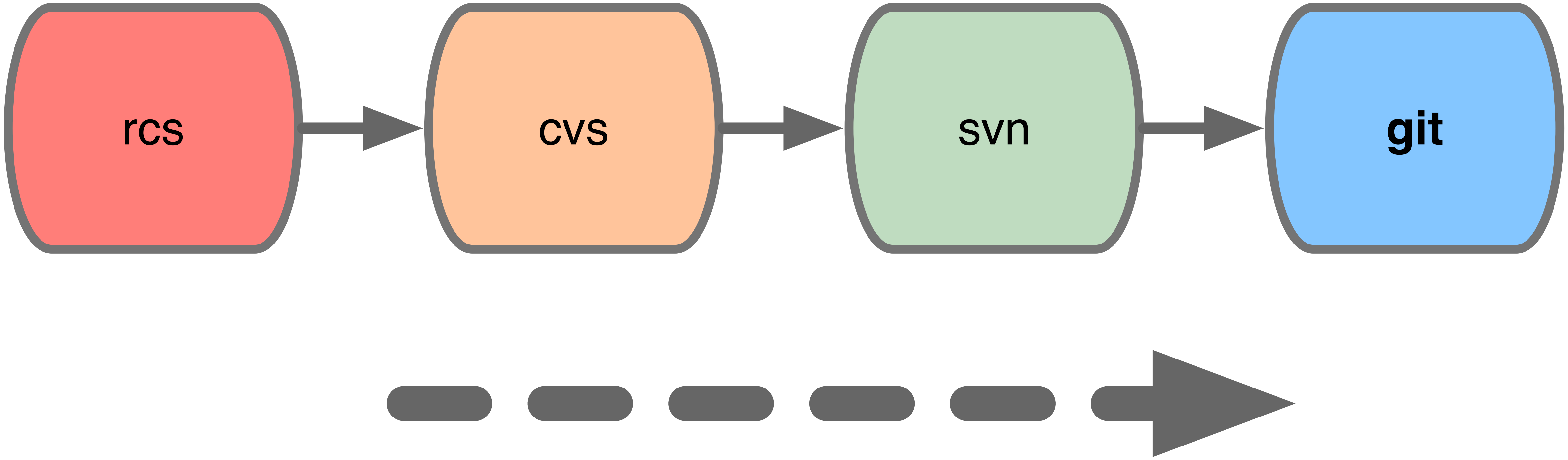
« Much superior source control management system than CVS is »

Controversé car commercial
Does the job right!
Conditions:

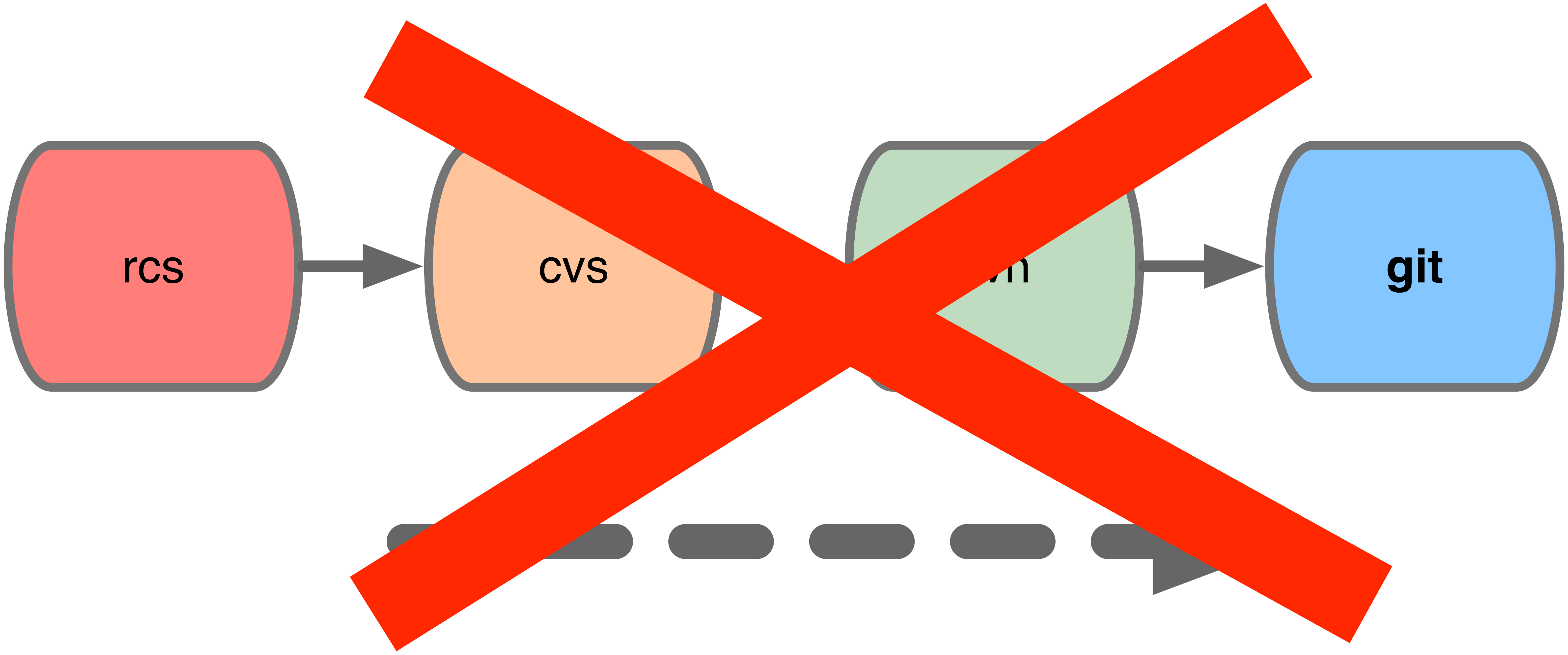
- No reverse-engineering
- No development of competing solution

- Linus Torvals
- Junio C Hamano

not an evolution



not an evolution



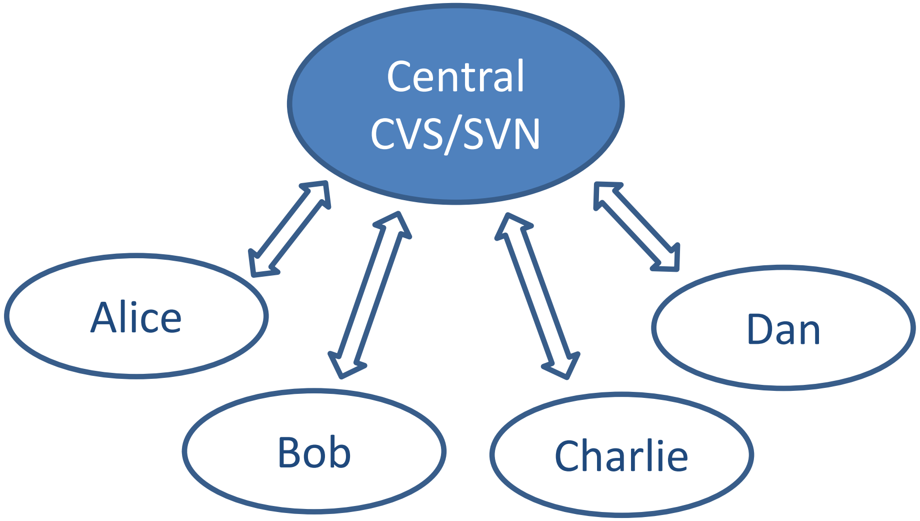
3 principes de design

Distribué

Rapide

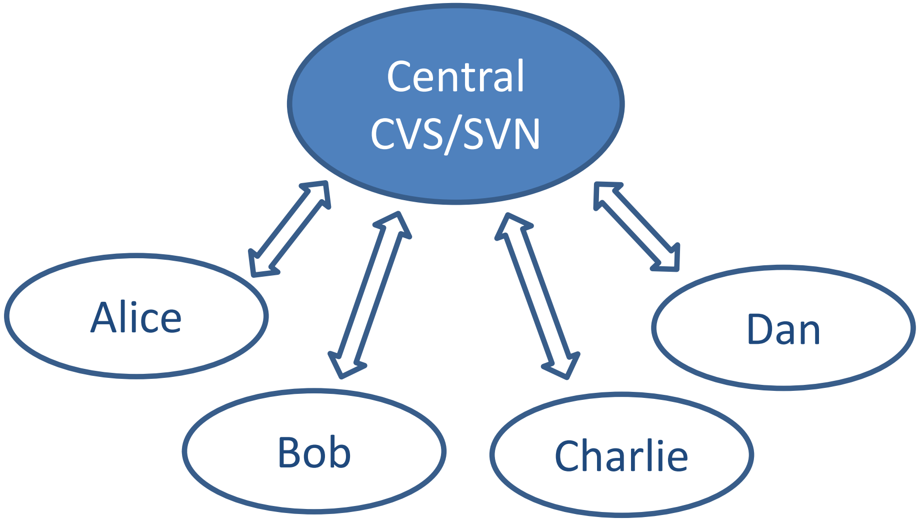
Fiable

Centralisé vs. Distribué



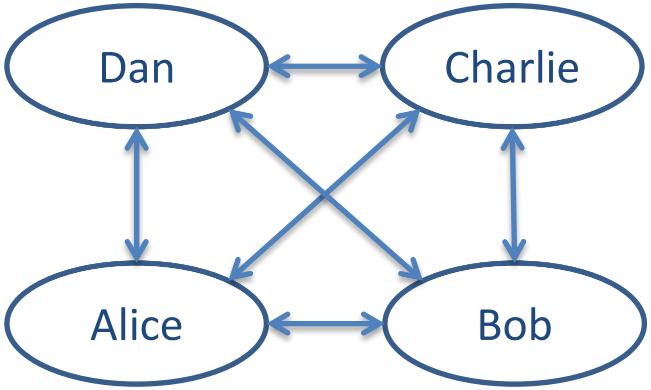
- Gestion de droits d’accès
- Commit policies
- Single Point of Failure (SPoF)

Centralisé vs. Distribué

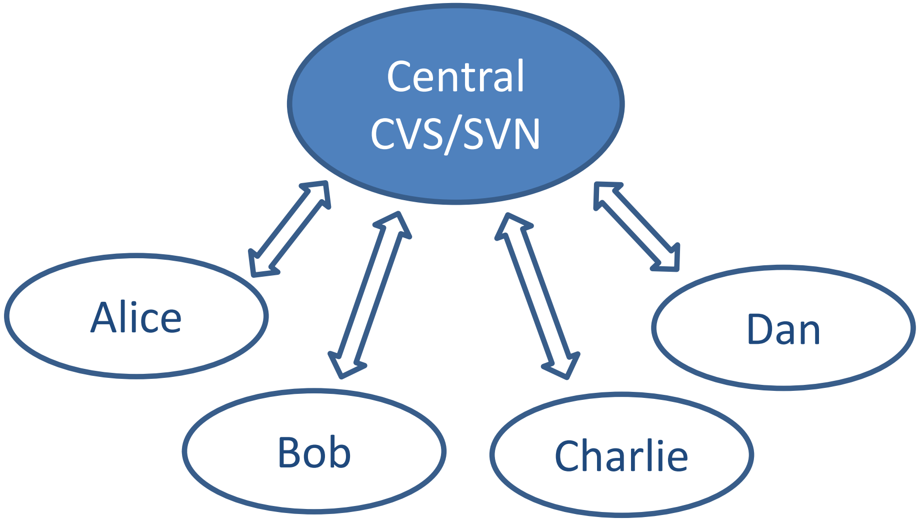


- Gestion de droits d'accès
- Commit policies
- Single Point of Failure (SPoF)

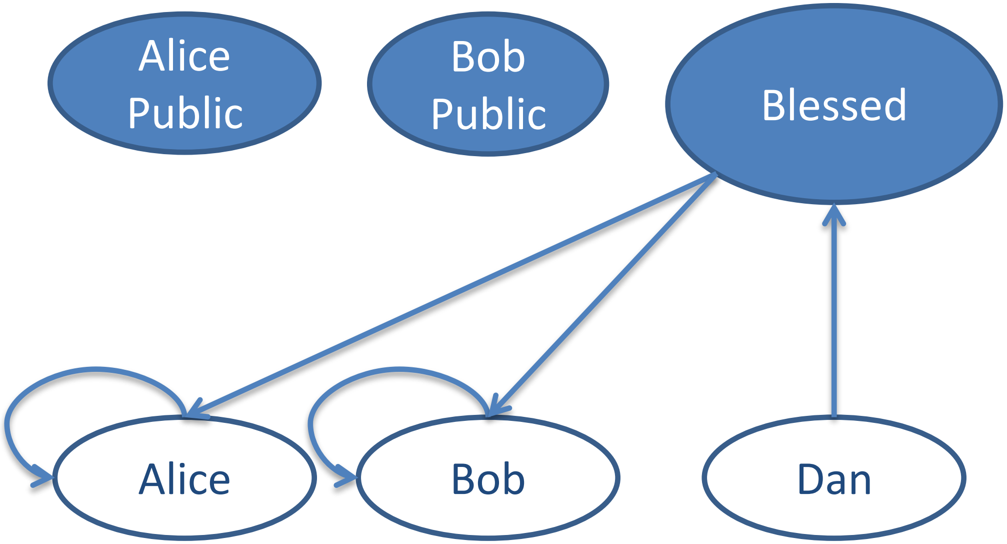
- Pas de workflow ou organisation imposé
- Contributeur bénéficie d'un SCM complet
- Intégration simplifiée



Centralisé vs. Distribué

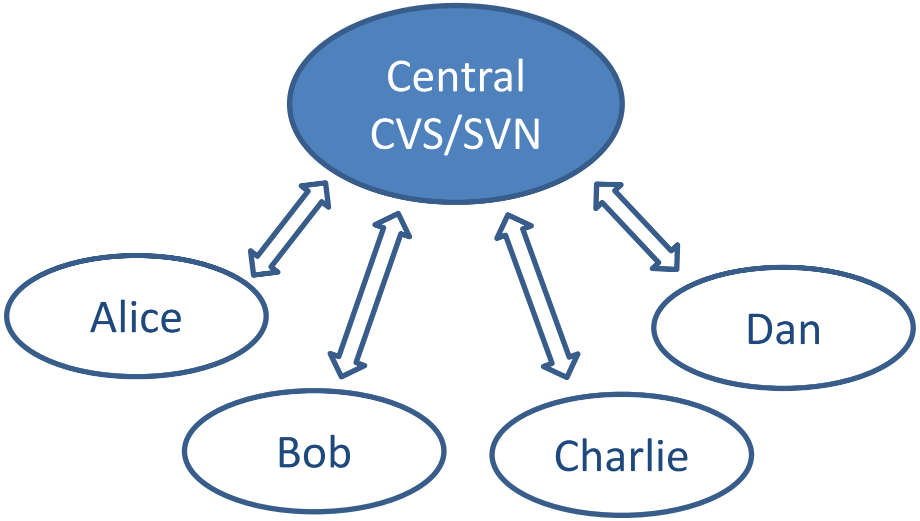


- Gestion de droits d’accès
- Commit policies
- Single Point of Failure (SPoF)

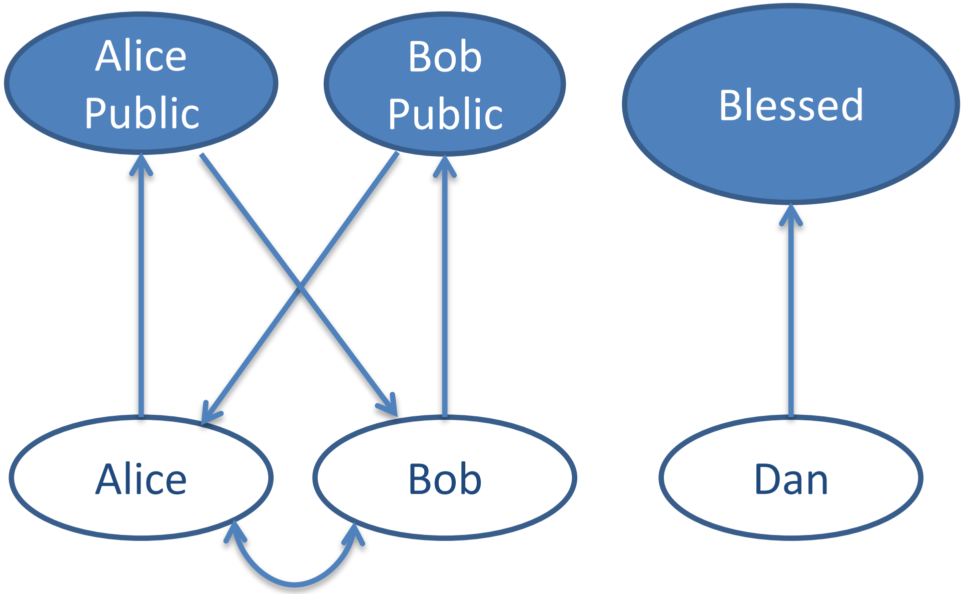


- Pas de workflow ou organisation imposé
- Contributeur bénéficie d’un SCM complet
- Intégration simplifiée

Centralisé vs. Distribué

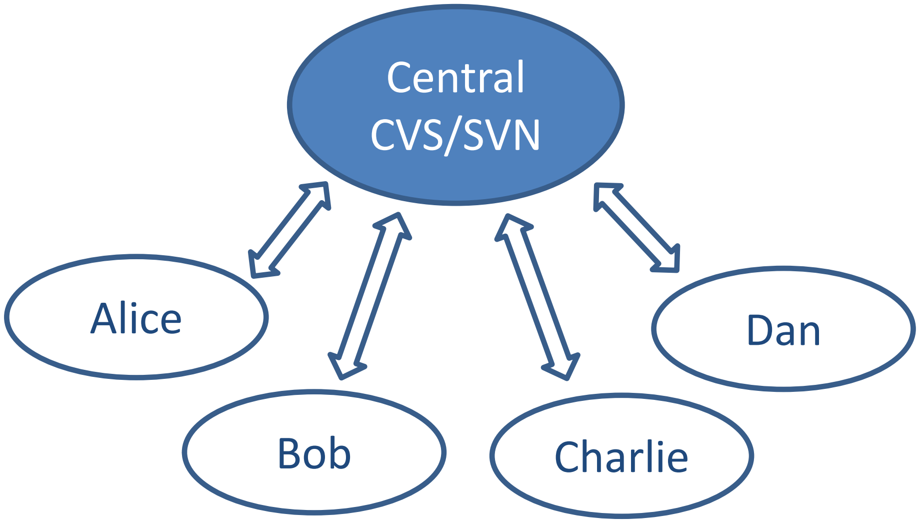


- Gestion de droits d’accès
- Commit policies
- Single Point of Failure (SPoF)

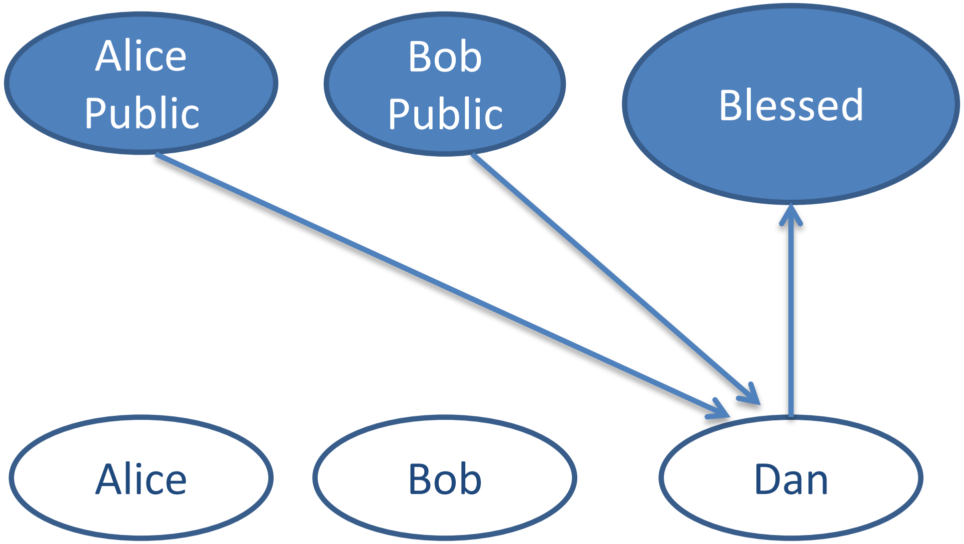


- Pas de workflow ou organisation imposé
- Contributeur bénéficie d’un SCM complet
- Intégration simplifiée

Centralisé vs. Distribué

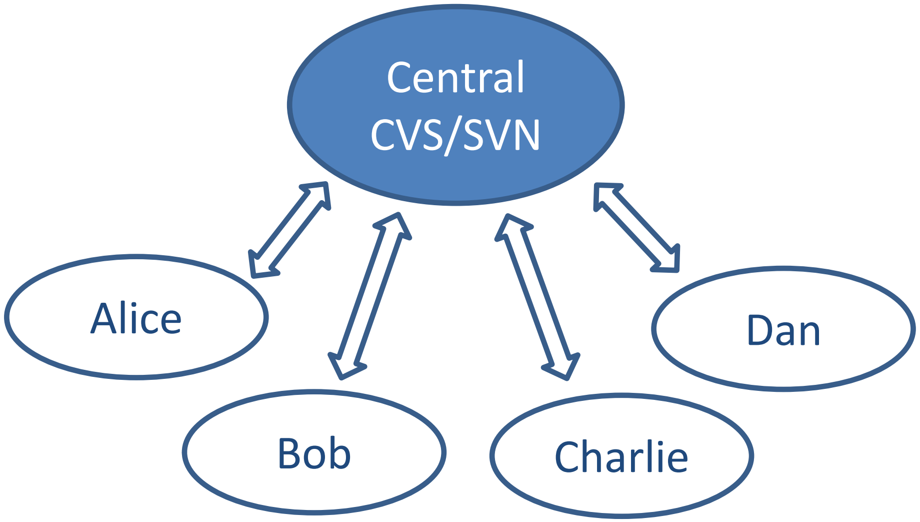


- Gestion de droits d’accès
- Commit policies
- Single Point of Failure (SPoF)



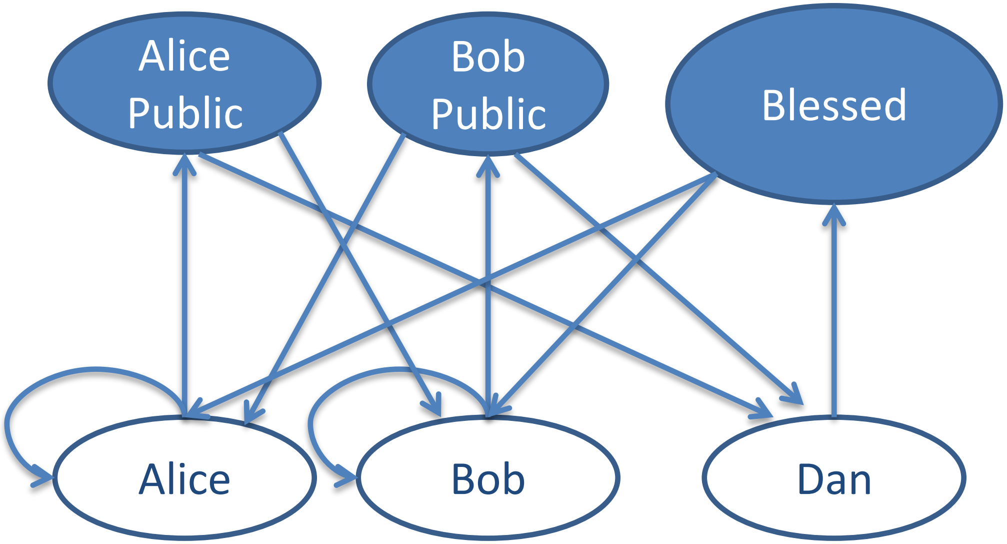
- Pas de workflow ou organisation imposé
- Contributeur bénéficie d’un SCM complet
- Intégration simplifiée

Centralisé vs. Distribué



- Gestion de droits d’accès
- Commit policies
- Single Point of Failure (SPoF)

- Pas de workflow ou organisation imposé
- Contributeur bénéficie d’un SCM complet
- Intégration simplifiée



Rapidité de Git

- Pas d'overhead réseau pour vos opérations quotidiennes:
  - Committer des changements
  - Diff
  - Visualiser l'historique
  - Brancher
  - Changer de branches
  - Merger des branches
  - …

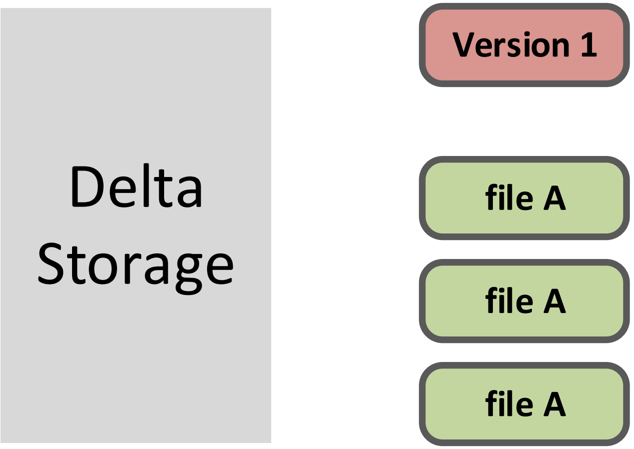
Rapidité de Git

- Content-Addressable File System
  - « Track contents, not files »
  - Index dans le data-store reflète le contenu

Fiabilité de Git

- Checksums (SHA-1) pour tous les objets manipulés par Git:
  - Détection de corruption
  - Assurance d’authenticité

- Distribué → Risque de SPoF minimisé
  - Même 1 dev seul, déjà 2 copies (local, remote)
  - Dossier .git/ contient tout



Delta Storage

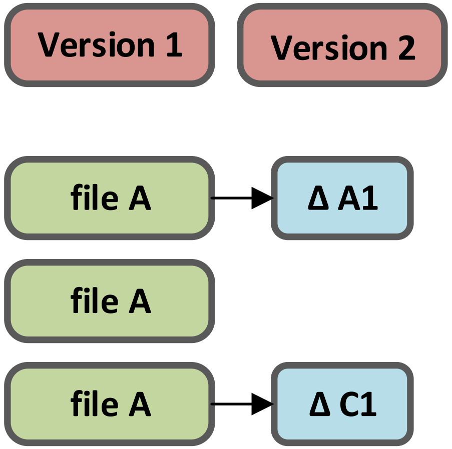
Version 1

file A

file A

file A

Delta Storage



Version 1

Version 2

file A

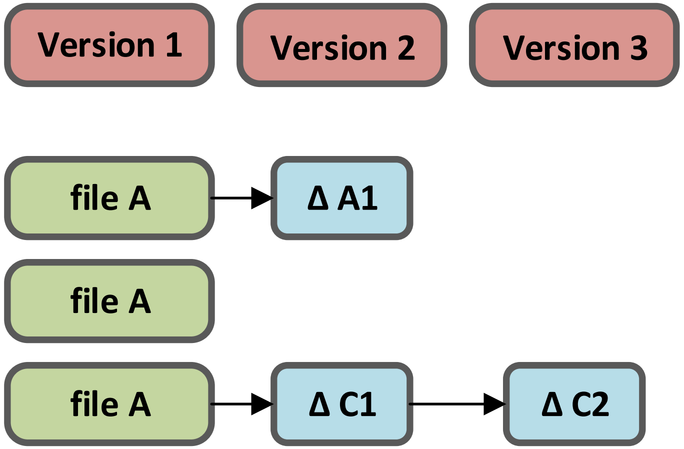
Δ A1

file A

file A

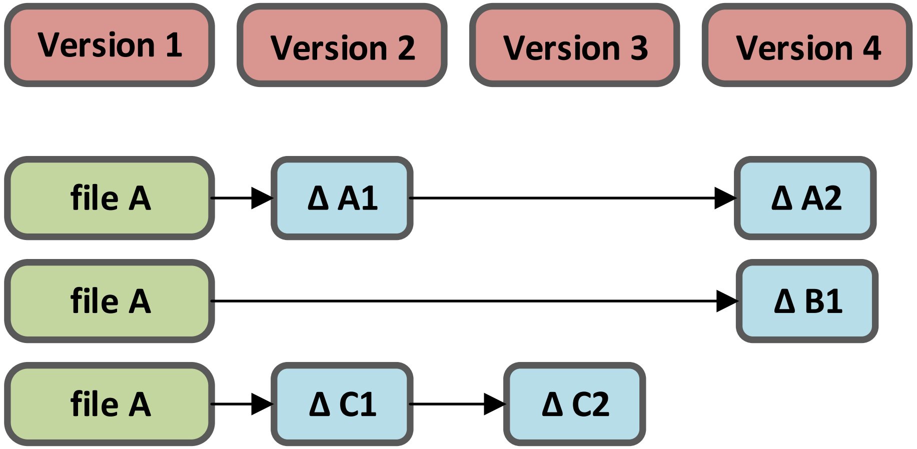
Δ C1

Delta Storage



Version 1	Version 2	Version 3
file A	Δ A1	
file A		
file A	Δ C1	Δ C2

Delta Storage



Version 1

Version 2

Version 3

Version 4

file A

Δ A1

Δ A2

file A

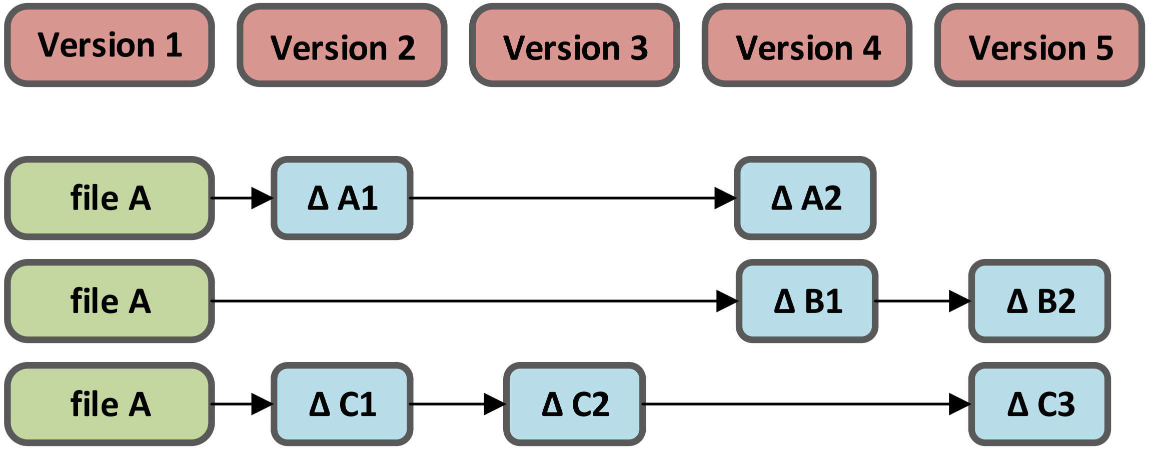
Δ B1

file A

Δ C1

Δ C2

Delta Storage



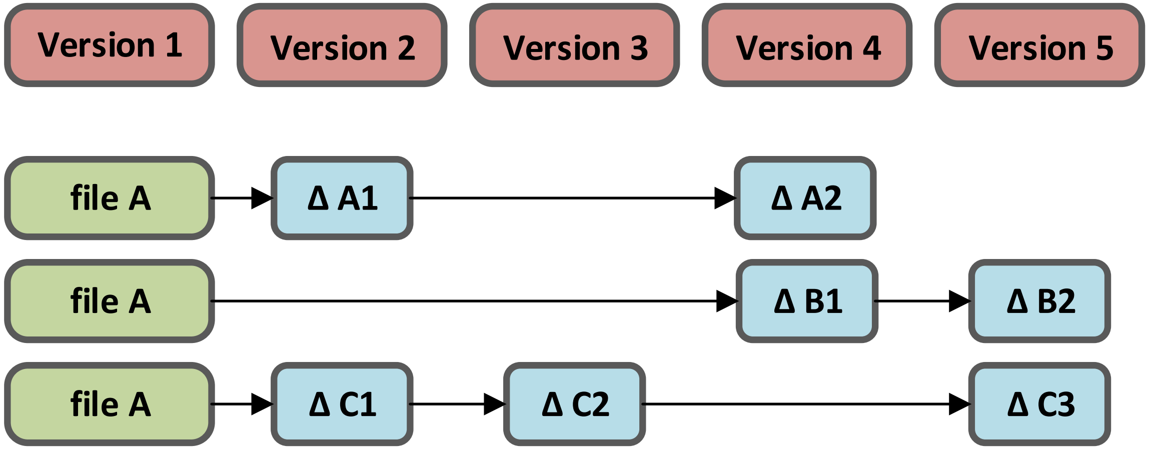
Version 1
Version 2
Version 3
Version 4
Version 5

file A
Δ A1
Δ A2

file A
Δ B1
Δ B2

file A
Δ C1
Δ C2
Δ C3

Delta Storage



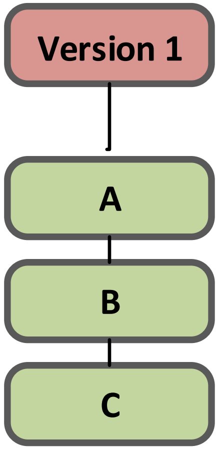
Version 1 Version 2 Version 3 Version 4 Version 5

file A → Δ A1 → Δ A2

file A → Δ B1 → Δ B2

file A → Δ C1 → Δ C2 → Δ C3

DAG Storage



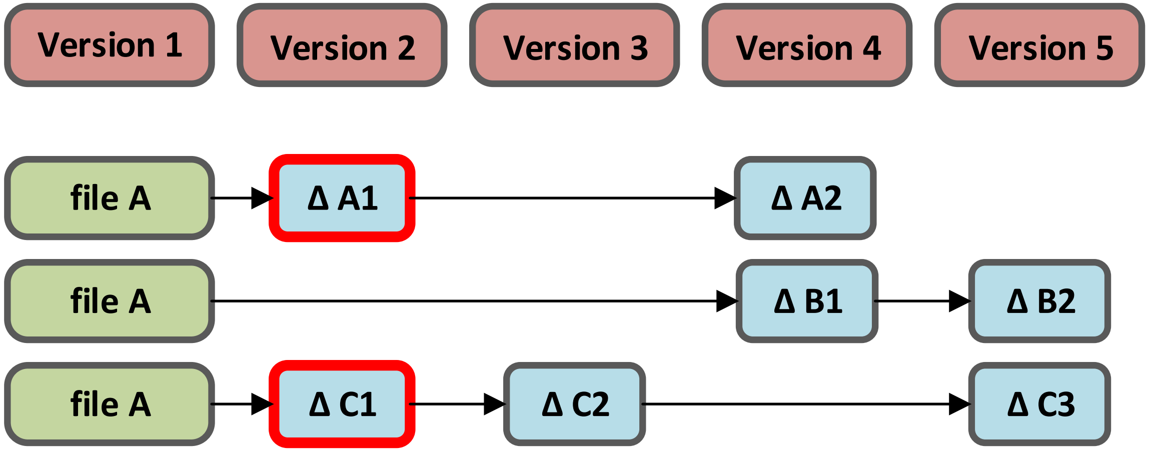
Version 1

A

B

C

Delta Storage



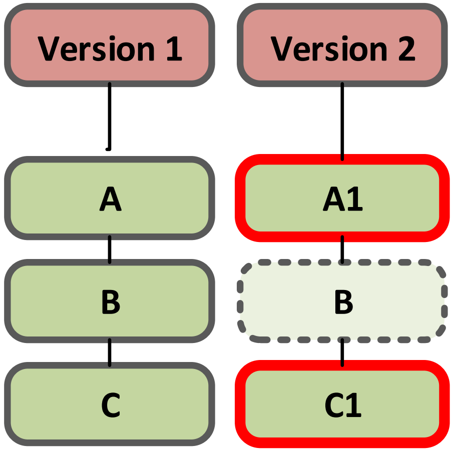
Version 1
Version 2
Version 3
Version 4
Version 5

file A
Δ A1
Δ A2

file A
Δ B1
Δ B2

file A
Δ C1
Δ C2
Δ C3

DAG Storage



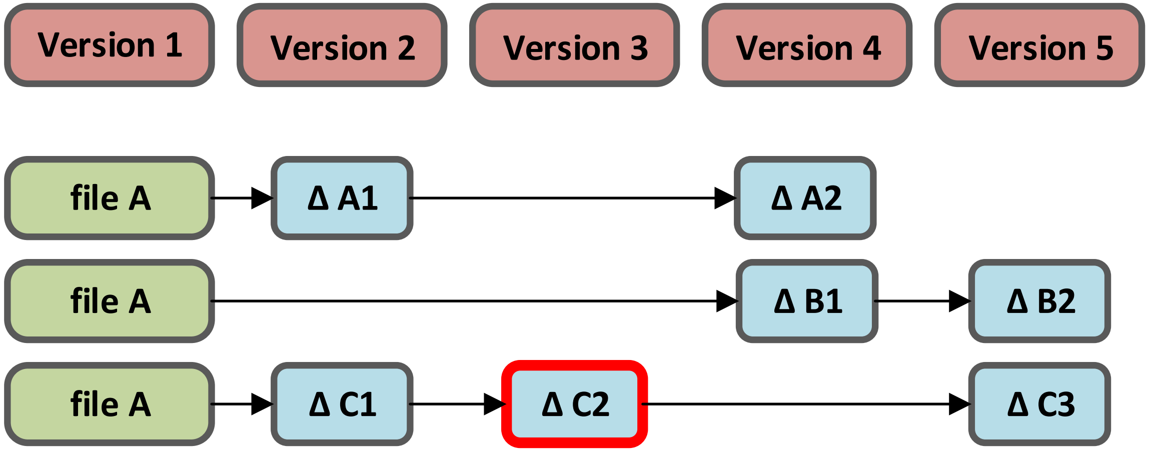
Version 1
Version 2

A
A1

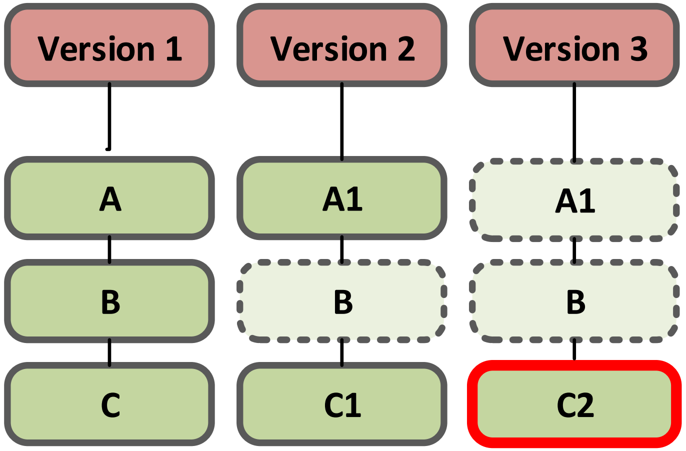
B
B

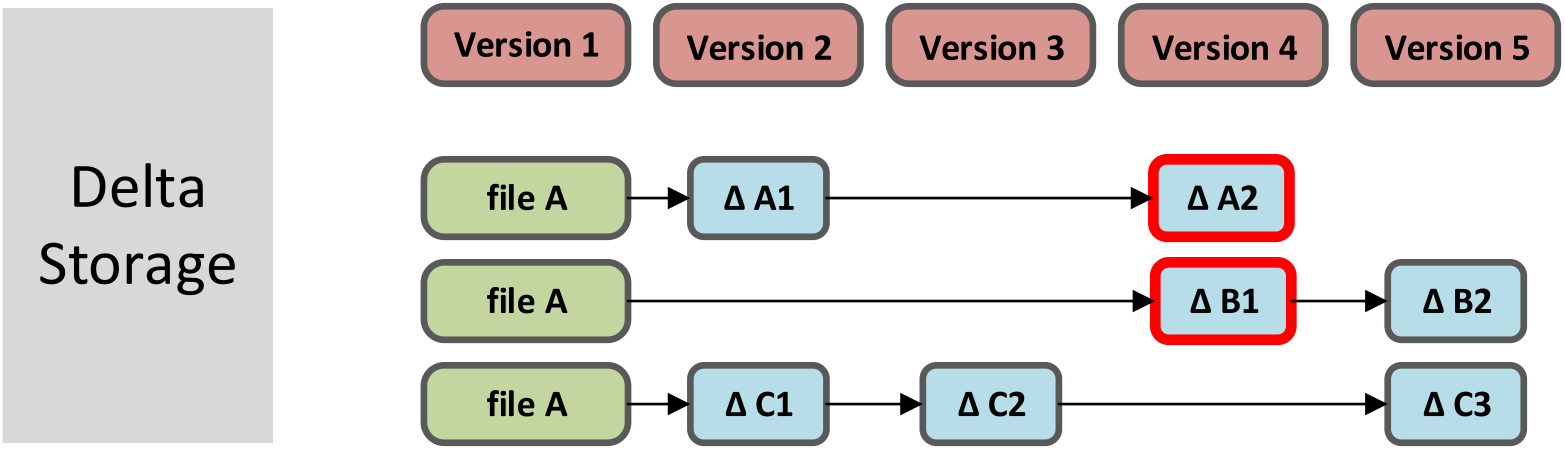
C
C1

Delta Storage



DAG Storage





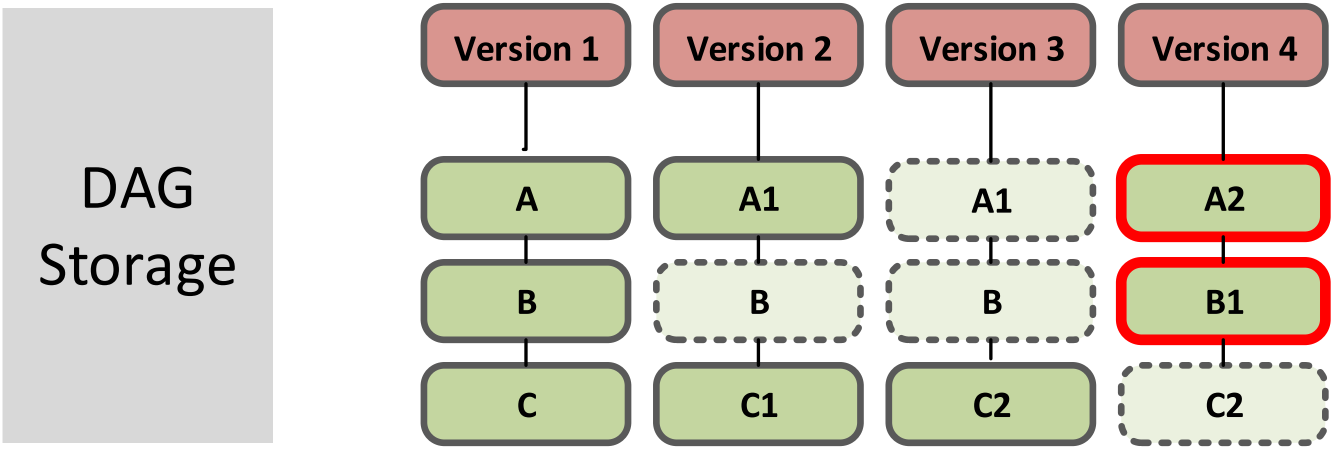
Delta Storage

Version 1 | Version 2 | Version 3 | Version 4 | Version 5

file A → Δ A1 → Δ A2

file A → Δ B1 → Δ B2

file A → Δ C1 → Δ C2 → Δ C3



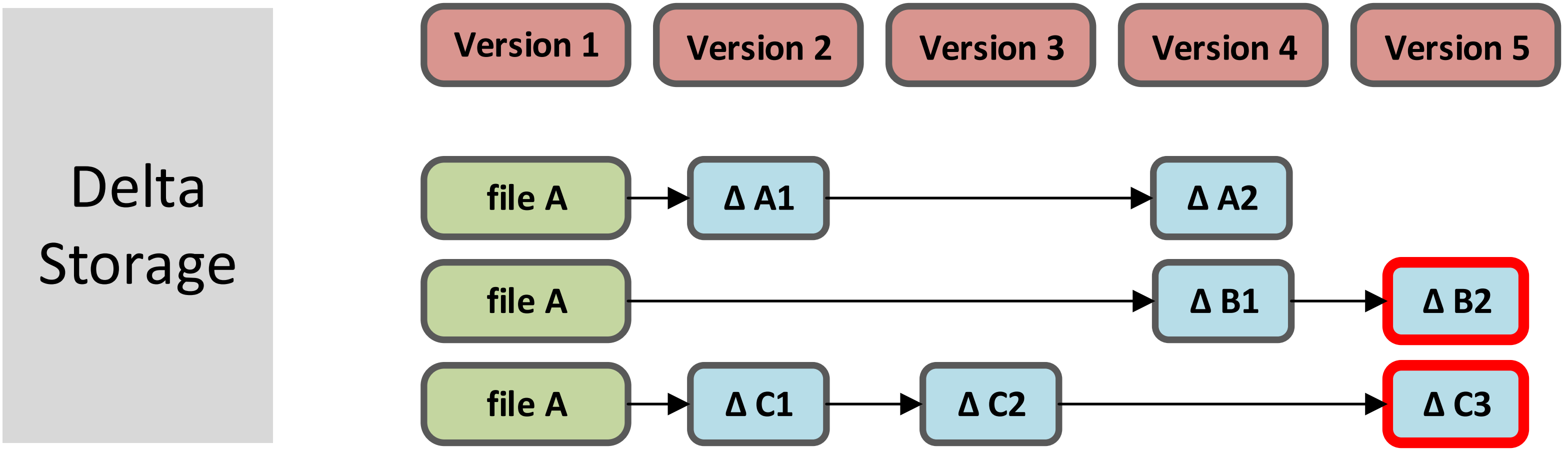
DAG Storage

Version 1: A, B, C

Version 2: A1, B, C1

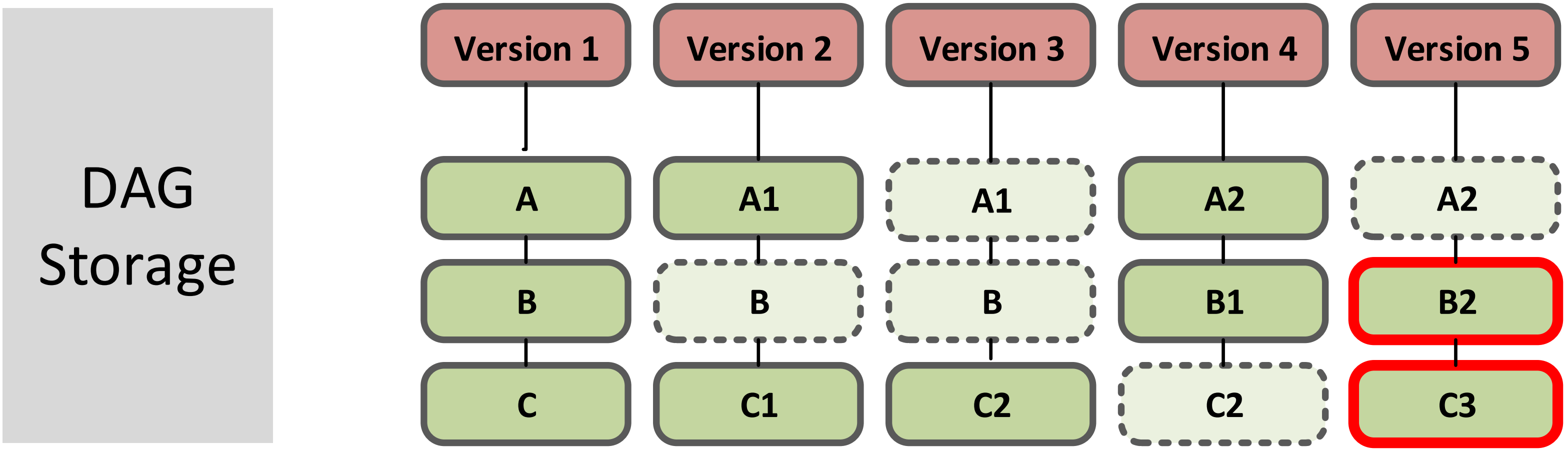
Version 3: A1, B, C2

Version 4: A2, B1, C2



Delta Storage

Version 1	Version 2	Version 3	Version 4	Version 5
file A	Δ A1		Δ A2	
file A			Δ B1	Δ B2
file A	Δ C1	Δ C2		Δ C3



DAG Storage

Version 1	Version 2	Version 3	Version 4	Version 5
A	A1	A1	A2	A2
B	B	B	B1	B2
C	C1	C2	C2	C3

Some Benchmarking

Operation	Git	SVN	
Add, commit and push 113 modified files (2164+, 2259-)	0.64	2.60	4x
Add, commit and push 1000 1k images	1.53	24.70	16x
Diff 187 changed files (1664+, 4859-) against last commit	0.25	1.09	4x
Diff against 4 commits back (269 changed/3609+,6898-)	0.25	3.99	16x
Diff two tags against each other	1.17	83.57	71x
Log of the last 50 commits (19k of output)	0.01	0.38	31x
Log of all commits (26,056 commits - 9.4M of output)	0.52	169.20	325x
Log of the history of a single file	0.60	82.84	138x
Pull of Commit A scenario (113 files changed, 2164+, 2259-)	0.90	2.82	3x
Line annotation of a single file (array.c)	1.91	3.04	1x

http://git-scm.com/about/small-and-fast

Git Object Database

blob

tree

commit

tag

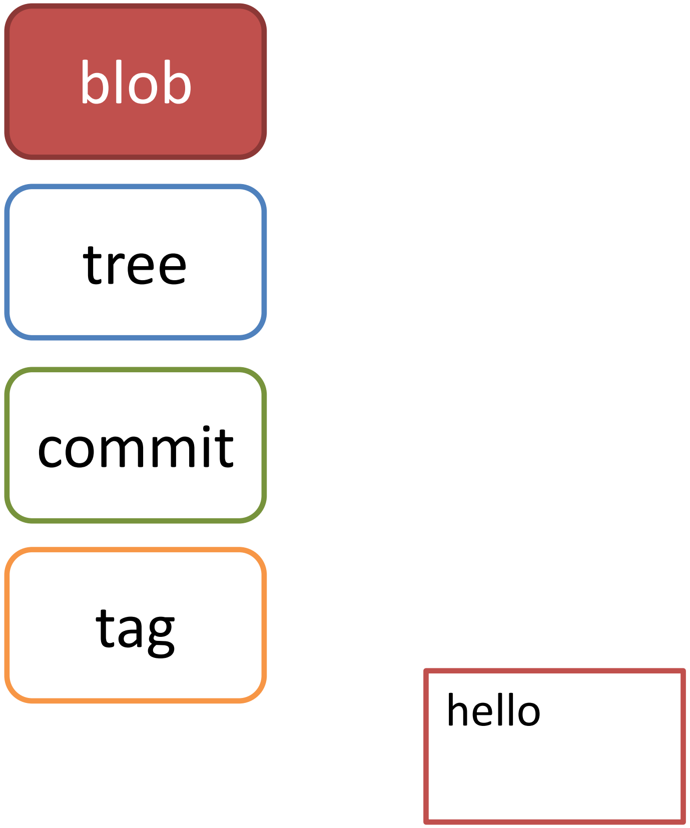
Git Object Database

- blob
- tree
- commit
- tag

```
$ git init
$ tree .git -L 1
.git
├── branches
├── config
├── description
├── HEAD
├── hooks
├── info
├── objects
└── refs
```

Git Object Database

```
$ echo hello > README
```



blob

tree

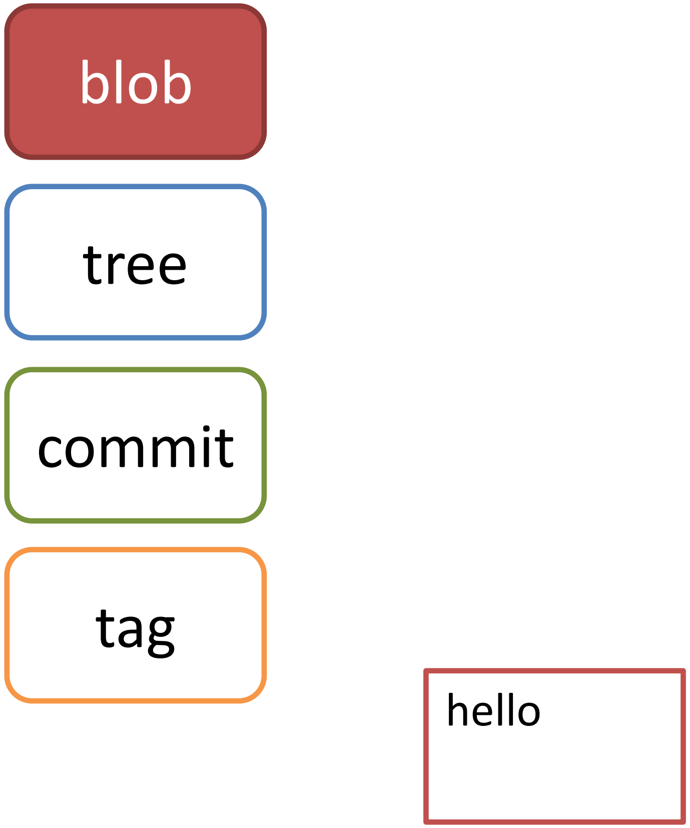
commit

tag

hello

Git Object Database

```
$ echo hello > README
$ git add README
```



blob

tree

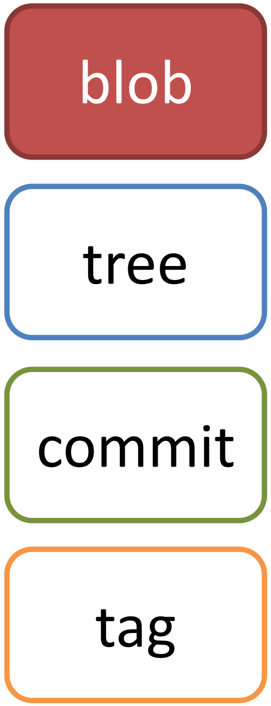
commit

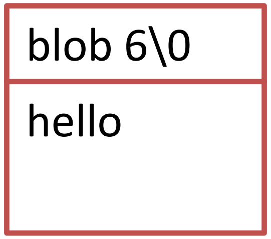
tag

hello

Git Object Database

```
$ echo hello > README
$ git add README
$ find .git/objects -type f
.git/objects/ce/013625030ba8dba906f756967f9e9ca394464a
```





Git Object Database

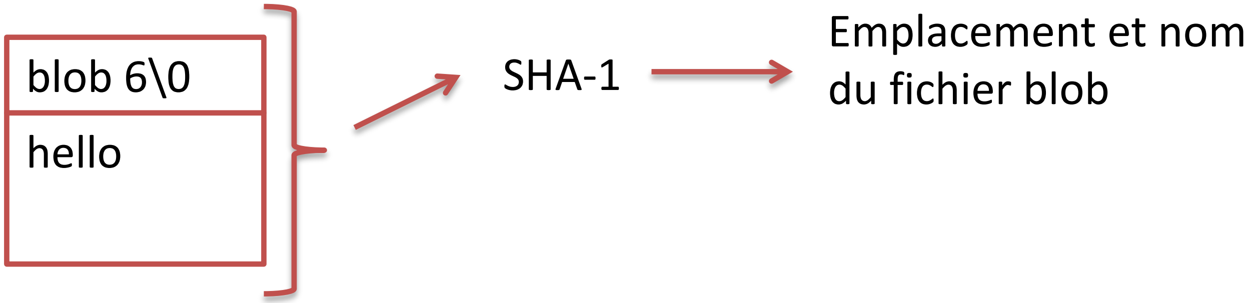
blob

tree

commit

tag

```
$ echo hello > README
$ git add README
$ find .git/objects -type f
.git/objects/ce/013625030ba8dba906f756967f9e9ca394464a

$ printf "blob 6\0hello\n" | sha1sum
ce013625030ba8dba906f756967f9e9ca394464a  -
```



blob 6\0

hello

SHA-1

Emplacement et nom
du fichier blob

Git Object Database

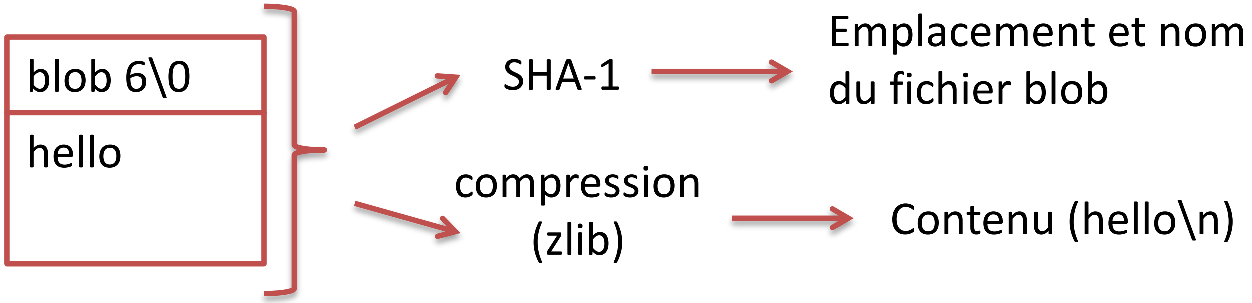
blob

tree

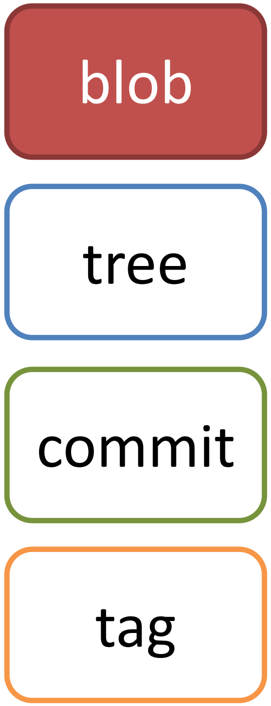
commit

tag

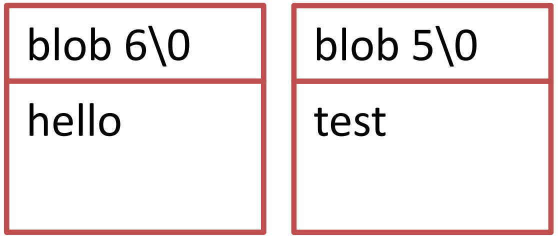
```
$ echo hello > README
$ git add README
$ find .git/objects -type f
.git/objects/ce/013625030ba8dba906f756967f9e9ca394464a

$ printf "blob 6\0hello\n" | sha1sum
ce013625030ba8dba906f756967f9e9ca394464a  -

$ git cat-file -p ce013625
hello
```



Git Object Database



```
$ echo hello > README
$ git add README
$ find .git/objects -type f
.git/objects/ce/013625030ba8dba906f756967f9e9ca394464a

$ printf "blob 6\0hello\n" | sha1sum
ce013625030ba8dba906f756967f9e9ca394464a  -

$ git cat-file -p ce013625
hello

$ cp README file2.txt
$ echo test > file3.txt
$ git add file2.txt file3.txt
```



Git Object Database

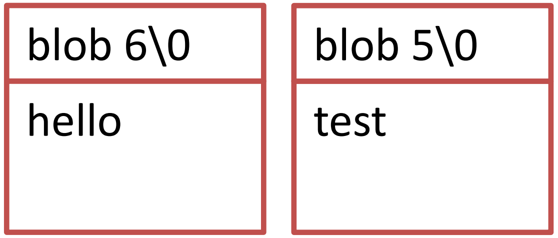
blob

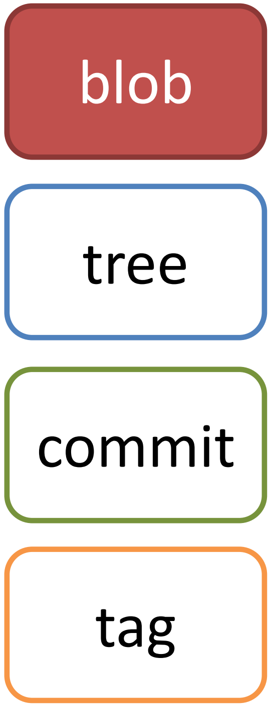
tree

commit

tag

```
$ echo hello > README
$ git add README
$ find .git/objects -type f
.git/objects/ce/013625030ba8dba906f756967f9e9ca394464a

$ printf "blob 6\0hello\n" | sha1sum
ce013625030ba8dba906f756967f9e9ca394464a  -

$ git cat-file -p ce013625
hello

$ cp README file2.txt
$ echo test > file3.txt
$ git add file2.txt file3.txt

$ find .git/objects -type f
.git/objects/9d/aeafb9864cf43055ae93beb0afd6c7d144bfa4
.git/objects/ce/013625030ba8dba906f756967f9e9ca394464a
```





```
$ tree .git/
.git/
├── branches
├── config
├── description
├── HEAD
├── hooks
│   ├── applypatch-msg.sample
│   ├── … etc …
│   └── update.sample
├── index
├── info
│   └── exclude
├── objects
│   ├── 9d
│   │   └── aeafb9864cf43055ae93beb0afd6c7d144bfa4
│   ├── ce
│   │   └── 013625030ba8dba906f756967f9e9ca394464a
│   ├── info
│   └── pack
└── refs
    ├── heads
    └── tags
```

contenu de file3.txt

contenu de README **=** **contenu de file2.txt**

Git Object Database

```
$ git commit -m "Mon premier commit"
$ git log
commit 8e8eb678d9512d421e590350e3100dc51ead6b7a
Author: Sébastien Dawans
<sebastien.dawans@cetic.be>
Date:   Sun Apr 14 22:26:29 2013 +0200

    Mon premier commit
```

blob

tree

commit

tag

Git Object Database

blob

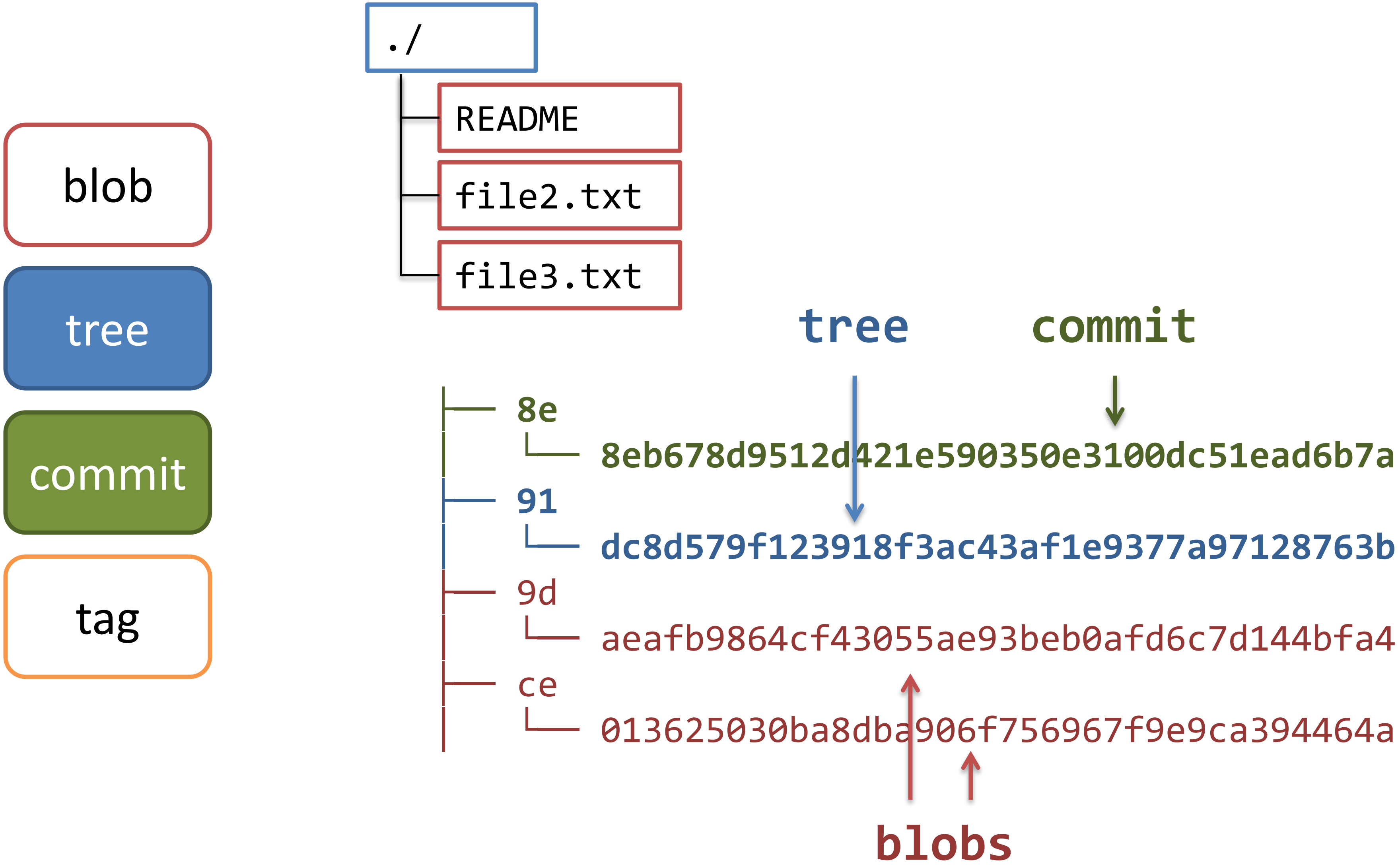
tree

commit

tag

```
$ git commit -m "Mon premier commit"
$ git log
commit 8e8eb678d9512d421e590350e3100dc51ead6b7a
Author: Sébastien Dawans
<sebastien.dawans@cetic.be>
Date:   Sun Apr 14 22:26:29 2013 +0200

    Mon premier commit

$ tree .git/objects/
.git/objects/
├── 8e
│   └── 8eb678d9512d421e590350e3100dc51ead6b7a
├── 91
│   └── dc8d579f123918f3ac43af1e9377a97128763b
├── 9d
│   └── aeafb9864cf43055ae93beb0afd6c7d144bfa4
├── ce
│   └── 013625030ba8dba906f756967f9e9ca394464a
```

Git Object Database



blob

tree

commit

tag

```
./
├── README
├── file2.txt
└── file3.txt
```

tree

commit

```
├── 8e
│   └── 8eb678d9512d421e590350e3100dc51ead6b7a
├── 91
│   └── dc8d579f123918f3ac43af1e9377a97128763b
├── 9d
│   └── aeafb9864cf43055ae93beb0afd6c7d144bfa4
├── ce
│   └── 013625030ba8dba906f756967f9e9ca394464a
```

blobs

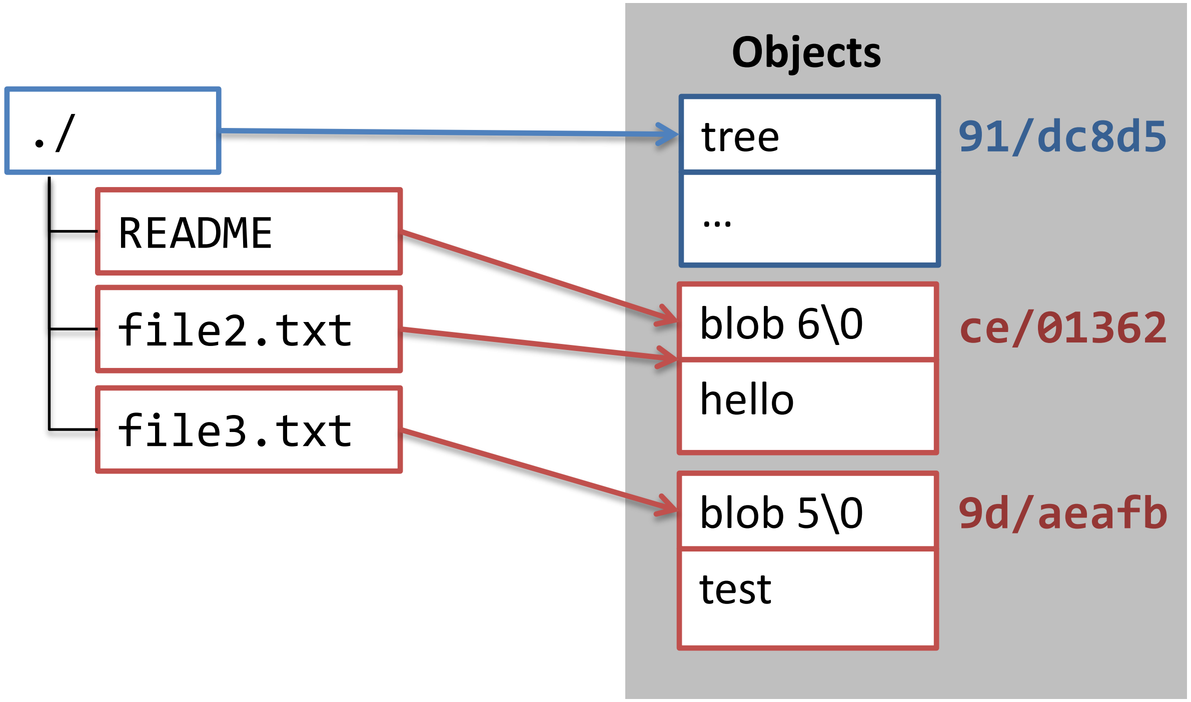
blob

tree

commit

tag

Git Object Database



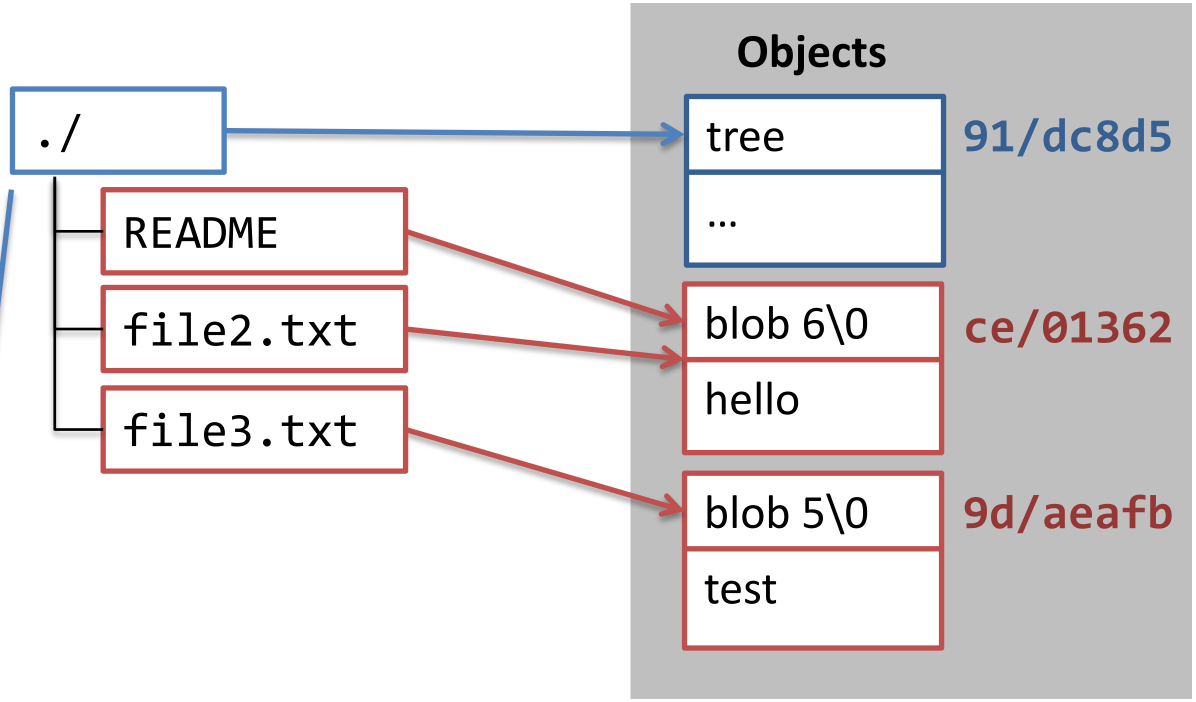
blob

tree

commit

tag

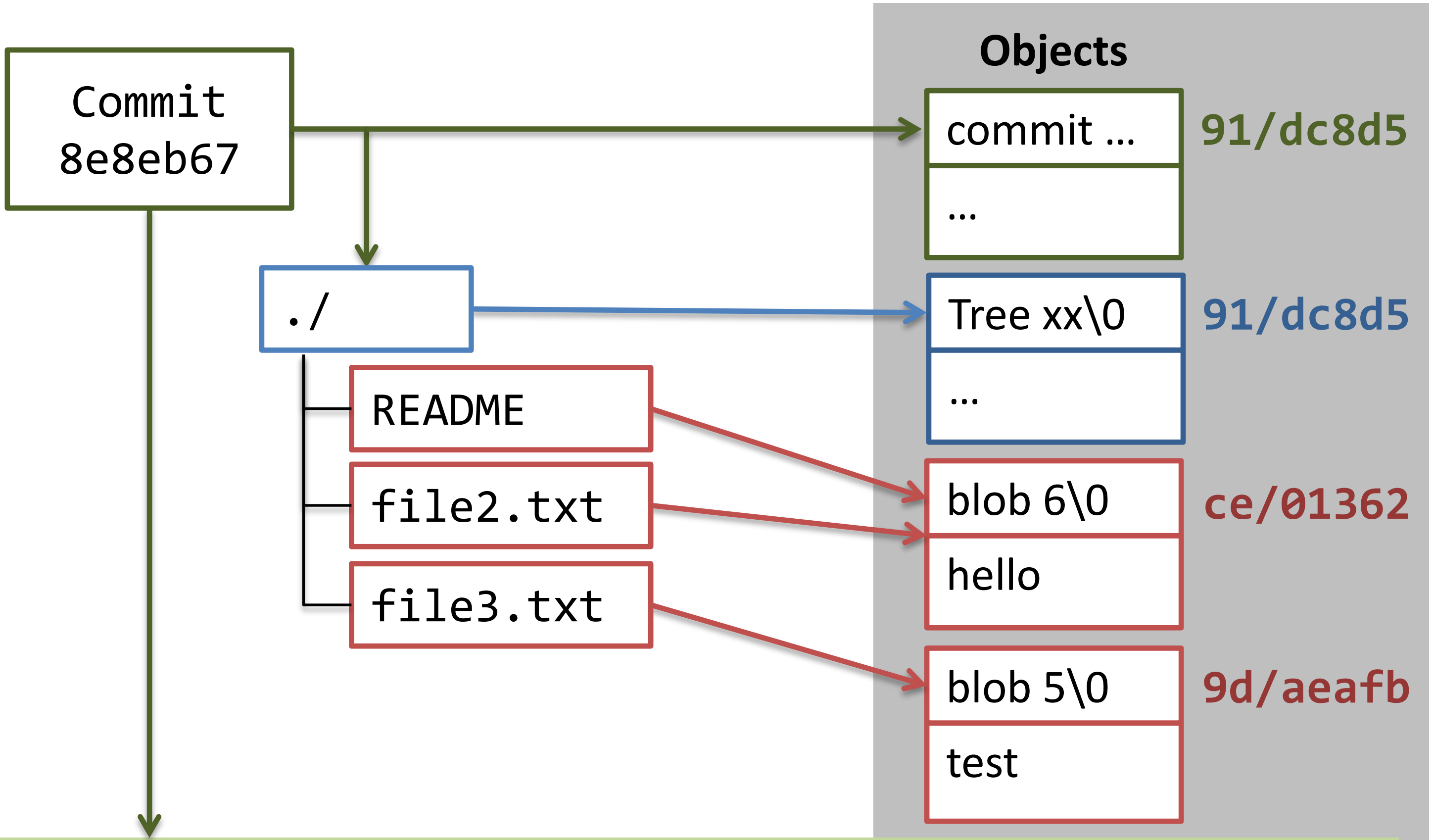
Git Object Database



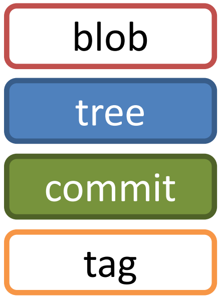
```
$ git cat-file -p 91dc8d5
100644 blob ce013625030ba8dba906f756967f9e9ca394464a     README
100644 blob ce013625030ba8dba906f756967f9e9ca394464a     file2.txt
100644 blob 9daeafb9864cf43055ae93beb0afd6c7d144bfa4     file3.txt
```

- blob
- tree
- commit
- tag

Git Object Database



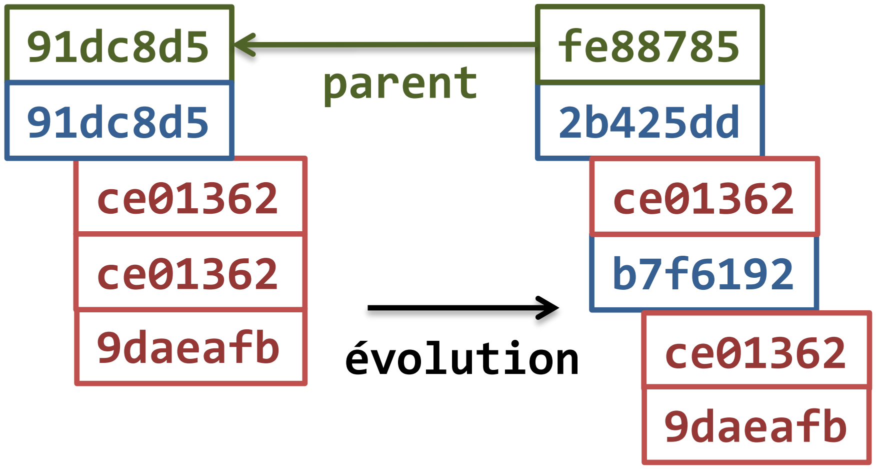
```
$ git cat-file -p 8e8eb67
tree 91dc8d579f123918f3ac43af1e9377a97128763b
author Sébastien Dawans <sebastien.dawans@cetic.be> 1365971189 +0200
committer Sébastien Dawans <sebastien.dawans@cetic.be> 1365971189 +0200

Mon premier commit
```

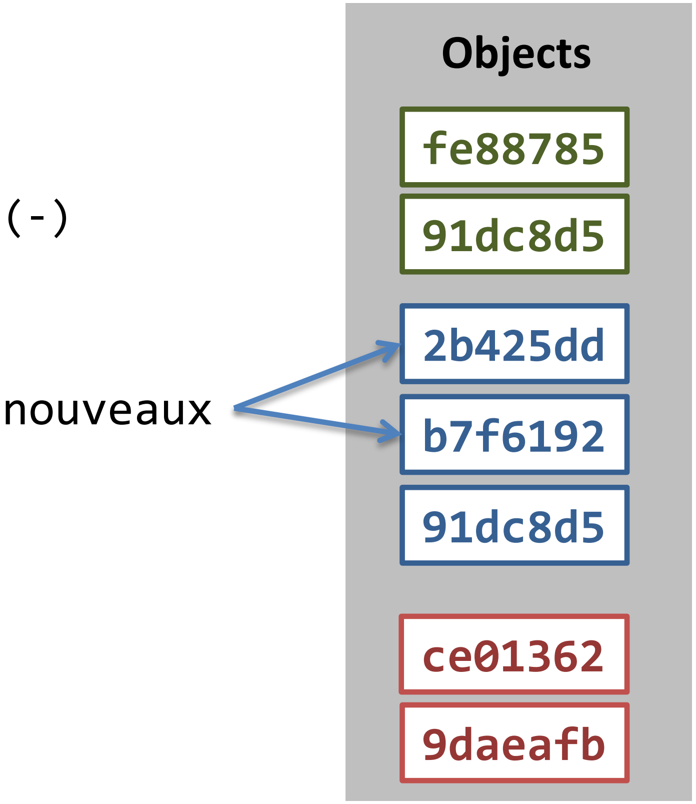


Reuse d’objets

- Déplacement de fichiers = renommage
  - > contenu ne change pas

```
$ mkdir doc
$ git mv file2.txt doc
$ git mv file3.txt doc
$ git commit -m "Deplacement de fichiers"
[master fe88785] Deplacement de fichiers
 2 files changed, 0 insertions(+), 0 deletions(-)
 rename file2.txt => doc/file2.txt (100%)
 rename file3.txt => doc/file3.txt (100%)
```



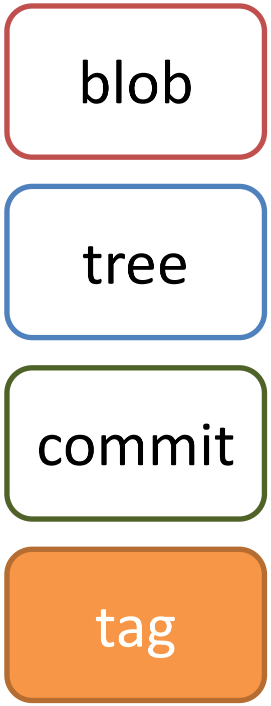


blob
tree
commit
tag

Reuse d’objets

- Réutilisation d’objets similaire
- Heuristiques poussées pour « packer » les nouvelles données par rapport à l’existant

Last but not least: Tags



blob

tree

commit

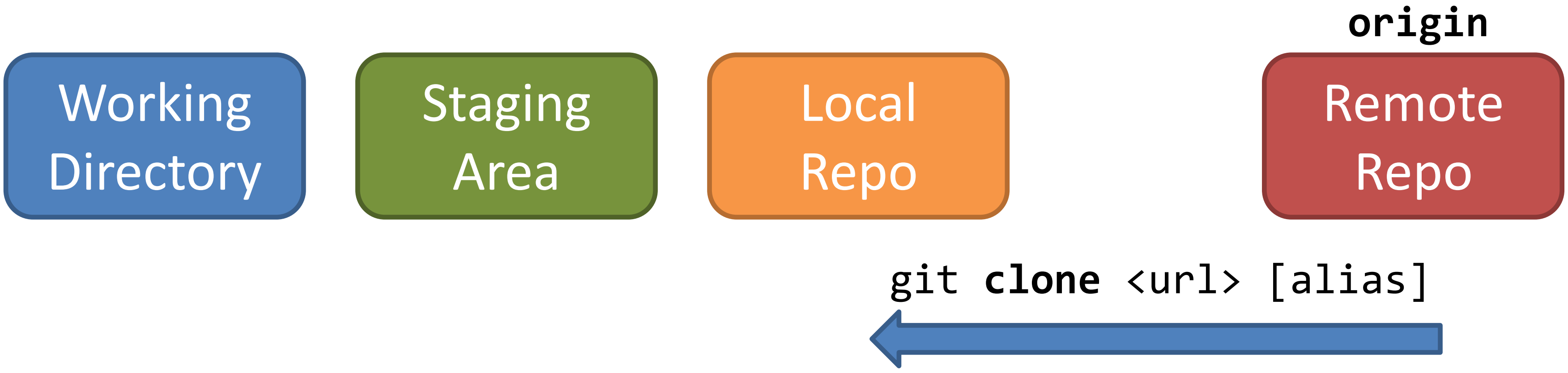
tag

Tag = Pointeur sur un commit

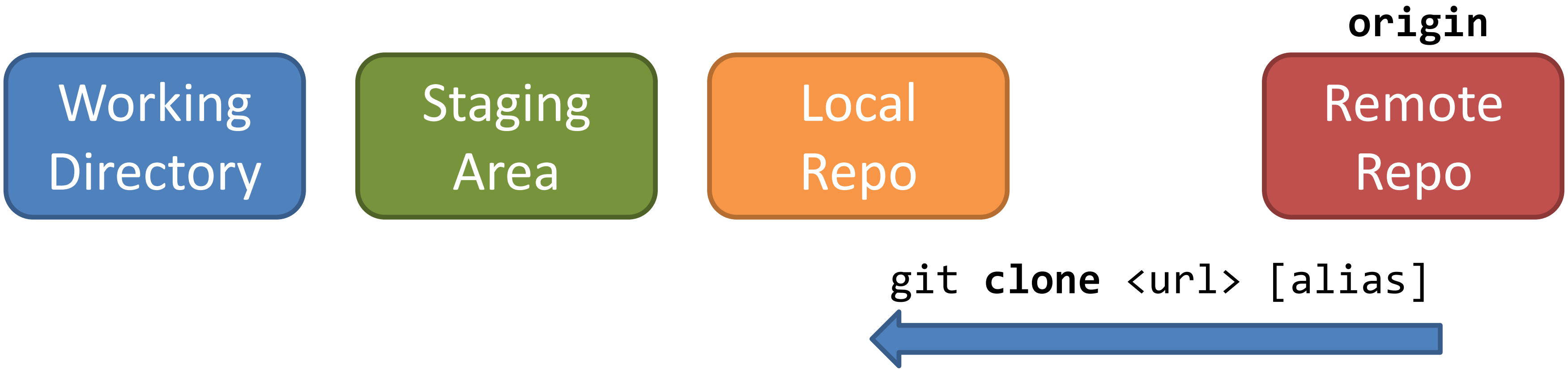
- Suivi de releases, mais aussi:
- Tags de développement,

Points de synchro

Using Git



```
$ git clone https://github.com/contiki-os/contiki.git
Cloning into 'contiki'...
remote: Counting objects: 67870, done.
remote: Compressing objects: 100% (13454/13454), done.
remote: Total 67870 (delta 49179), reused 67358 (delta 48872)
Receiving objects: 100% (67870/67870), 51.40 MiB | 5.15 MiB/s, done.
Resolving deltas: 100% (49179/49179), done.
```



Working Directory

Staging Area

Local Repo

origin

Remote Repo

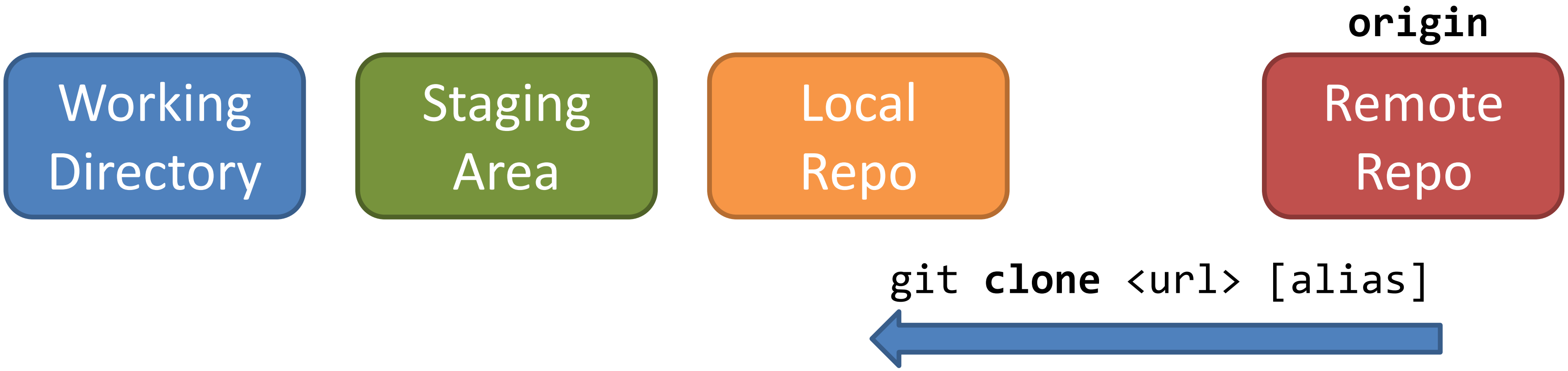
`git clone <url> [alias]`

```
$ git clone https://github.com/contiki-os/contiki.git
Cloning into 'contiki'...
remote: Counting objects: 67870, done.
remote: Compressing objects: 100% (13454/13454), done.
remote: Total 67870 (delta 49179), reused 67358 (delta 48872)
Receiving objects: 100% (67870/67870), 51.40 MiB | 5.15 MiB/s, done.
Resolving deltas: 100% (49179/49179), done.

$ cd contiki
$ ls -a
```

Local Repo

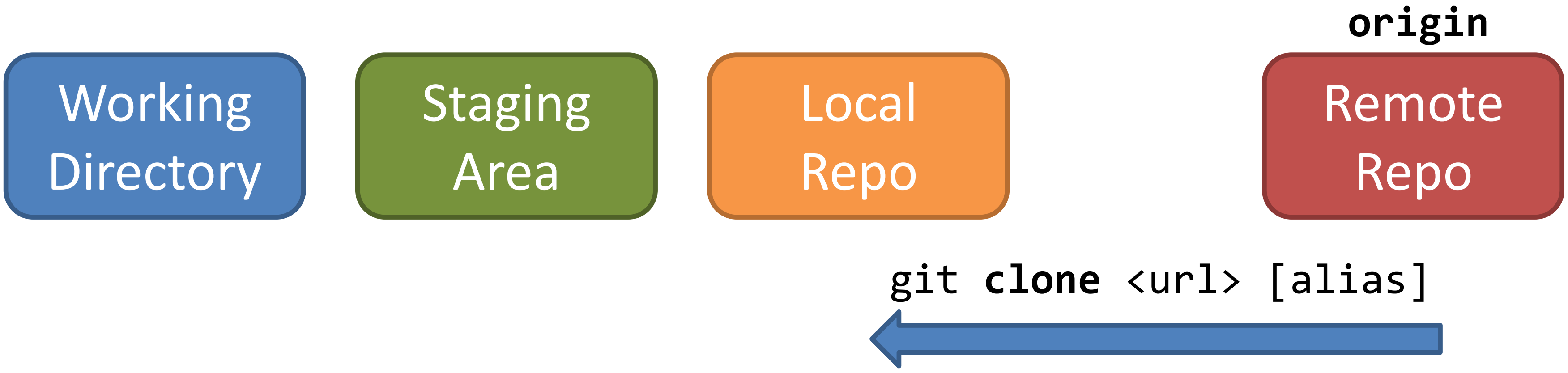
Working Directory

```
. .. apps core cpu doc examples .git .gitignore LICENSE
Makefile.include platform README README-BUILDING README-EXAMPLES
regression-tests
```



`git clone <url> [alias]`

```
$ git remote -v
origin https://github.com/contiki-os/contiki.git (fetch)
origin https://github.com/contiki-os/contiki.git (push)
```



origin

Working Directory

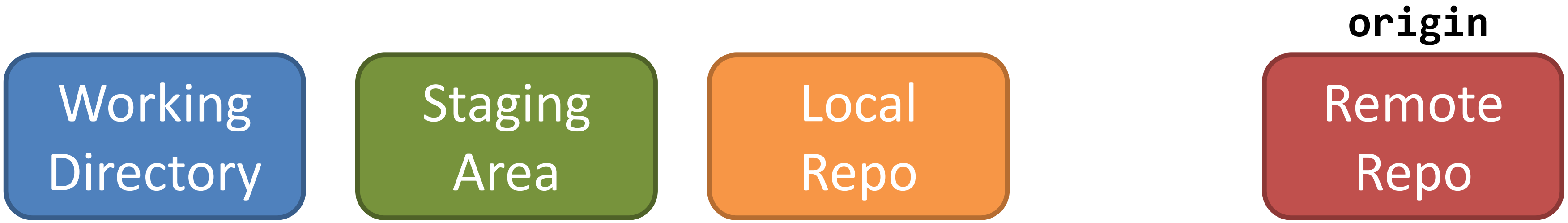
Staging Area

Local Repo

Remote Repo

git **clone** <url> [alias]

```
$ git remote –v
origin https://github.com/contiki-os/contiki.git (fetch)
origin https://github.com/contiki-os/contiki.git (push)

$ git branch
* master

$ git branch –a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master

$ git log –n 3 –oneline
424a7b2 Merge pull request #202 from g-oikonomou/cc2538-minor-fixes
704309c Change the InfoPage Location of the IEEE address
8b5b2bd CC2538 Documentation typo and grammar fixes
```



origin

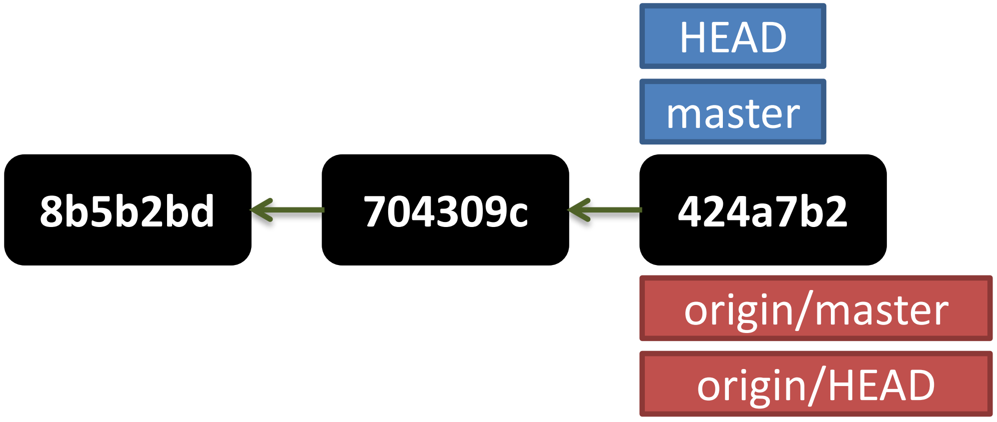
Working Directory

Staging Area

Local Repo

Remote Repo

```
$ git branch -a
* master
remotes/origin/HEAD -> origin/master
remotes/origin/master
```



HEAD

master

8b5b2bd

704309c

424a7b2

origin/master

origin/HEAD

origin

Working Directory

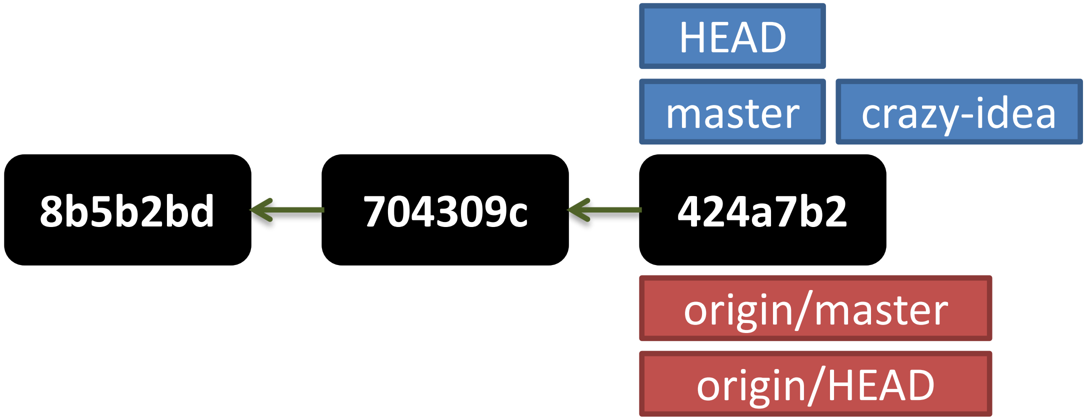
Staging Area

Local Repo

Remote Repo

```
$ git branch crazy-idea
```

```
$ git branch -a
crazy-idea
* master
remotes/origin/HEAD -> origin/master
remotes/origin/master
```



HEAD

master

crazy-idea

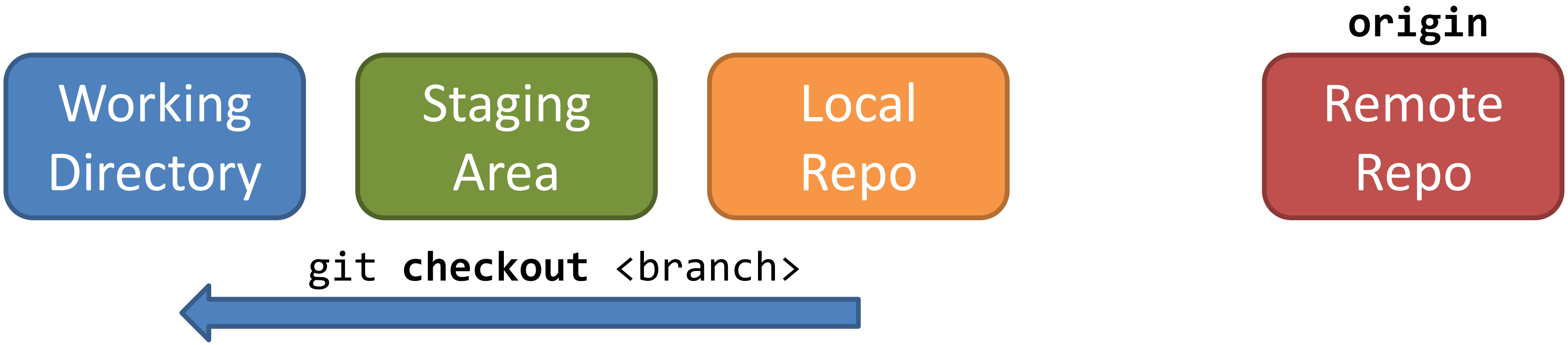
8b5b2bd

704309c

424a7b2

origin/master

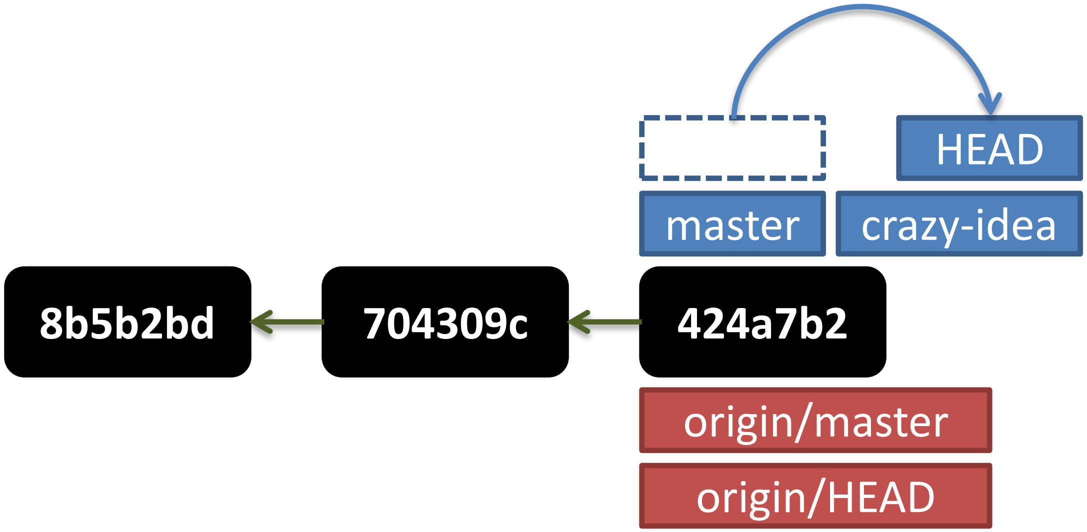
origin/HEAD

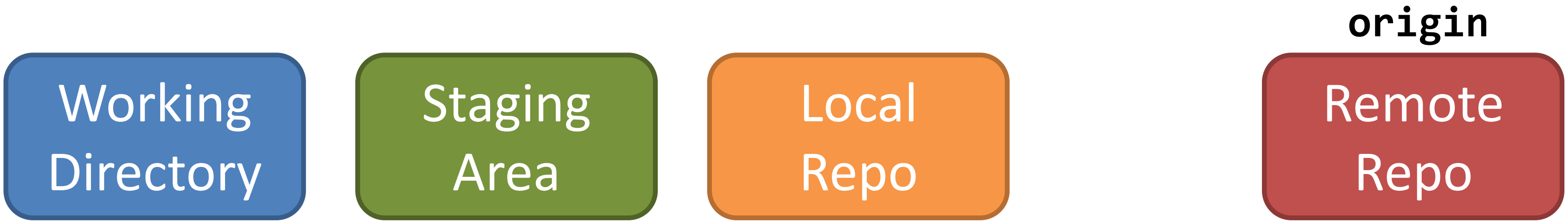


git **checkout** <branch>

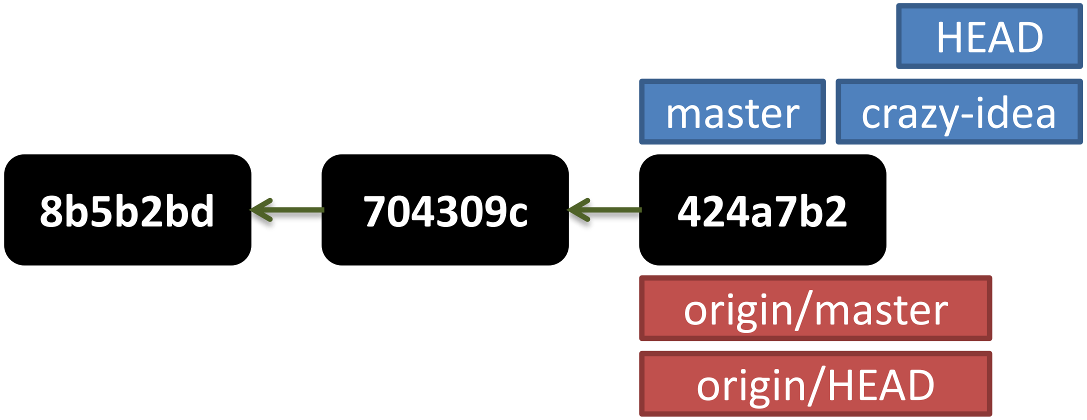
```
$ git branch crazy-idea

$ git checkout crazy-idea
```

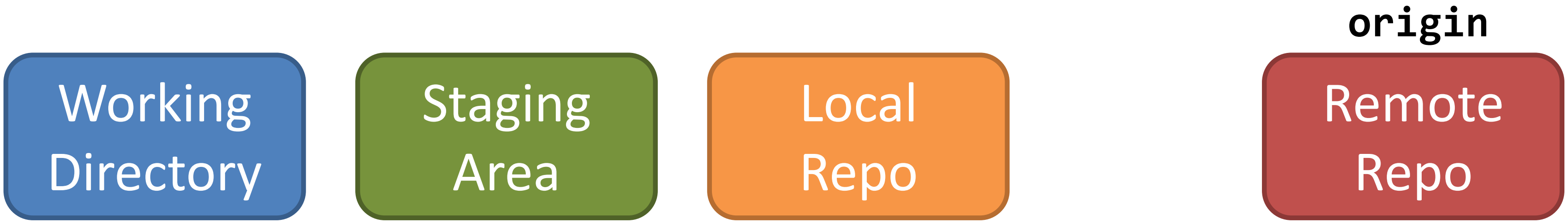
```
$ git branch -a
* crazy-idea
master
remotes/origin/HEAD -> origin/master
remotes/origin/master
```



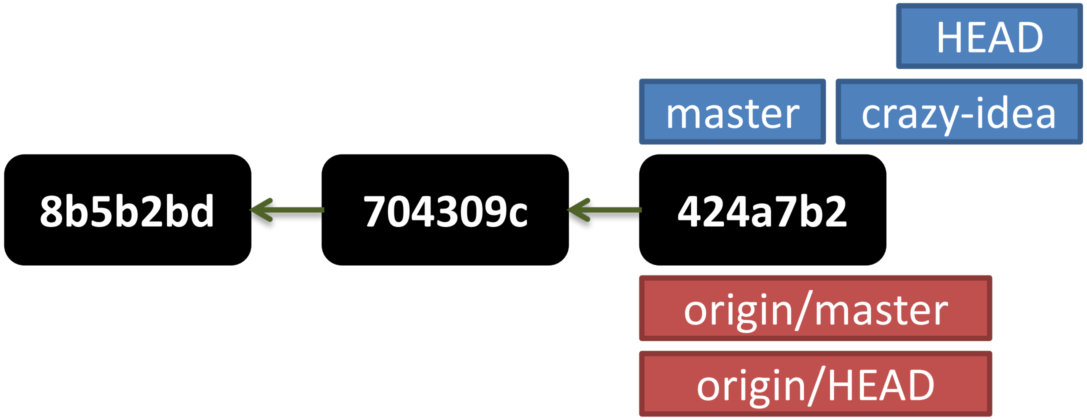


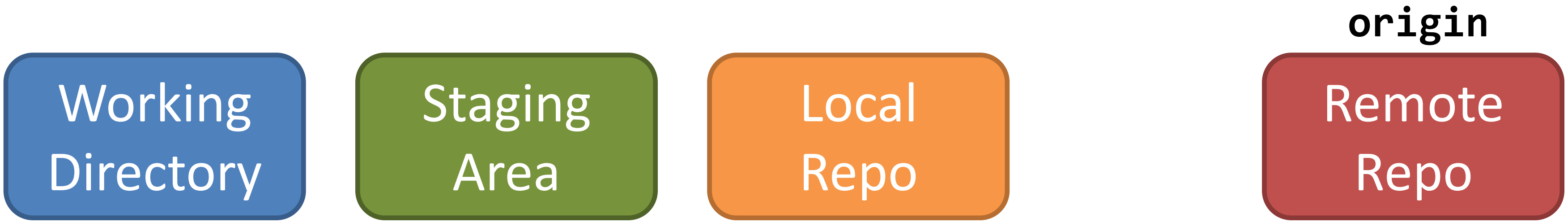
```
$ git status
# On branch crazy-idea
nothing to commit (working directory clean)

... hack your crazy idea: add new files, modify others ...
```





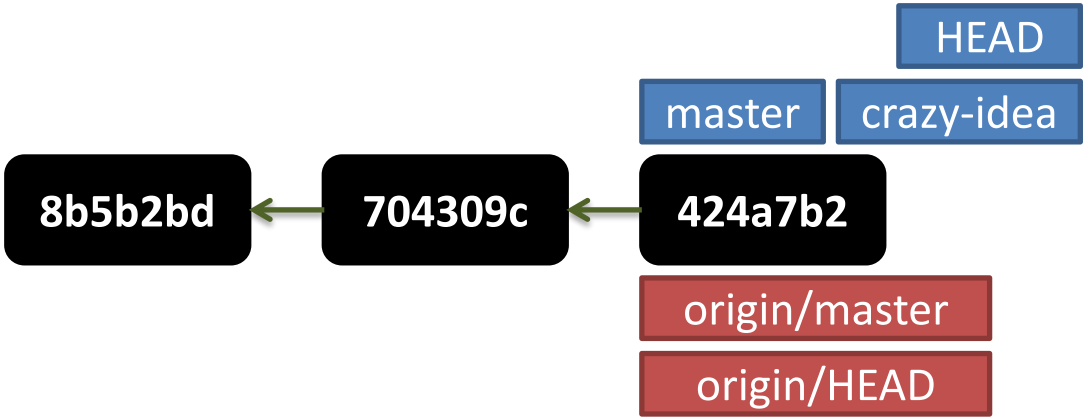
```
$ git status
# On branch crazy-idea
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#        modified:   core/net/rpl/rpl.c
#        modified:   core/net/tcpip.c
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#        core/net/newfile.c
no changes added to commit (use "git add" and/or "git commit -a")
```

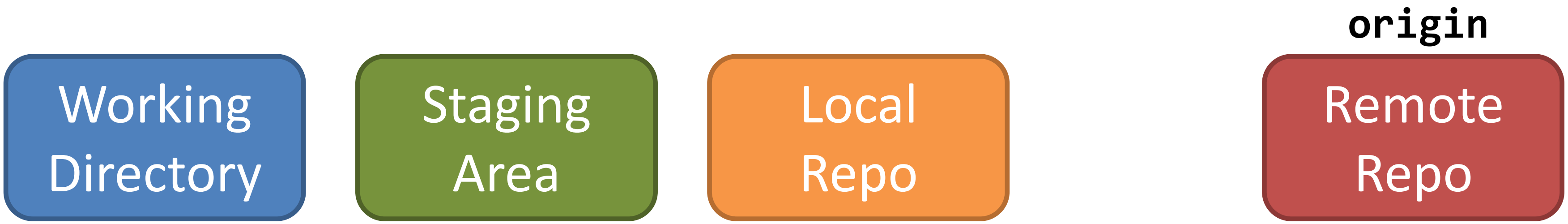




```
$ git status
# On branch crazy-idea
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#       modified:   core/net/rpl/rpl.c
#       modified:   core/net/tcpip.c
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       core/net/newfile.c
no changes added to commit (use "git add" and/or "git commit -a")
```

Modification of tracked content





origin

Working Directory

Staging Area

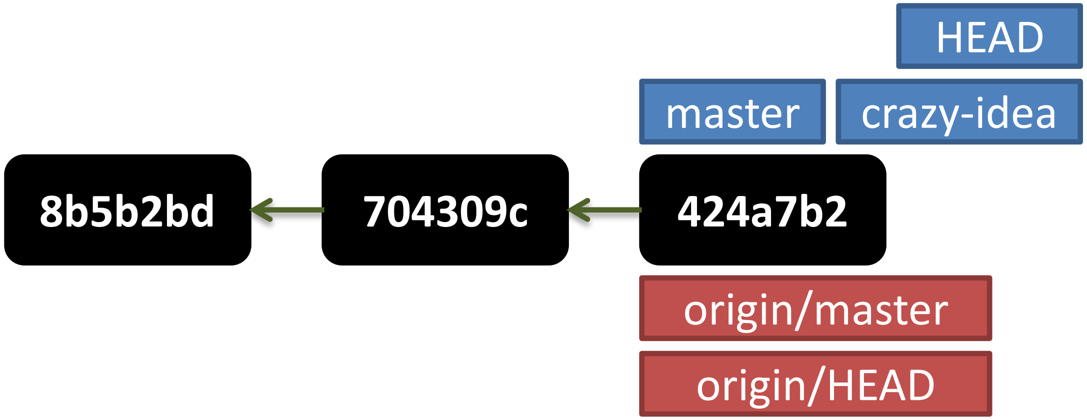
Local Repo

Remote Repo

```
$ git status
# On branch crazy-idea
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#       modified:   core/net/rpl/rpl.c
#       modified:   core/net/tcpip.c
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       core/net/newfile.c
no changes added to commit (use "git add" and/or "git commit -a")
```

Modification of tracked content

New, untracked content



HEAD

master

crazy-idea

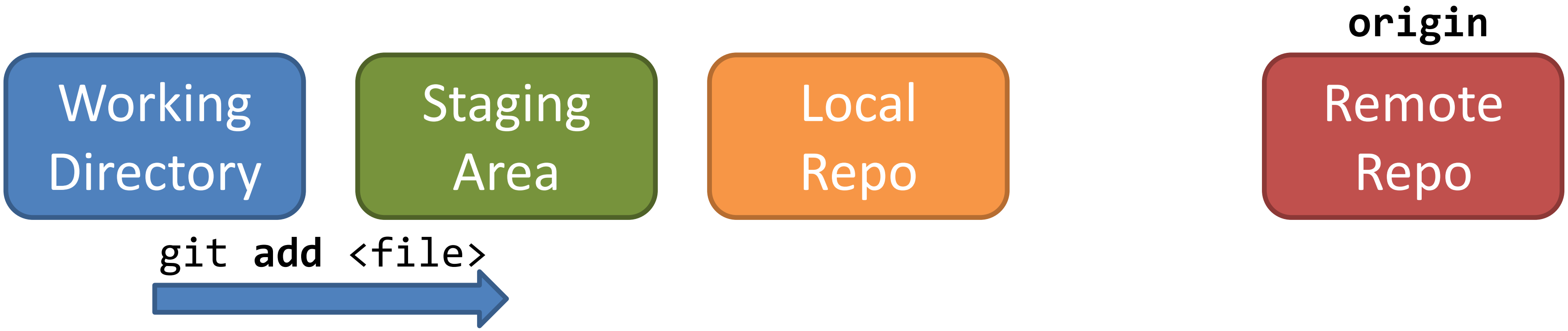
8b5b2bd

704309c

424a7b2

origin/master

origin/HEAD



origin

Working Directory

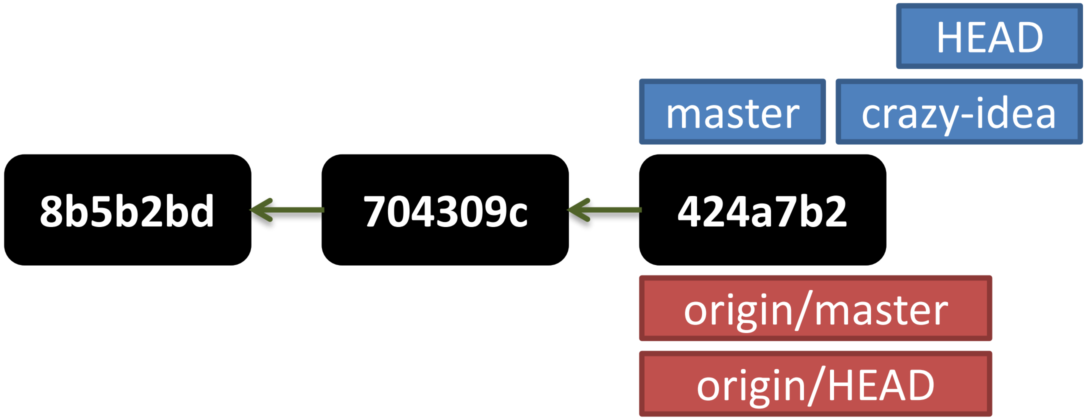
Staging Area

Local Repo

Remote Repo

git **add** <file>

```
$ git add core/net/rpl/rpl.c
```



HEAD

master

crazy-idea

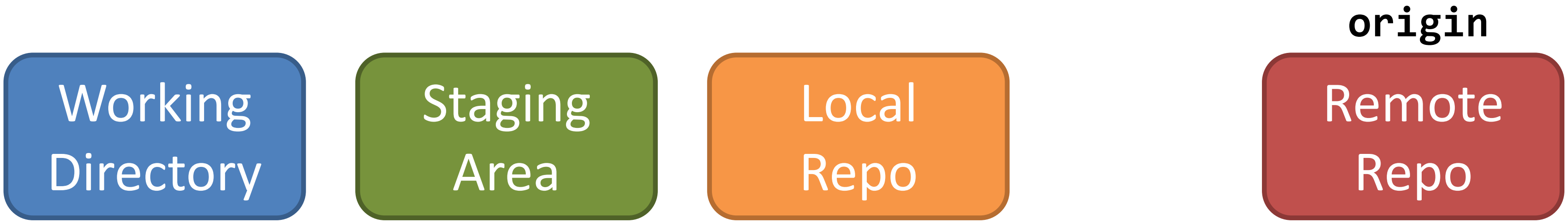
8b5b2bd

704309c

424a7b2

origin/master

origin/HEAD



origin

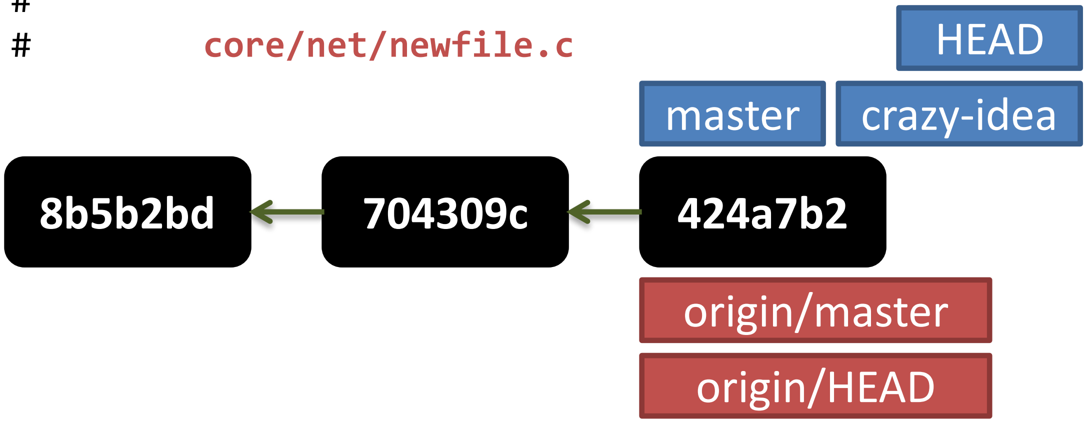
Working Directory

Staging Area

Local Repo

Remote Repo

```
$ git status
# # On branch crazy-idea
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#        modified:   core/net/rpl/rpl.c
#
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#        modified:   core/net/tcpip.c
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#        core/net/newfile.c
```



HEAD

master

crazy-idea

8b5b2bd

704309c

424a7b2

origin/master

origin/HEAD

Working Directory

Staging Area

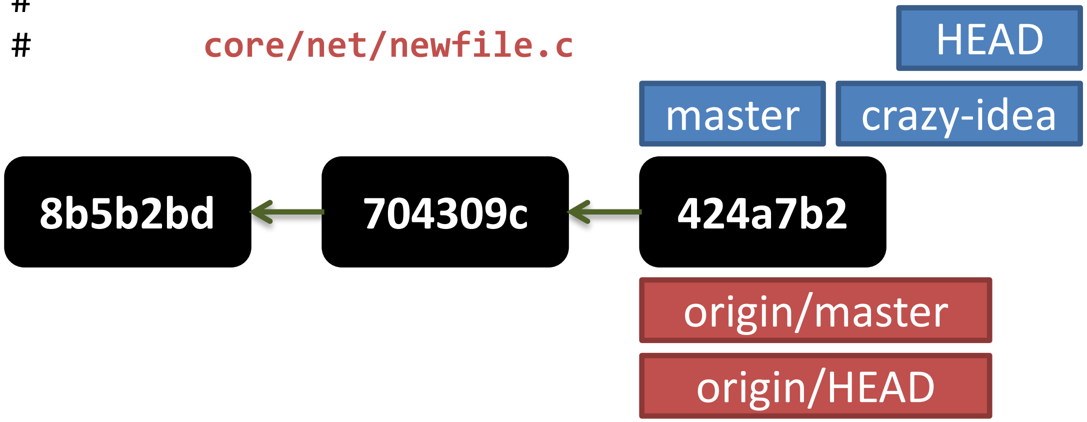
Local Repo

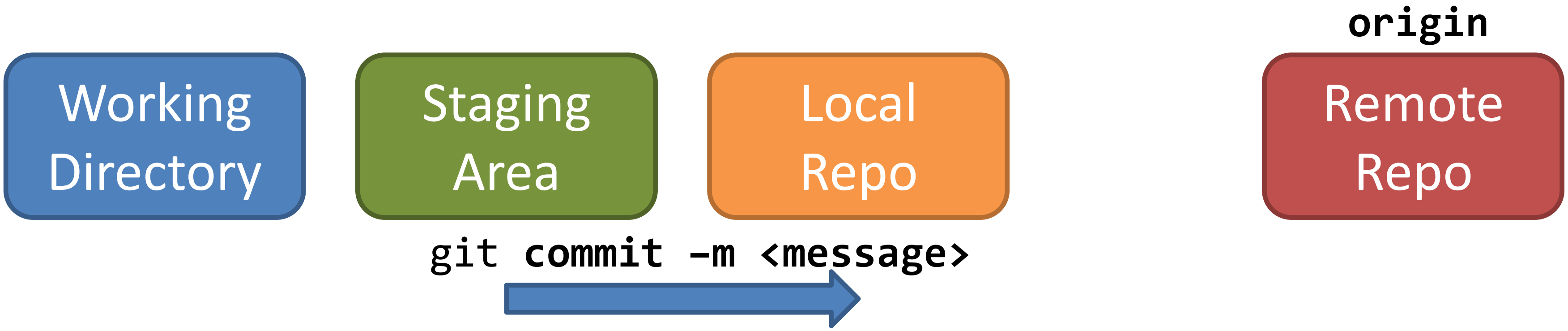
origin

Remote Repo

```
$ git status
# # On branch crazy-idea
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   core/net/rpl/rpl.c
#
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#       modified:   core/net/tcpip.c
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       core/net/newfile.c
```

Content of the next commit





Working Directory

Staging Area

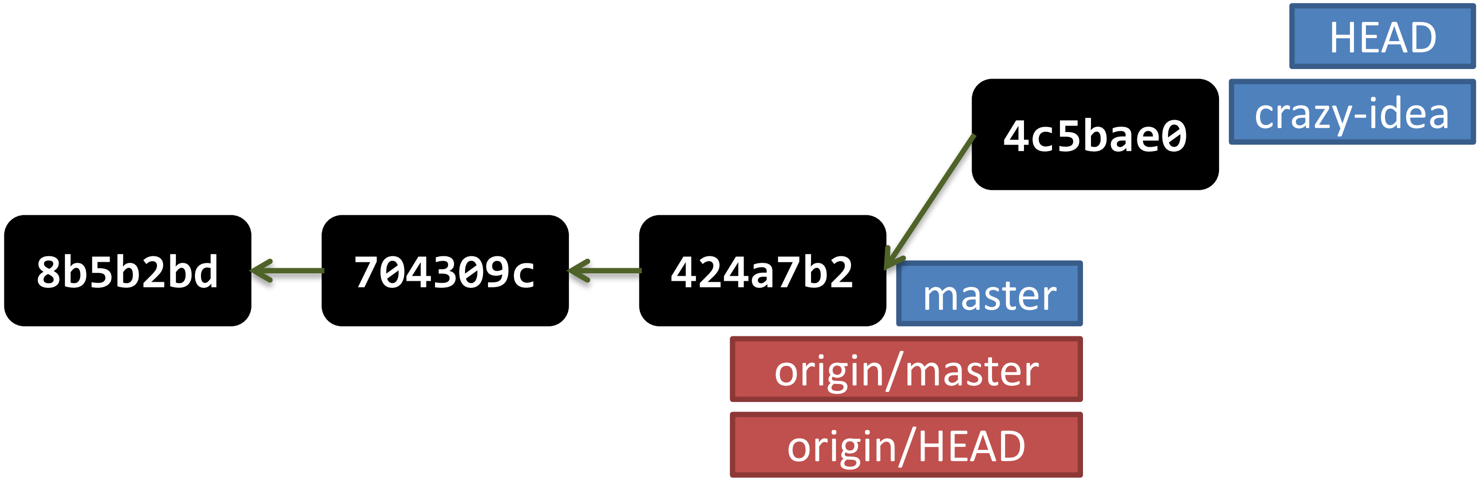
Local Repo

origin

Remote Repo

git **commit** –m <message>

```
$ git commit –m "my crazy idea - part 1«

[crazy-idea 4c5bae0] my crazy idea - part 1
 1 file changed, 2 insertions(+)
```



8b5b2bd

704309c

424a7b2

4c5bae0

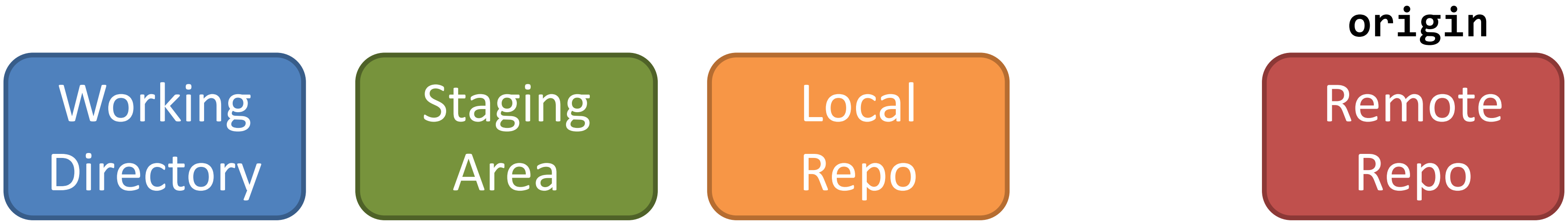
HEAD

crazy-idea

master

origin/master

origin/HEAD



origin

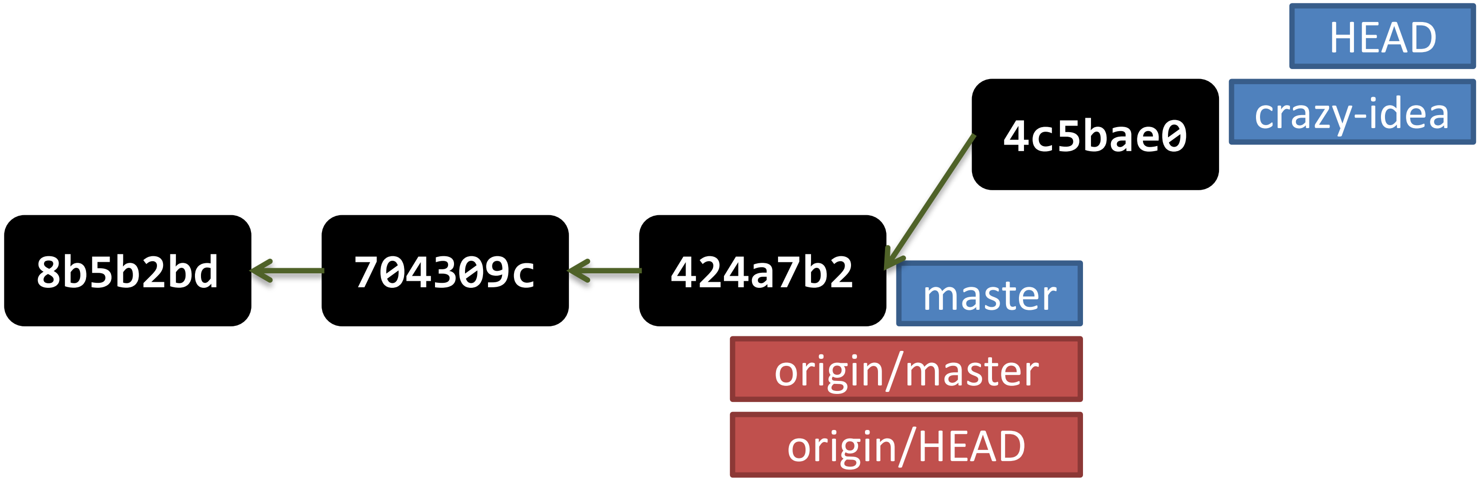
Working Directory

Staging Area

Local Repo

Remote Repo

```
$ git status
# On branch crazy-idea
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#        modified:   core/net/tcpip.c
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#        core/net/newfile.c
no changes added to commit (use "git add" and/or "git commit -a")
```



HEAD

crazy-idea

4c5bae0

8b5b2bd

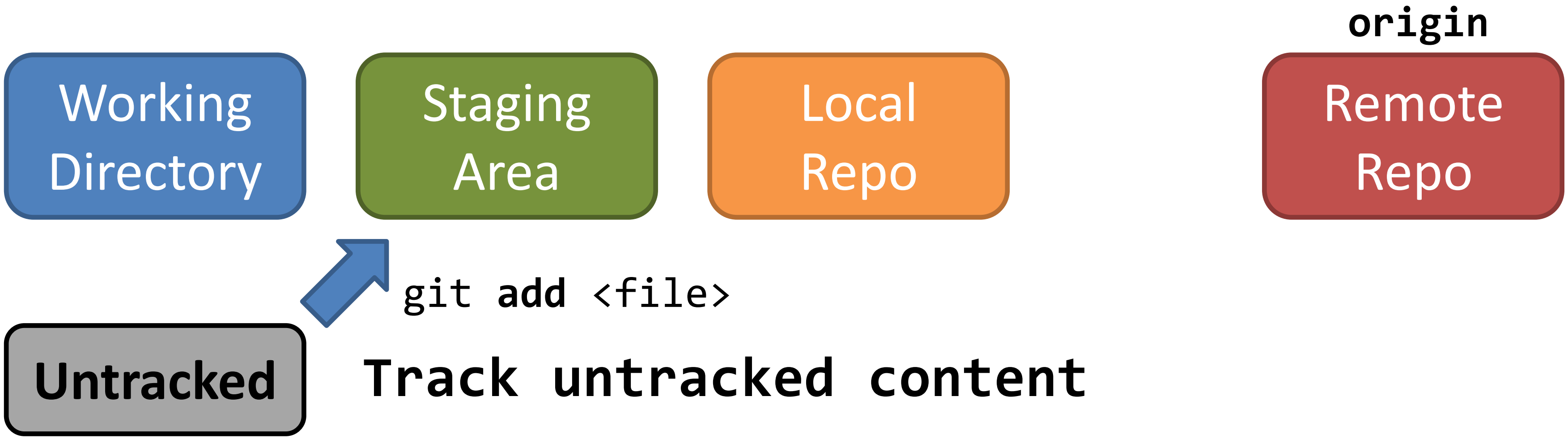
704309c

424a7b2

master

origin/master

origin/HEAD



origin

Working Directory

Staging Area

Local Repo

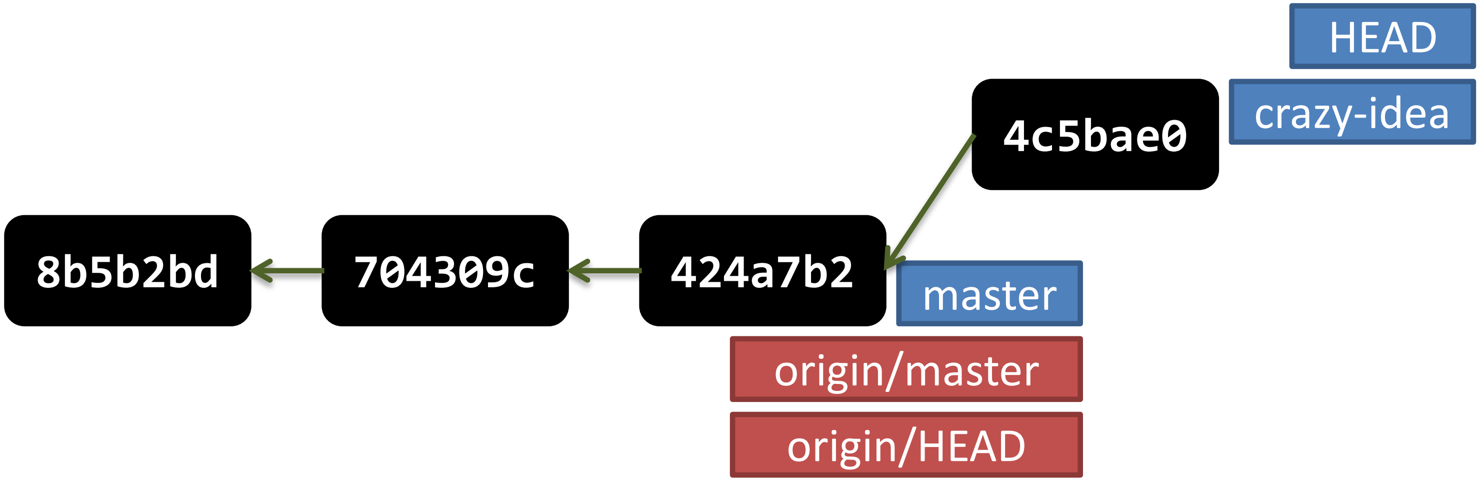
Remote Repo

git **add** <file>

Untracked

Track untracked content

```
$ git add core/net/newfile.c
```



HEAD

crazy-idea

4c5bae0

8b5b2bd

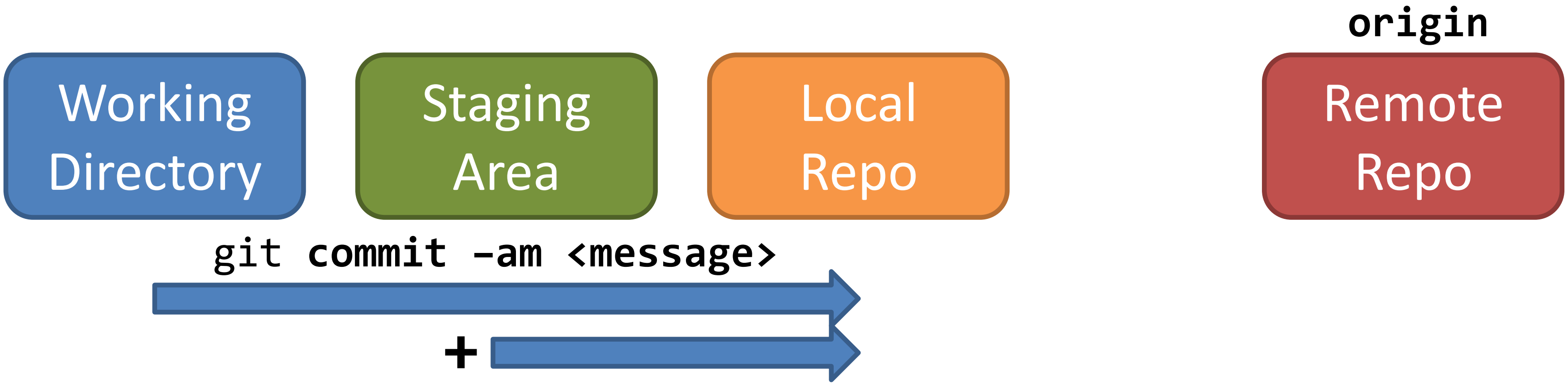
704309c

424a7b2

master

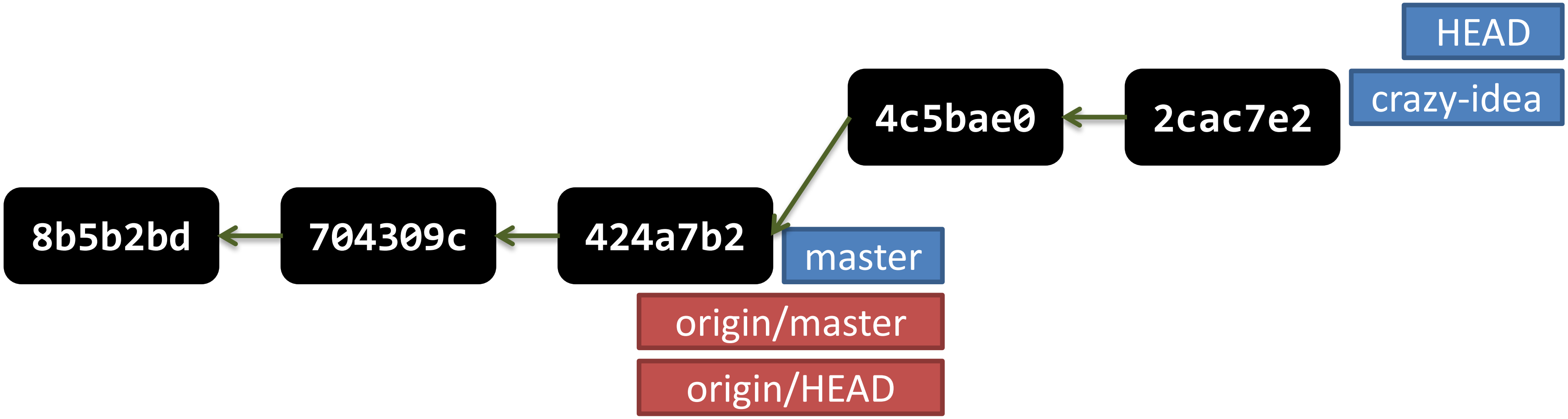
origin/master

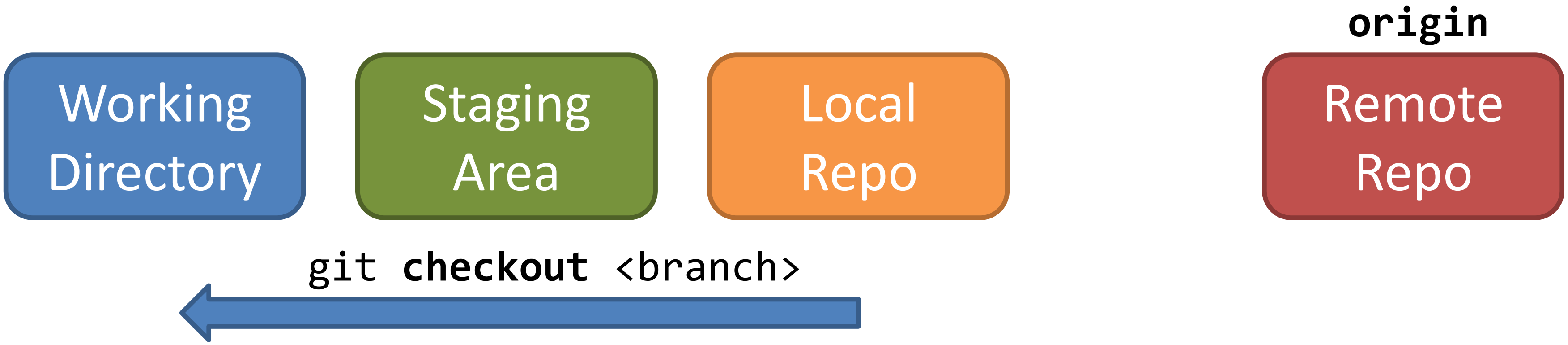
origin/HEAD



```
git commit -am <message>
```

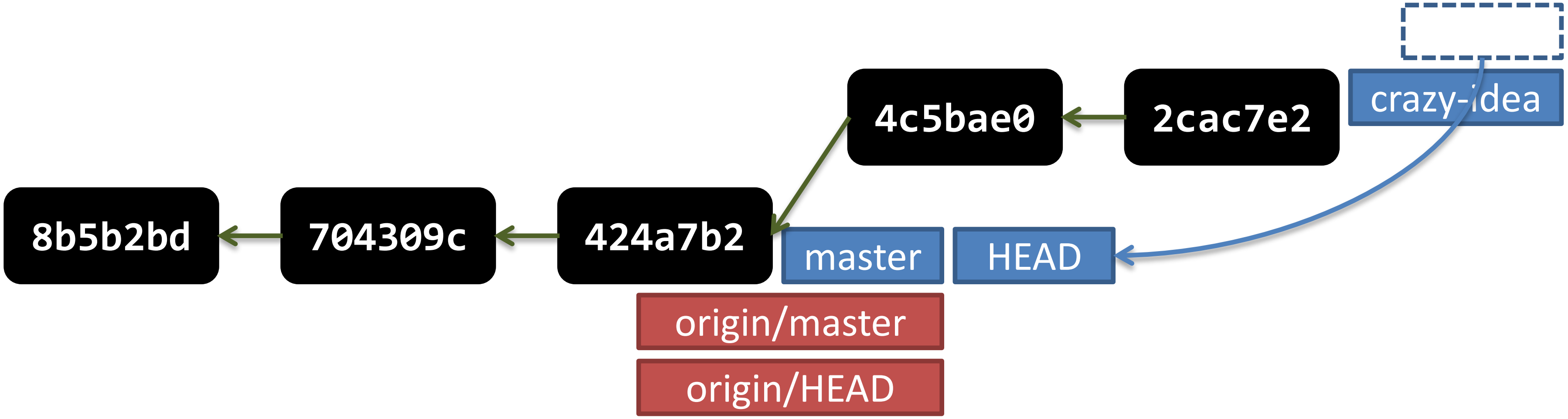
```
$ git add core/net/newfile.c
$ git commit -am "The rest of my crazy idea"

[crazy-idea 2cac7e2] The rest of my crazy idea
 2 files changed, 4 insertions(+)
 create mode 100644 core/net/newfile.c
```

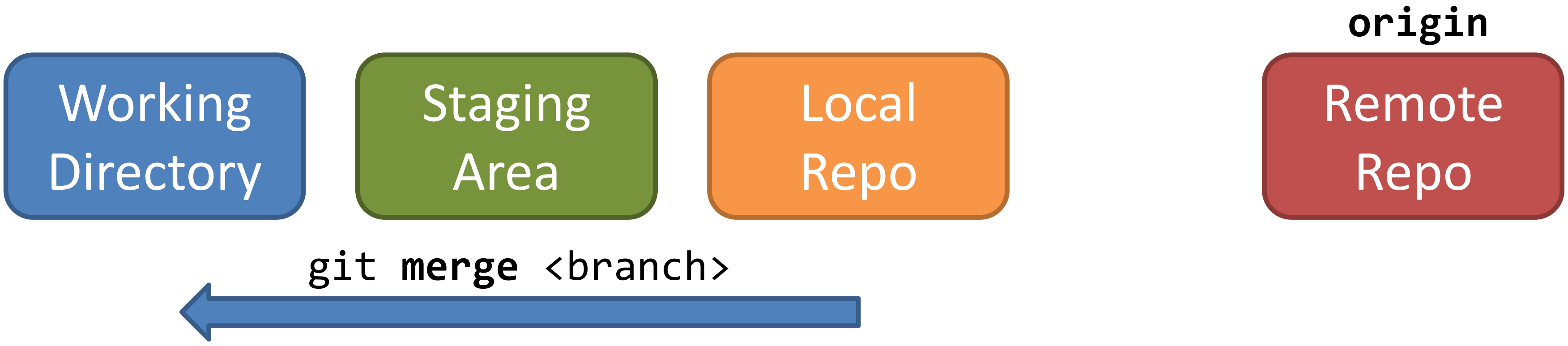




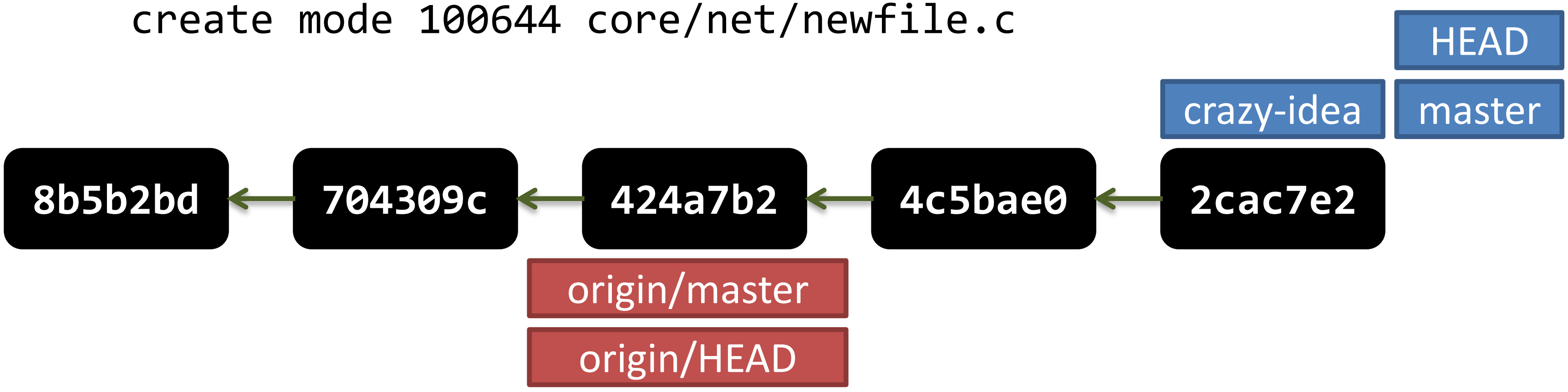
```
git checkout <branch>
```

```
$ git checkout master
```

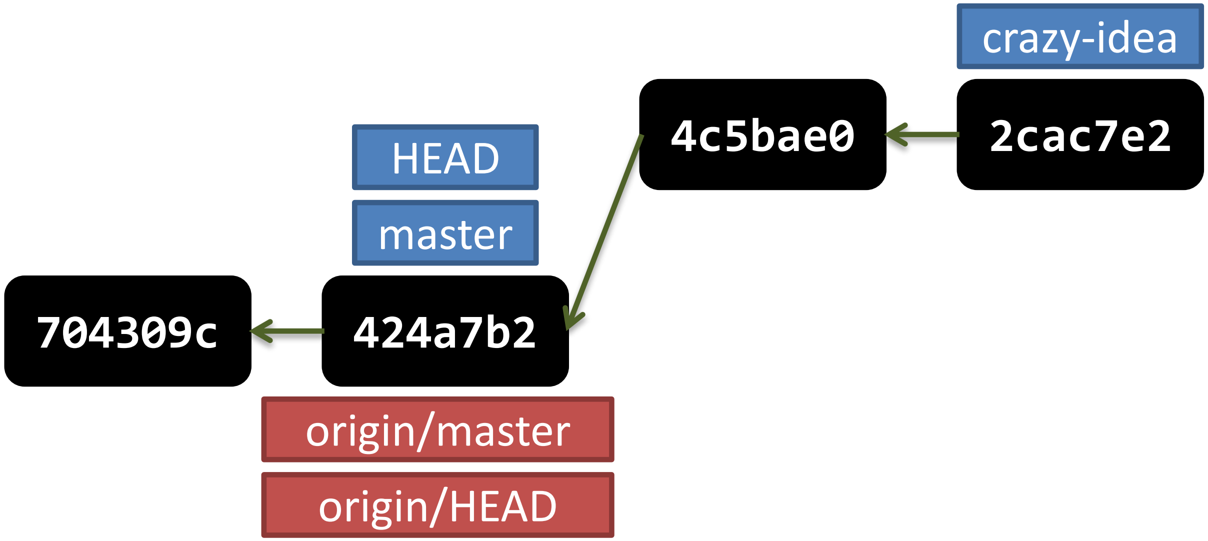




```
$ git checkout master

$ git merge crazy-idea

    Updating 424a7b2..2cac7e2
    Fast-forward
     core/net/newfile.c |    2 ++
     core/net/rpl/rpl.c |    2 ++
     core/net/tcpip.c   |    2 ++
     3 files changed, 6 insertions(+)
     create mode 100644 core/net/newfile.c
```

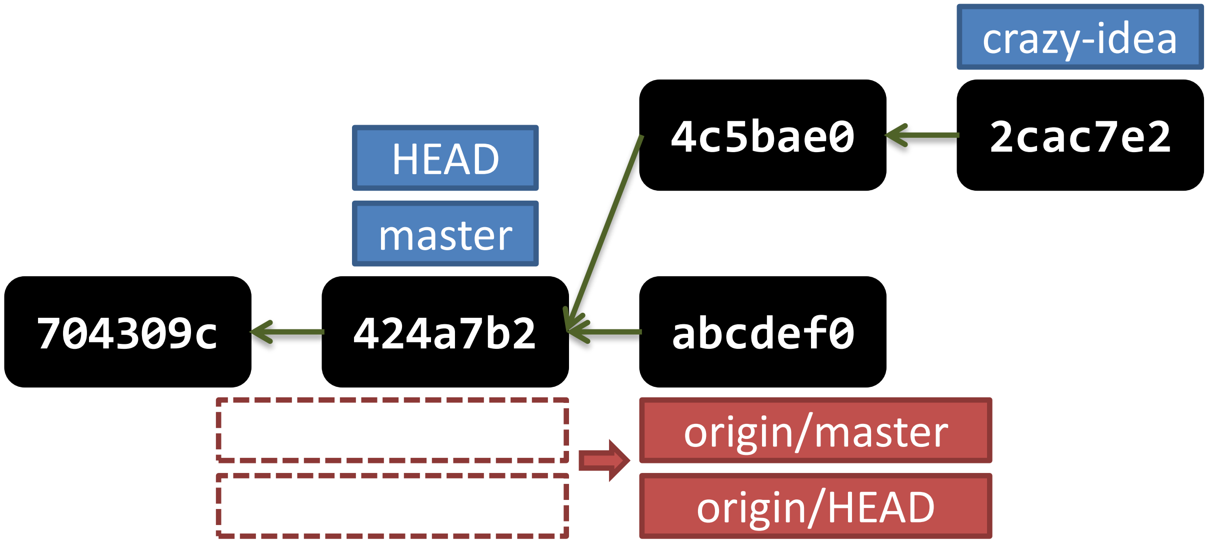


1 step back, with new changes on master before the merge



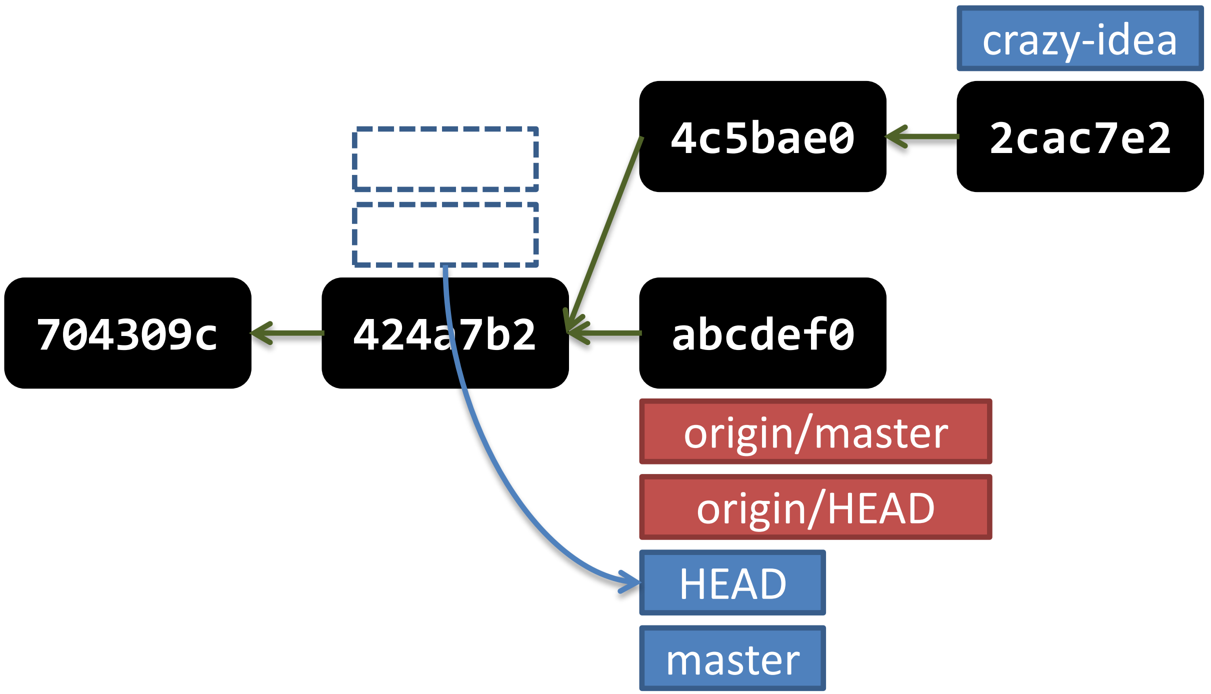
1 step back, with new changes on master before the merge (must be pulled in)

```
$ git checkout master
$ git fetch origin
 424a7b2..abcdef0 master       -> origin/master
```



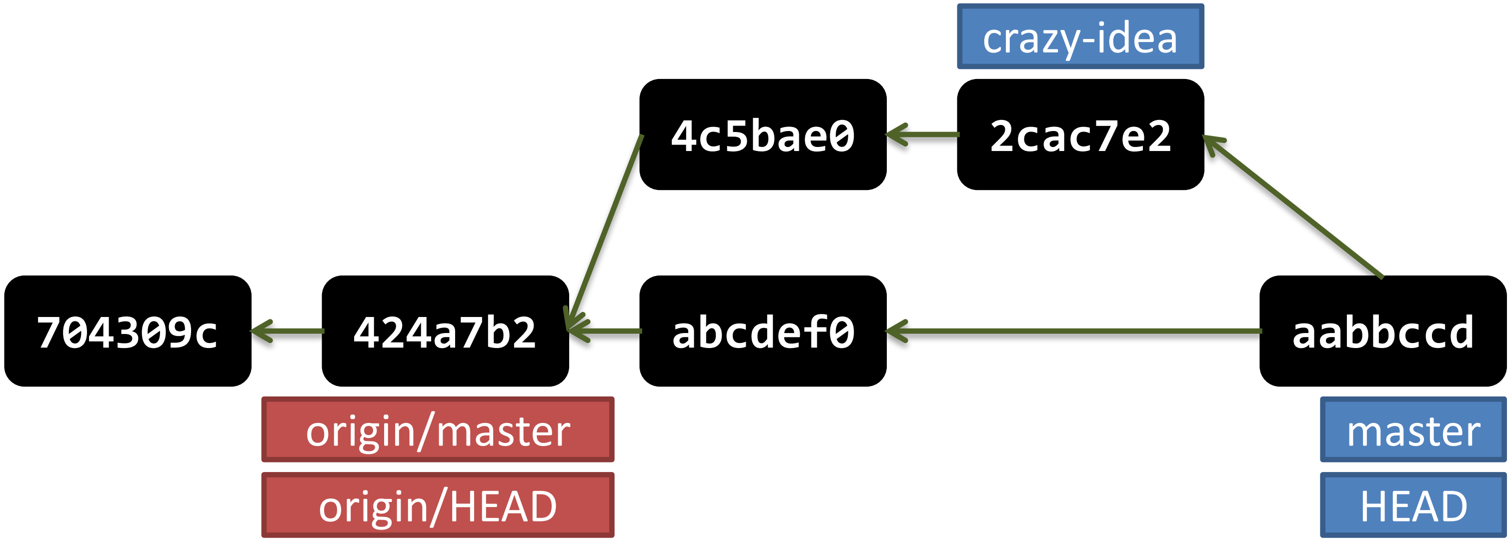
1 step back, with new changes on master before the merge

```
$ git checkout master
$ git fetch origin
 424a7b2..abcdef0 master       -> origin/master

$ git merge origin/master
```



1 step back, with new changes on master before the merge

```
Merge made by the 'recursive' strategy.
 core/net/newfile.c |    2 ++
 core/net/rpl/rpl.c |    2 ++
 core/net/tcpip.c   |    2 ++
 3 files changed, 6 insertions(+)
 create mode 100644 core/net/newfile.c
```

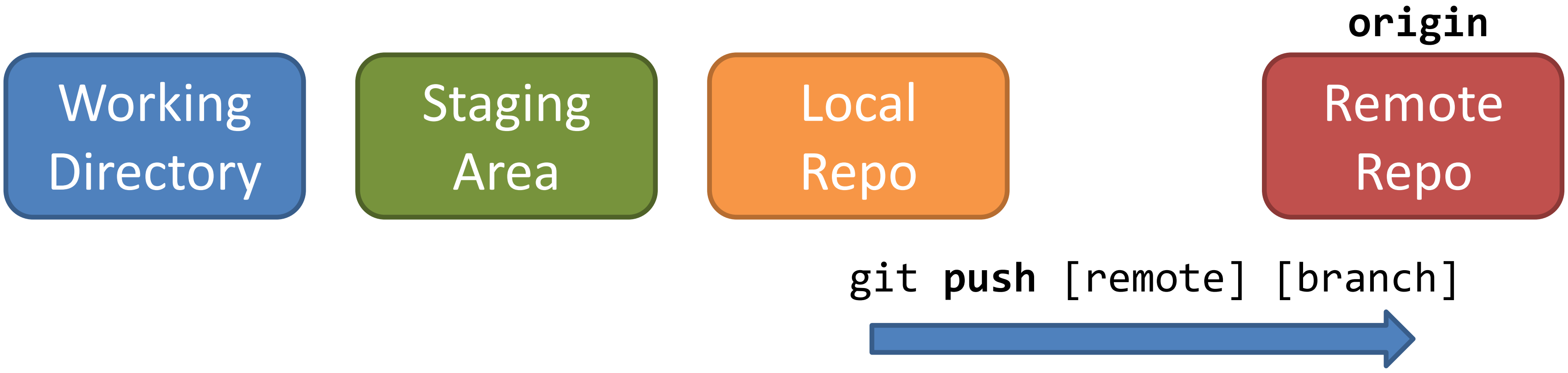


Merges are done LOCALLY!

```
“Git push failed, To prevent from
 losing history, non-fast forward
      updates were rejected”
```

→Not an error, but common-sense

→Files are rarely decorrelated



Working Directory

Staging Area

Local Repo

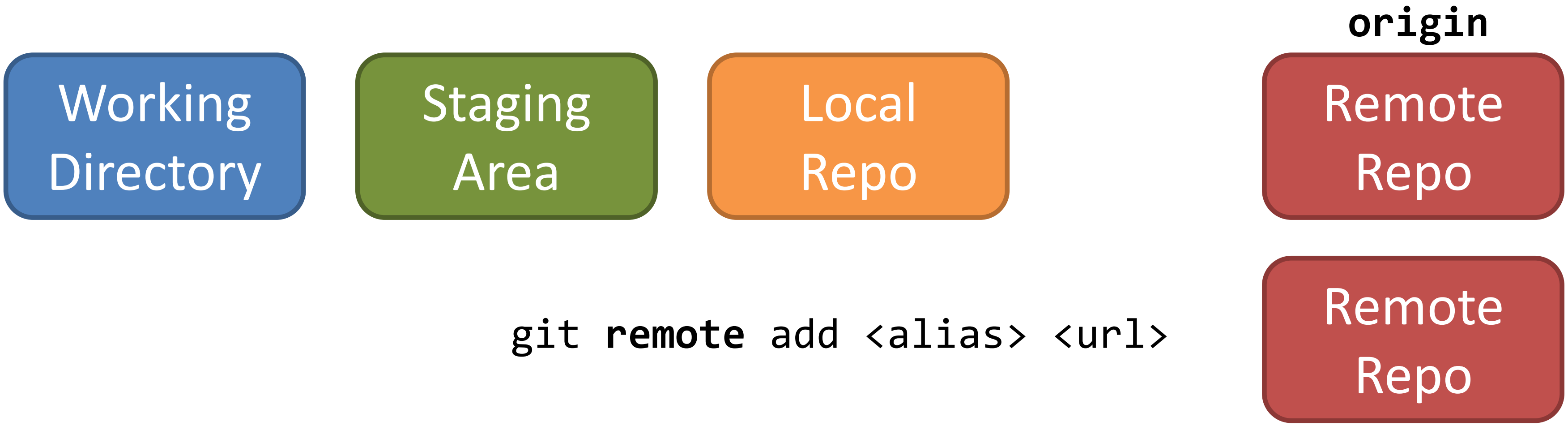
origin

Remote Repo

git **push** [remote] [branch]

```
$ git push origin master
```

Not permitted: How can Contiki authors trust you?



origin

Working Directory

Staging Area

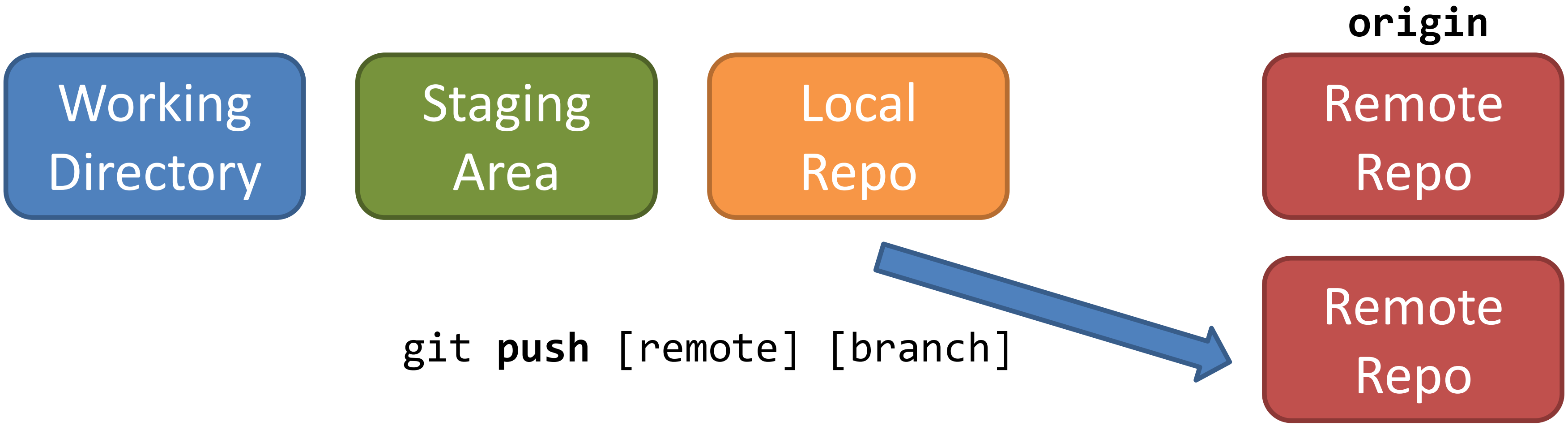
Local Repo

Remote Repo

Remote Repo

```
git remote add <alias> <url>
```

```
$ git remote add github-cetic git@github.com:cetic/contiki.git
```



origin

Working Directory

Staging Area

Local Repo

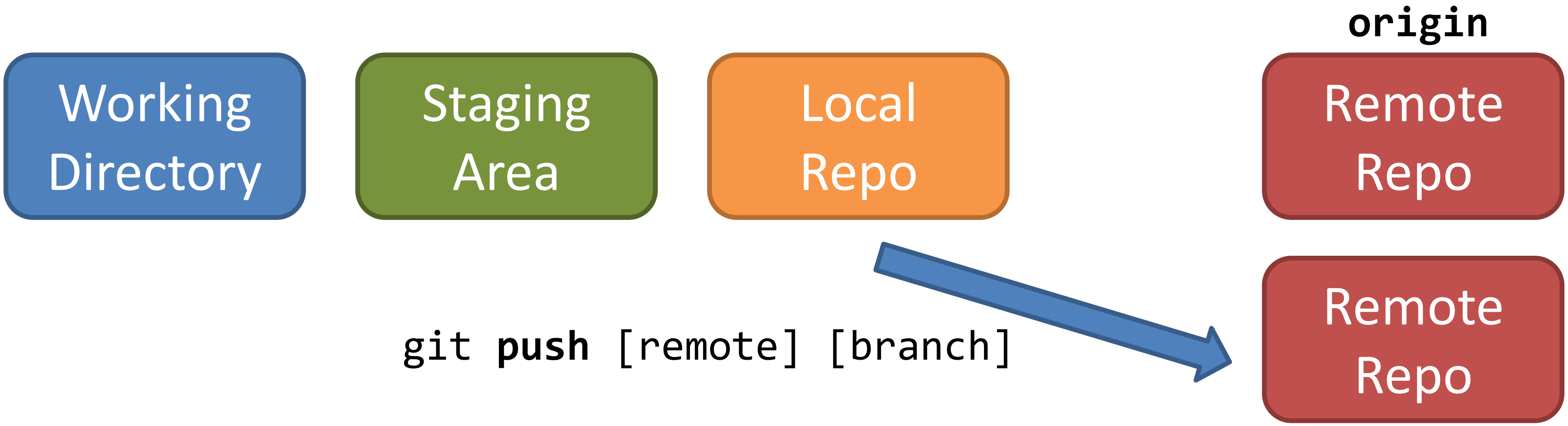
Remote Repo

Remote Repo

git **push** [remote] [branch]

```
$ git remote add github-cetic git@github.com:cetic/contiki.git

$ git push github-cetic master
```

OK



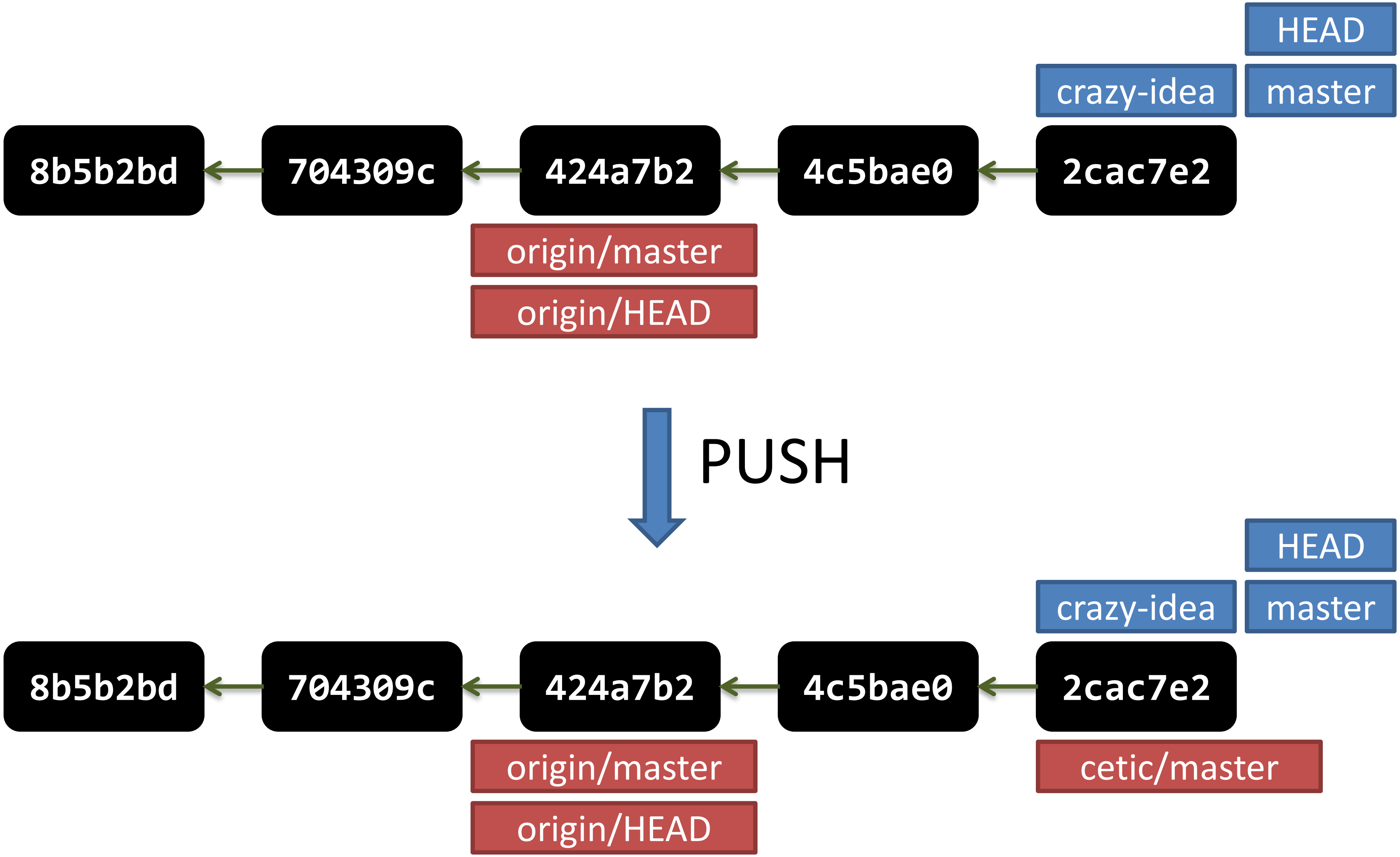
git **push** [remote] [branch]

```
$ git remote add github-cetic git@github.com:cetic/contiki.git
```

```
$ git push github-cetic master
```

OK

```
$ git push [remote] [branch[:alias]]
```



HEAD

crazy-idea

master

8b5b2bd

704309c

424a7b2

4c5bae0

2cac7e2

origin/master

origin/HEAD

PUSH

HEAD

crazy-idea

master

8b5b2bd

704309c

424a7b2

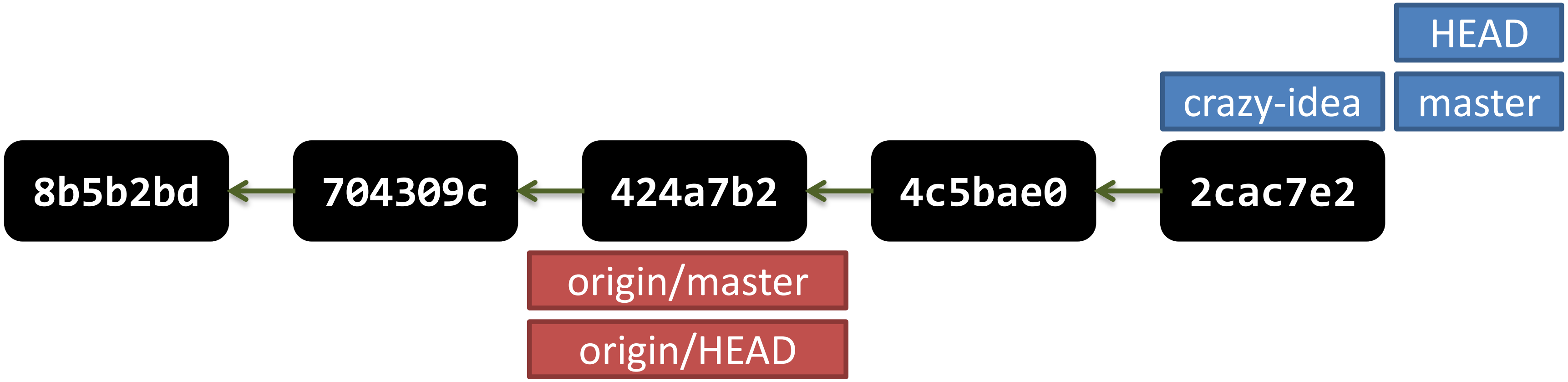
4c5bae0

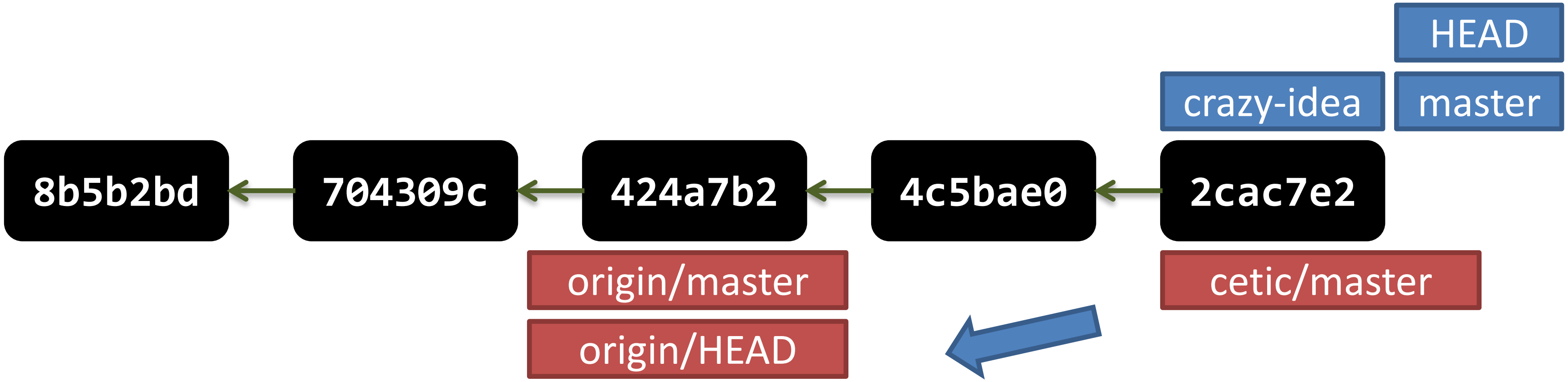
2cac7e2

origin/master

origin/HEAD

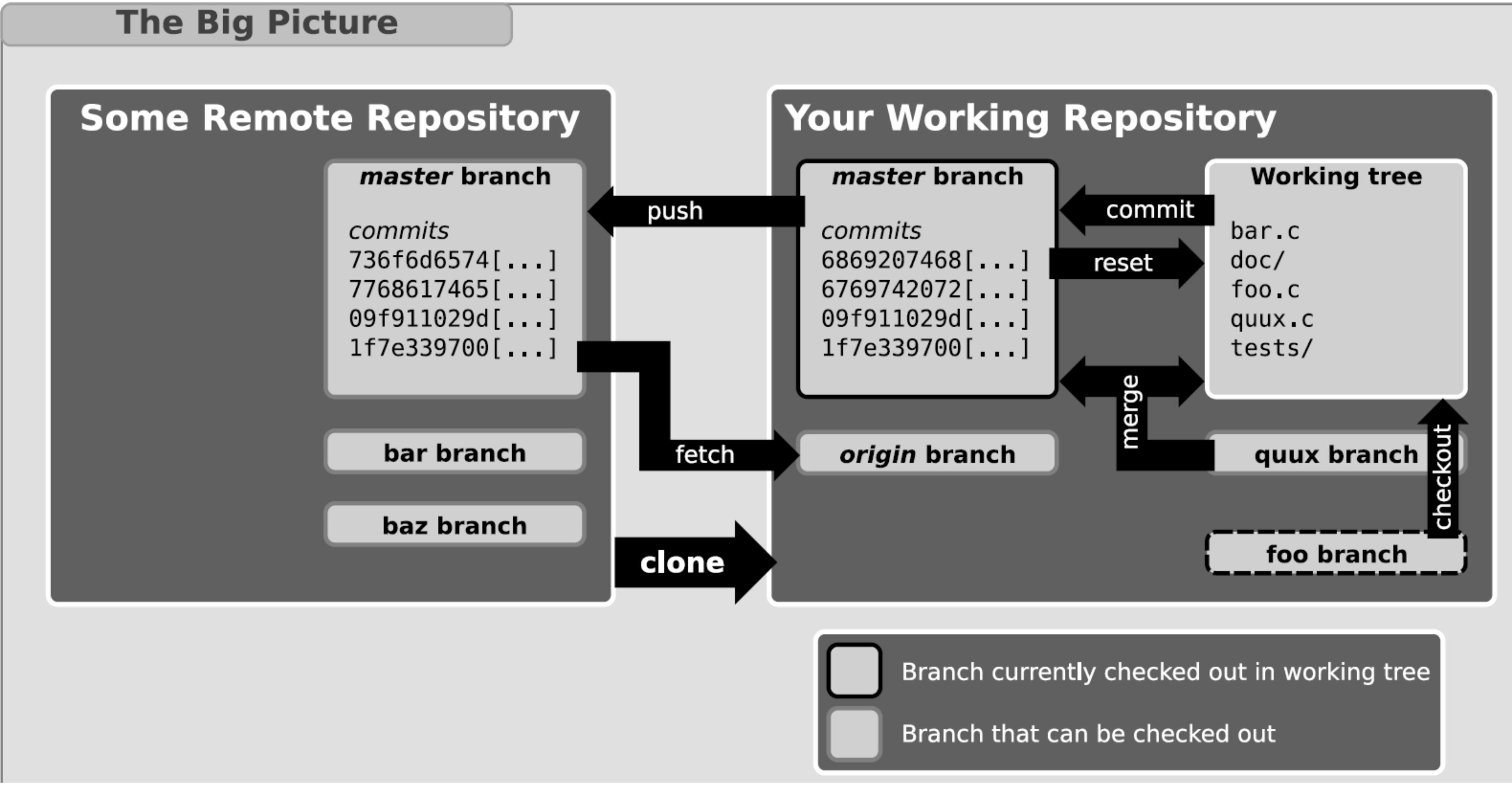
cetic/master





PULL

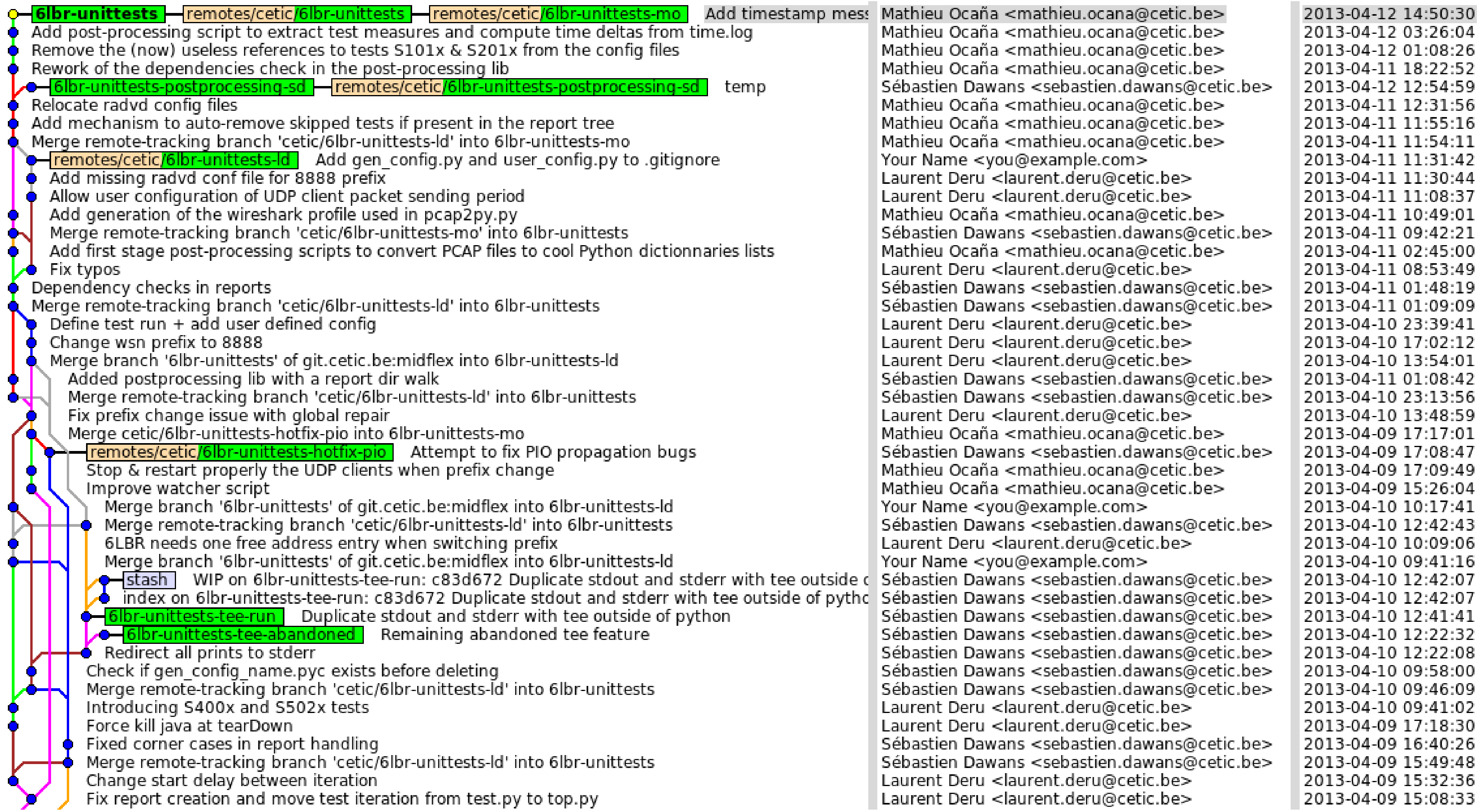
(done by owners of origin)



Jan Krüger's Cheat Sheet

http://jan-krueger.net/git

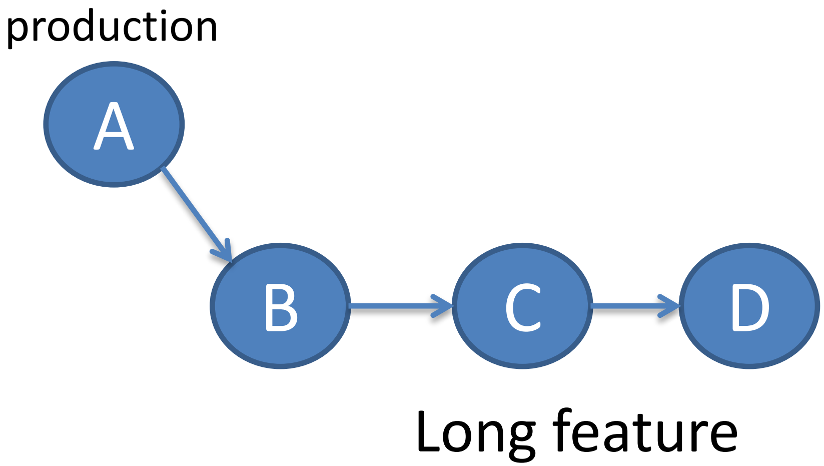
Git LOVES branches, and so should you



6lbr-unittests remotes/cetic/6lbr-unittests remotes/cetic/6lbr-unittests-mo Add timestamp mess	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-12 14:50:30
Add post-processing script to extract test measures and compute time deltas from time.log	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-12 03:26:04
Remove the (now) useless references to tests S101x & S201x from the config files	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-12 01:08:26
Rework of the dependencies check in the post-processing lib	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-11 18:22:52
6lbr-unittests-postprocessing-sd remotes/cetic/6lbr-unittests-postprocessing-sd temp	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-12 12:54:59
Relocate radvd config files	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-11 12:31:56
Add mechanism to auto-remove skipped tests if present in the report tree	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-11 11:55:16
Merge remote-tracking branch 'cetic/6lbr-unittests-ld' into 6lbr-unittests-mo	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-11 11:54:11
remotes/cetic/6lbr-unittests-ld Add gen_config.py and user_config.py to .gitignore	Your Name <you@example.com>	2013-04-11 11:31:42
Add missing radvd conf file for 8888 prefix	Laurent Deru <laurent.deru@cetic.be>	2013-04-11 11:30:44
Allow user configuration of UDP client packet sending period	Laurent Deru <laurent.deru@cetic.be>	2013-04-11 11:08:37
Add generation of the wireshark profile used in pcap2py.py	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-11 10:49:01
Merge remote-tracking branch 'cetic/6lbr-unittests-mo' into 6lbr-unittests	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-11 09:42:21
Add first stage post-processing scripts to convert PCAP files to cool Python dictionnaries lists	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-11 02:45:00
Fix typos	Laurent Deru <laurent.deru@cetic.be>	2013-04-11 08:53:49
Dependency checks in reports	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-11 01:48:19
Merge remote-tracking branch 'cetic/6lbr-unittests-ld' into 6lbr-unittests	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-11 01:09:09
Define test run + add user defined config	Laurent Deru <laurent.deru@cetic.be>	2013-04-10 23:39:41
Change wsn prefix to 8888	Laurent Deru <laurent.deru@cetic.be>	2013-04-10 17:02:12
Merge branch '6lbr-unittests' of git.cetic.be:midflex into 6lbr-unittests-ld	Laurent Deru <laurent.deru@cetic.be>	2013-04-10 13:54:01
Added postprocessing lib with a report dir walk	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-11 01:08:42
Merge remote-tracking branch 'cetic/6lbr-unittests-ld' into 6lbr-unittests	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-10 23:13:56
Fix prefix change issue with global repair	Laurent Deru <laurent.deru@cetic.be>	2013-04-10 13:48:59
Merge cetic/6lbr-unittests-hotfix-pio into 6lbr-unittests-mo	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-09 17:17:01
remotes/cetic/6lbr-unittests-hotfix-pio Attempt to fix PIO propagation bugs	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-09 17:08:47
Stop & restart properly the UDP clients when prefix change	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-09 17:09:49
Improve watcher script	Mathieu Ocaña <mathieu.ocana@cetic.be>	2013-04-09 15:26:04
Merge branch '6lbr-unittests' of git.cetic.be:midflex into 6lbr-unittests-ld	Your Name <you@example.com>	2013-04-10 10:17:41
Merge remote-tracking branch 'cetic/6lbr-unittests-ld' into 6lbr-unittests	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-10 12:42:43
6LBR needs one free address entry when switching prefix	Laurent Deru <laurent.deru@cetic.be>	2013-04-10 10:09:06
Merge branch '6lbr-unittests' of git.cetic.be:midflex into 6lbr-unittests-ld	Your Name <you@example.com>	2013-04-10 09:41:16
stash WIP on 6lbr-unittests-tee-run: c83d672 Duplicate stdout and stderr with tee outside c	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-10 12:42:07
index on 6lbr-unittests-tee-run: c83d672 Duplicate stdout and stderr with tee outside of pytho	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-10 12:42:07
6lbr-unittests-tee-run Duplicate stdout and stderr with tee outside of python	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-10 12:41:41
6lbr-unittests-tee-abandoned Remaining abandoned tee feature	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-10 12:22:32
Redirect all prints to stderr	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-10 12:22:08
Check if gen_config_name.pyc exists before deleting	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-10 09:58:00
Merge remote-tracking branch 'cetic/6lbr-unittests-ld' into 6lbr-unittests	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-10 09:46:09
Introducing S400x and S502x tests	Laurent Deru <laurent.deru@cetic.be>	2013-04-10 09:41:02
Force kill java at tearDown	Laurent Deru <laurent.deru@cetic.be>	2013-04-09 17:18:30
Fixed corner cases in report handling	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-09 16:40:26
Merge remote-tracking branch 'cetic/6lbr-unittests-ld' into 6lbr-unittests	Sébastien Dawans <sebastien.dawans@cetic.be>	2013-04-09 15:49:48
Change start delay between iteration	Laurent Deru <laurent.deru@cetic.be>	2013-04-09 15:32:36
Fix report creation and move test iteration from test.py to top.py	Laurent Deru <laurent.deru@cetic.be>	2013-04-09 15:08:33

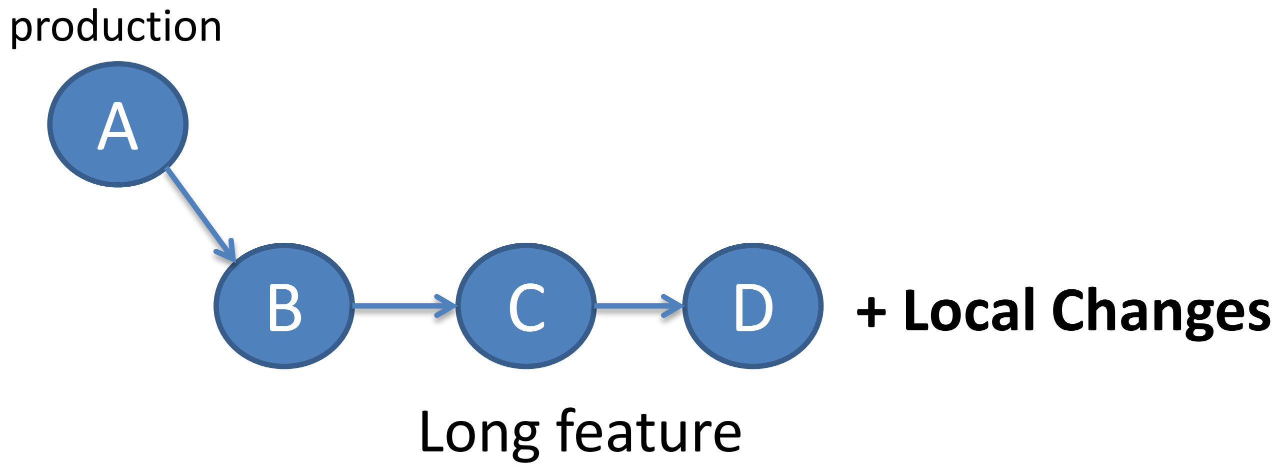
Checkouts are cheap and fast

Typical Workday



Checkouts are cheap and fast

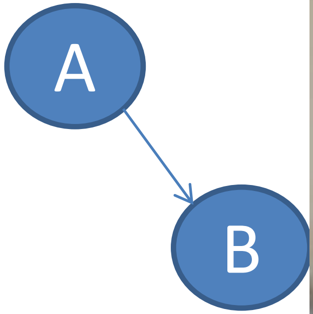
Typical Workday

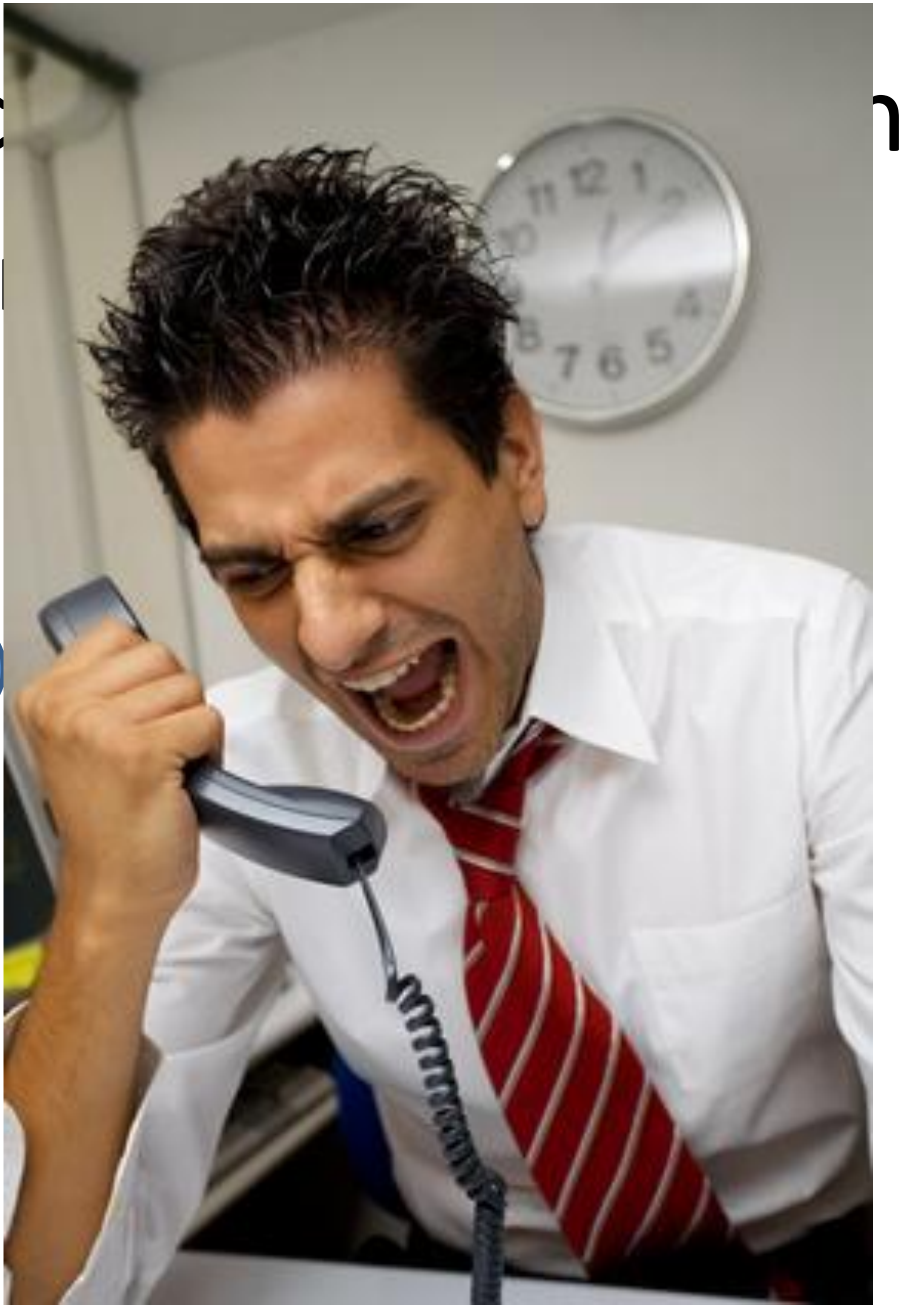


Chec nd fast

Typical Wor

production

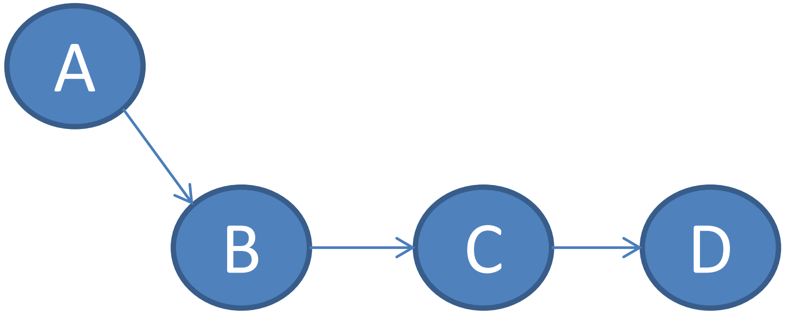




Checkouts are cheap and fast

Typical Workday

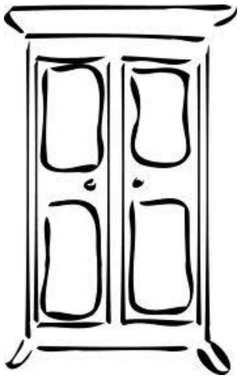
production



Long feature

Don’t Panic!!

STASH!

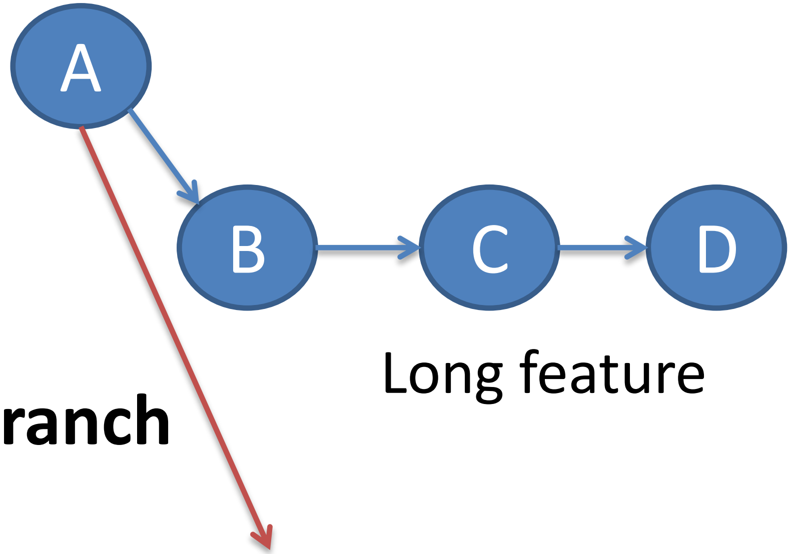


Local Changes

Checkouts are cheap and fast

Typical Workday

production



A

B

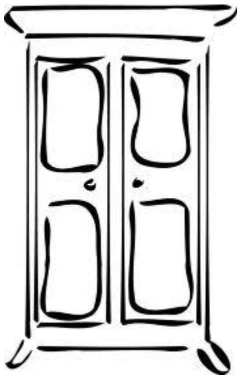
C

D

Long feature

branch

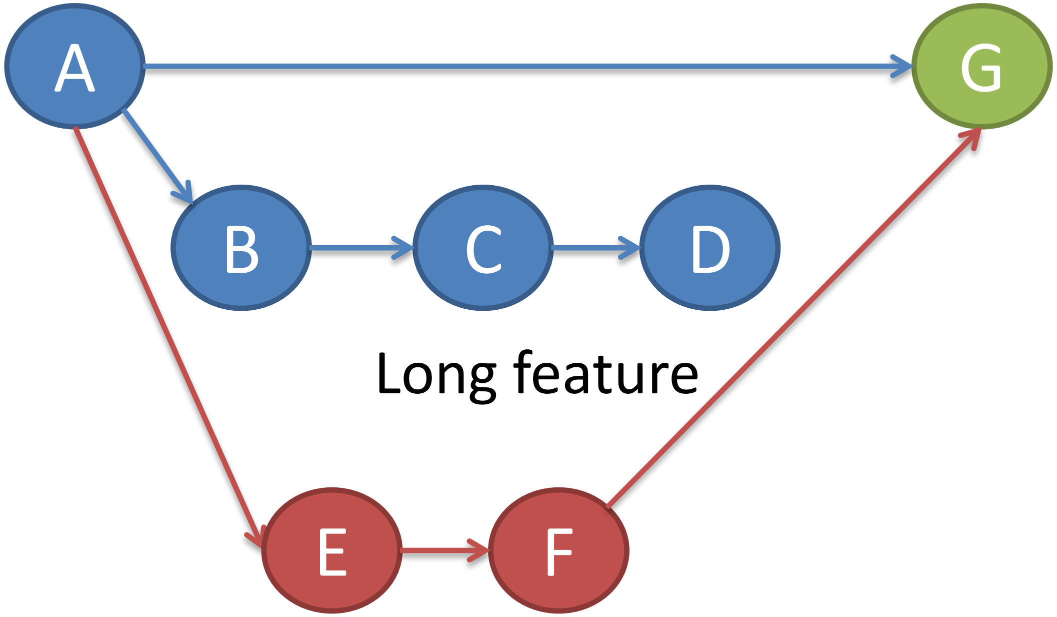
HOTFIX



Checkouts are cheap and fast

Typical Workday

production



A

G

B

C

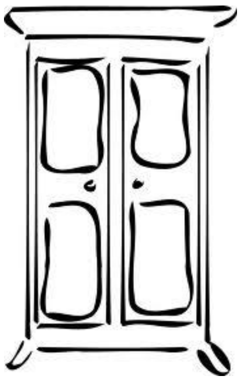
D

Long feature

E

F

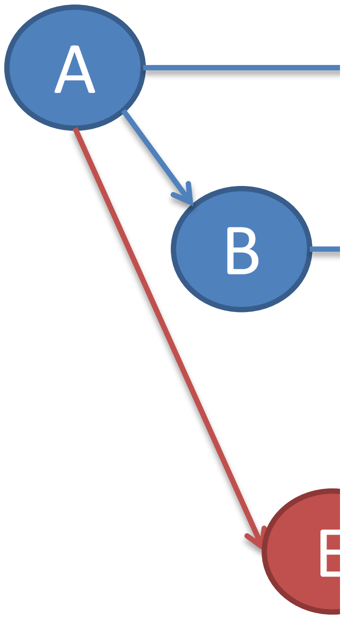
HOTFIX

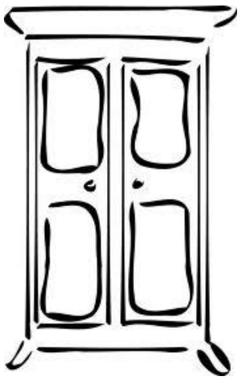


Check nd fast

Typical Work

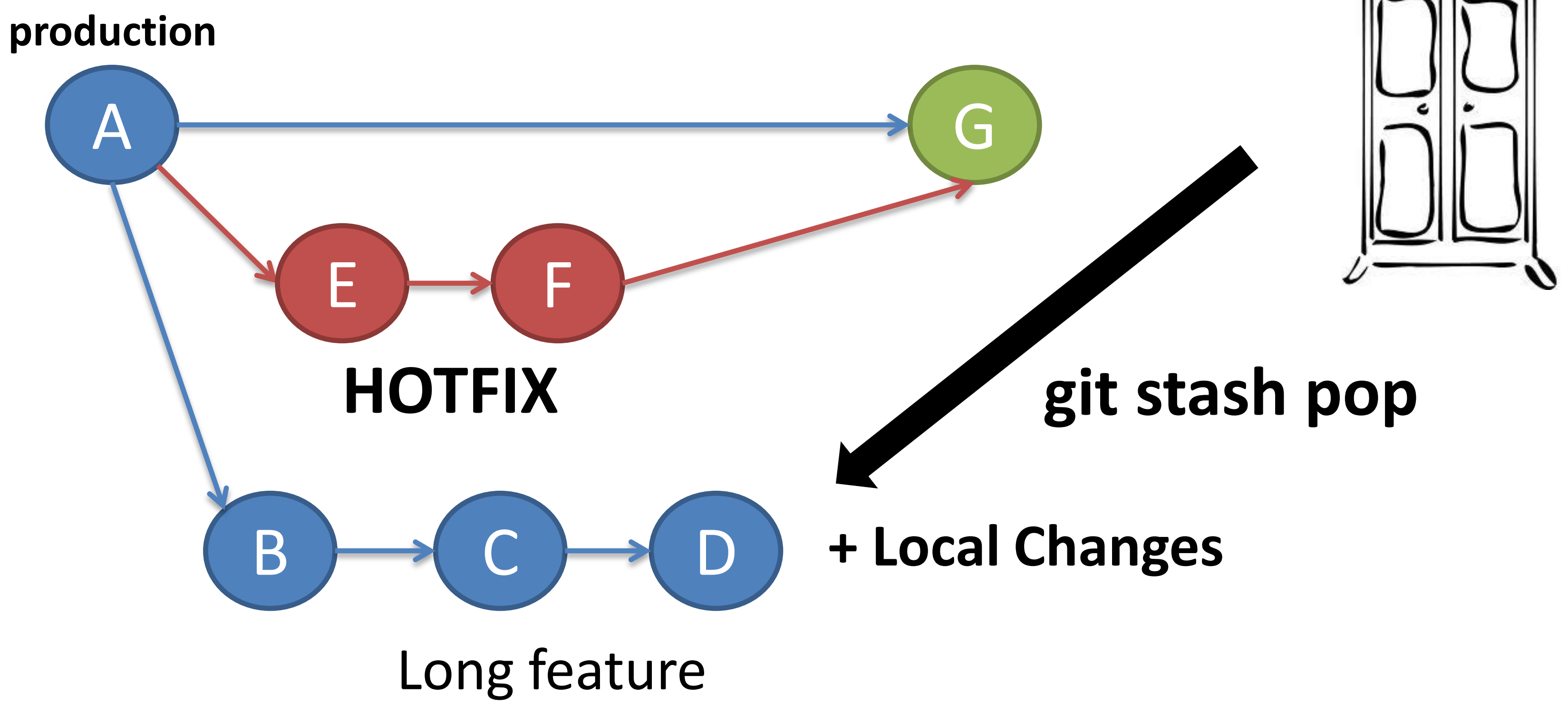
production





Checkouts are cheap and fast

Typical Workday



production

HOTFIX

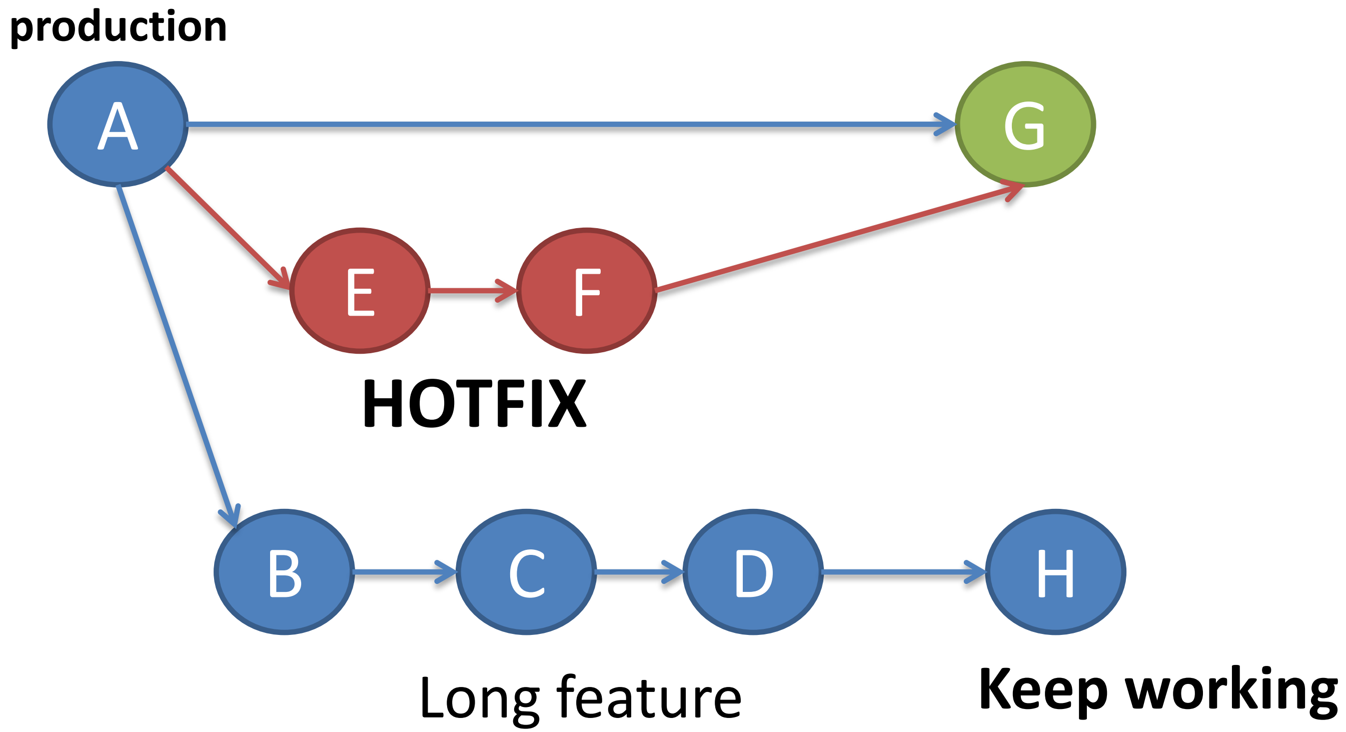
git stash pop

+ Local Changes

Long feature

Checkouts are cheap and fast

Typical Workday



production

A

G

E

F

HOTFIX

B

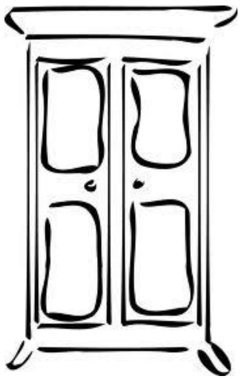
C

D

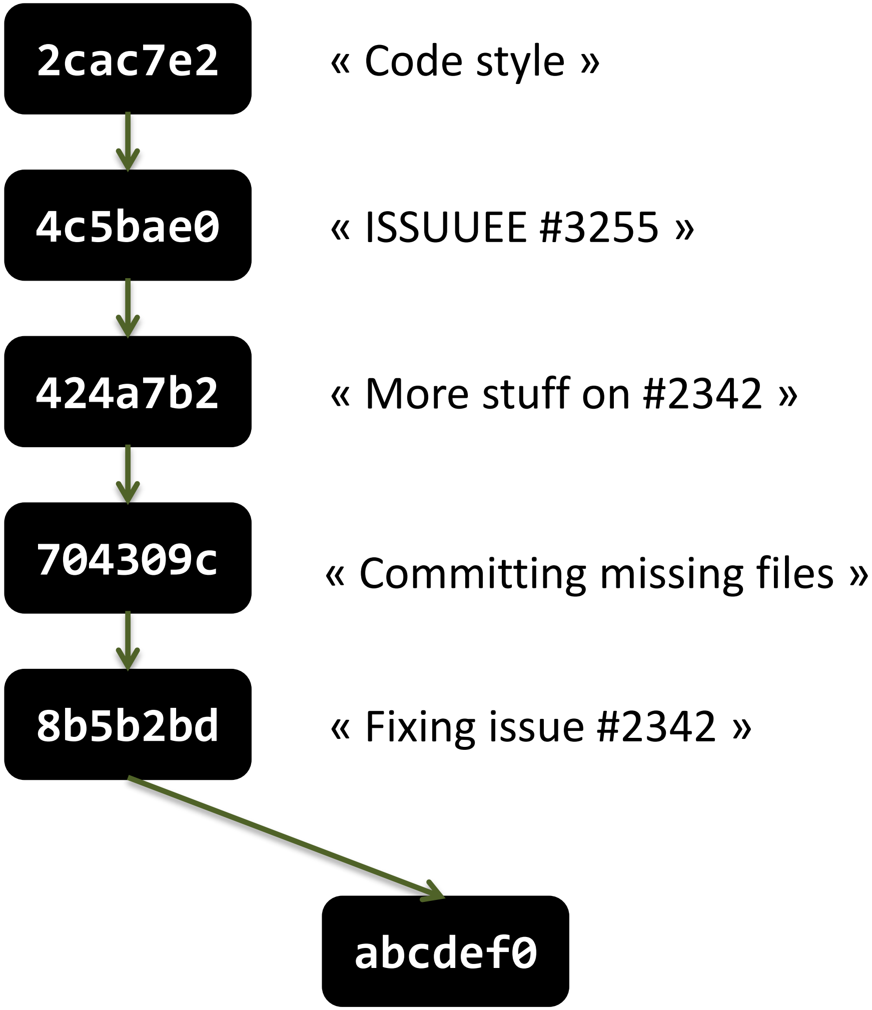
H

Long feature

Keep working



Re-Writing History



2cac7e2 « Code style »

4c5bae0 « ISSUUEE #3255 »

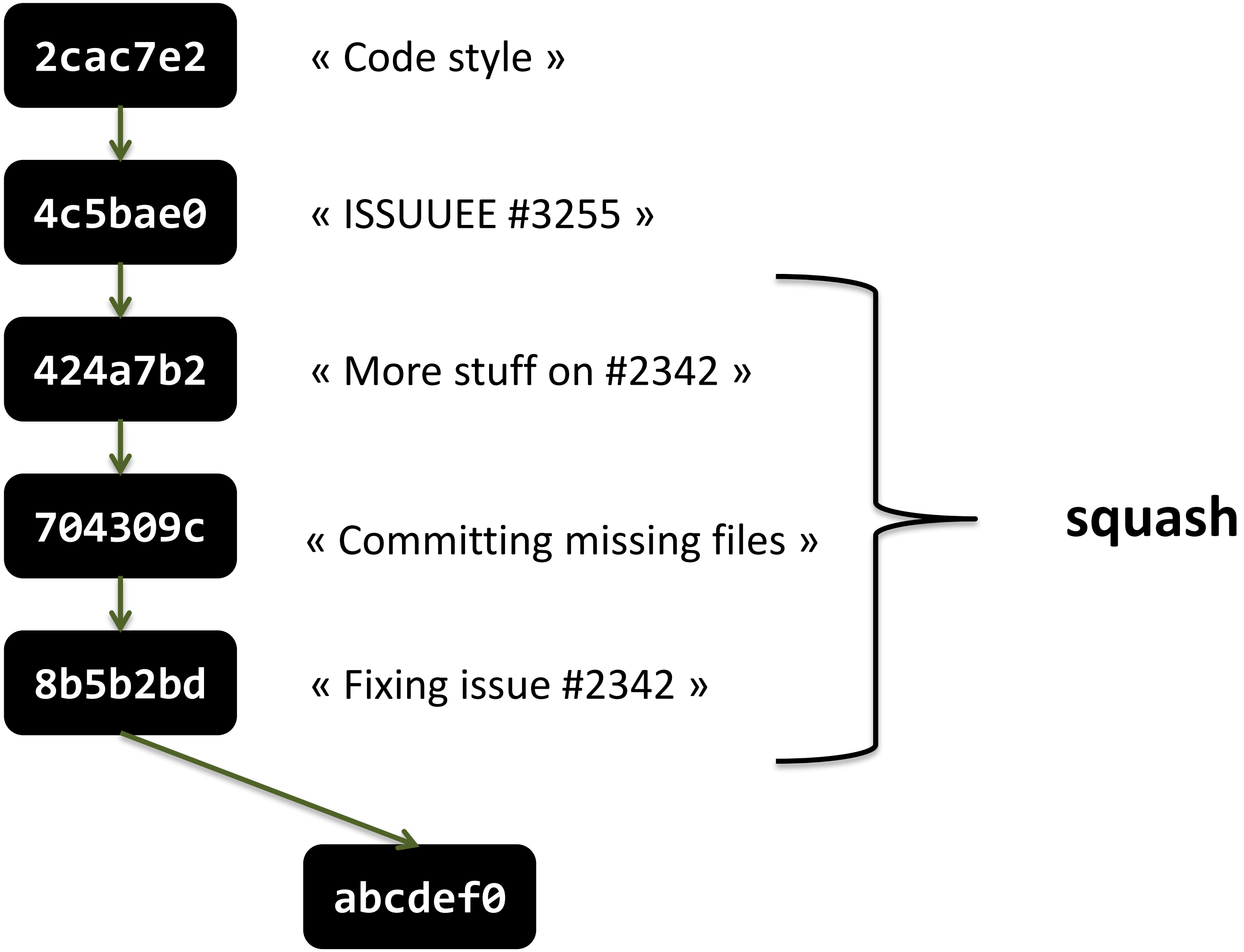
424a7b2 « More stuff on #2342 »

704309c « Committing missing files »

8b5b2bd « Fixing issue #2342 »

abcdef0

Squashing commits



2cac7e2 « Code style »

4c5bae0 « ISSUUEE #3255 »

424a7b2 « More stuff on #2342 »

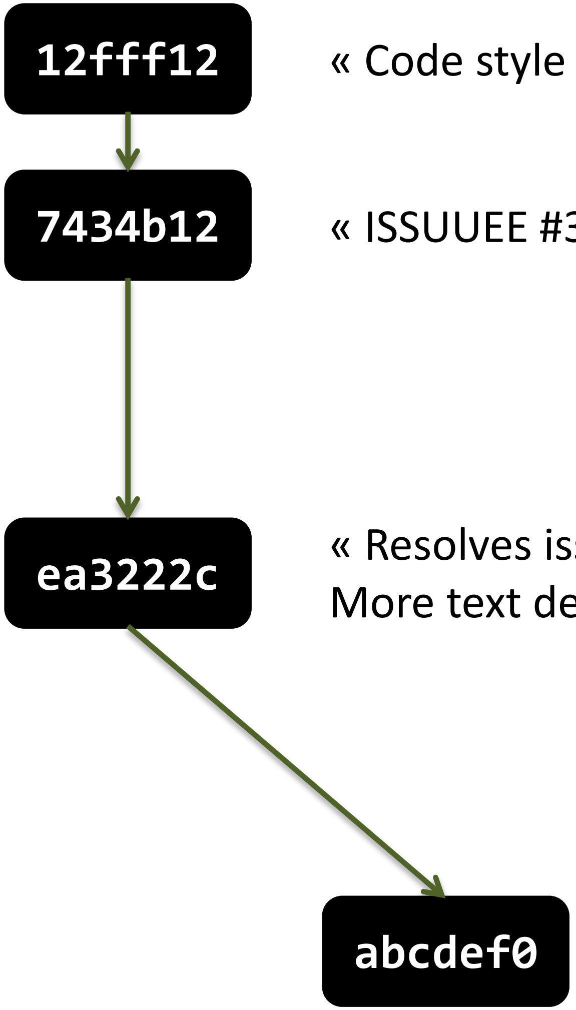
704309c « Committing missing files »

8b5b2bd « Fixing issue #2342 »

abcdef0

squash

Squashing commits



12fff12

« Code style »

7434b12

« ISSUUEE #3255 »

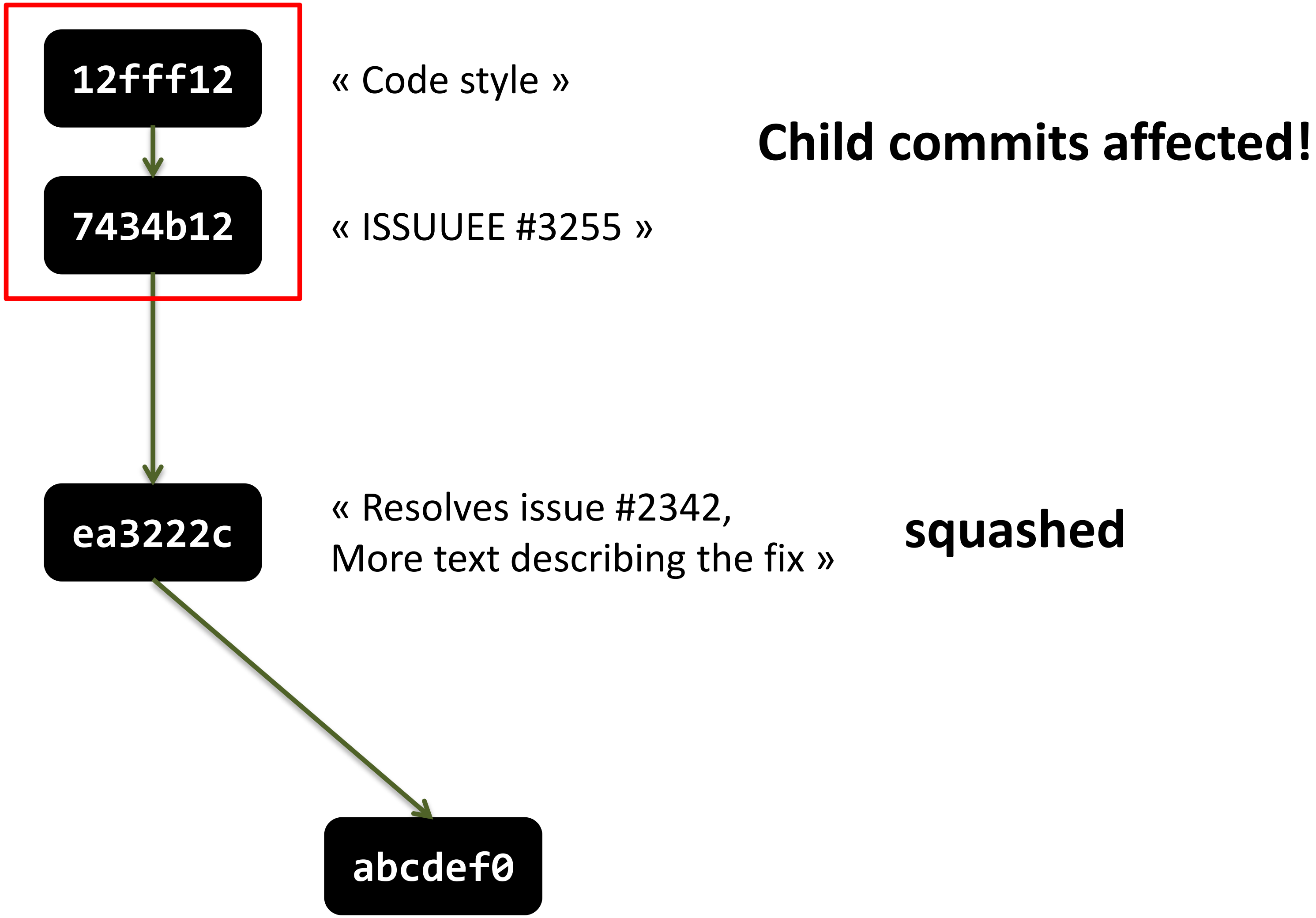
ea3222c

« Resolves issue #2342,
More text describing the fix »

squashed

abcdef0

Squashing commits



« Code style »

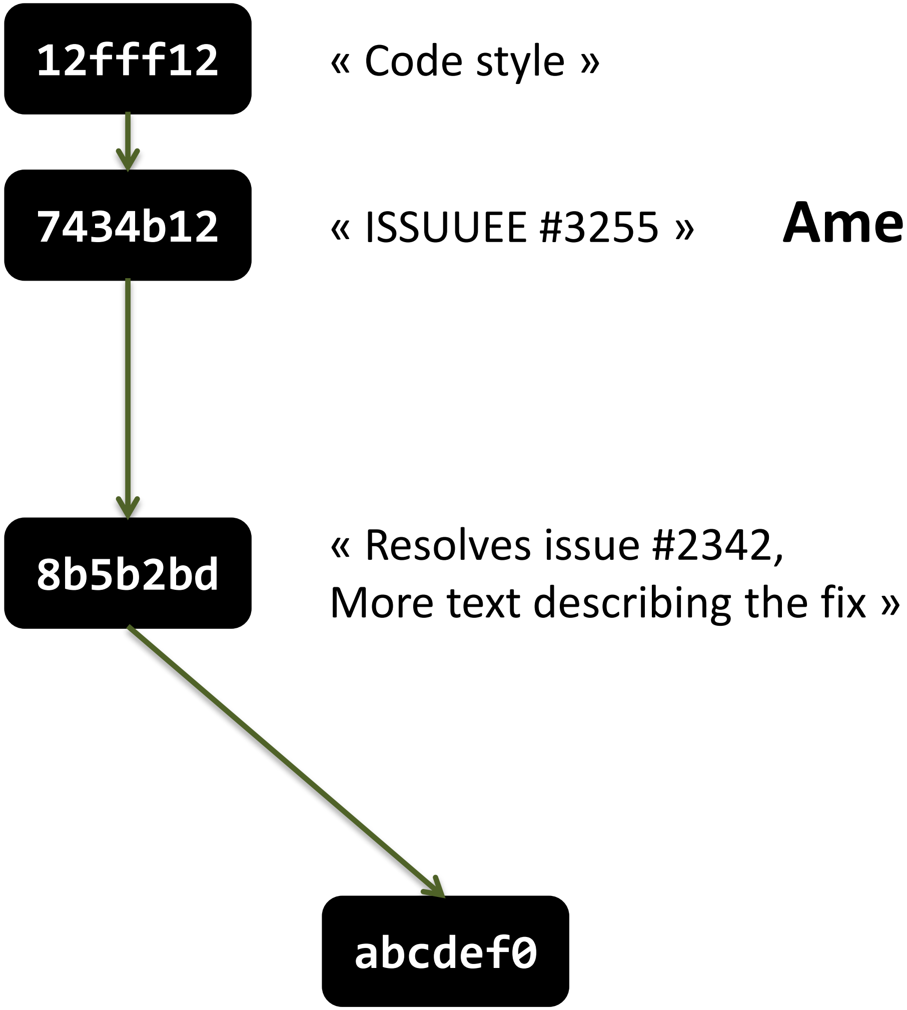
« ISSUUEE #3255 »

Child commits affected!

« Resolves issue #2342,
More text describing the fix »

squashed

Squashing commits



12fff12

« Code style »

7434b12

« ISSUUEE #3255 »

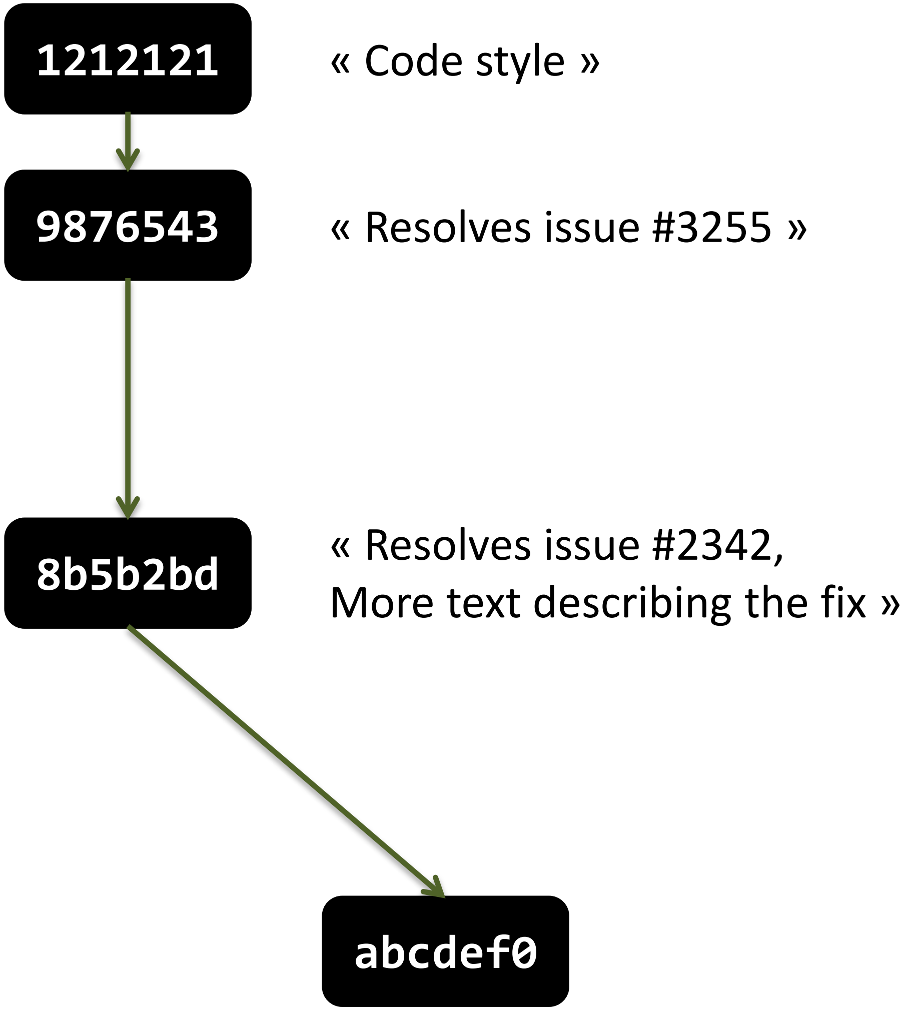
Amend the commit message

8b5b2bd

« Resolves issue #2342,
More text describing the fix »

abcdef0

Squashing commits



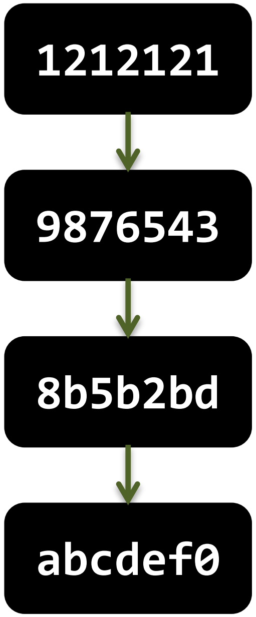
1212121 « Code style »

9876543 « Resolves issue #3255 »

8b5b2bd « Resolves issue #2342,
More text describing the fix »

abcdef0

**Clean History,
can be merged**



1212121 « Code style »

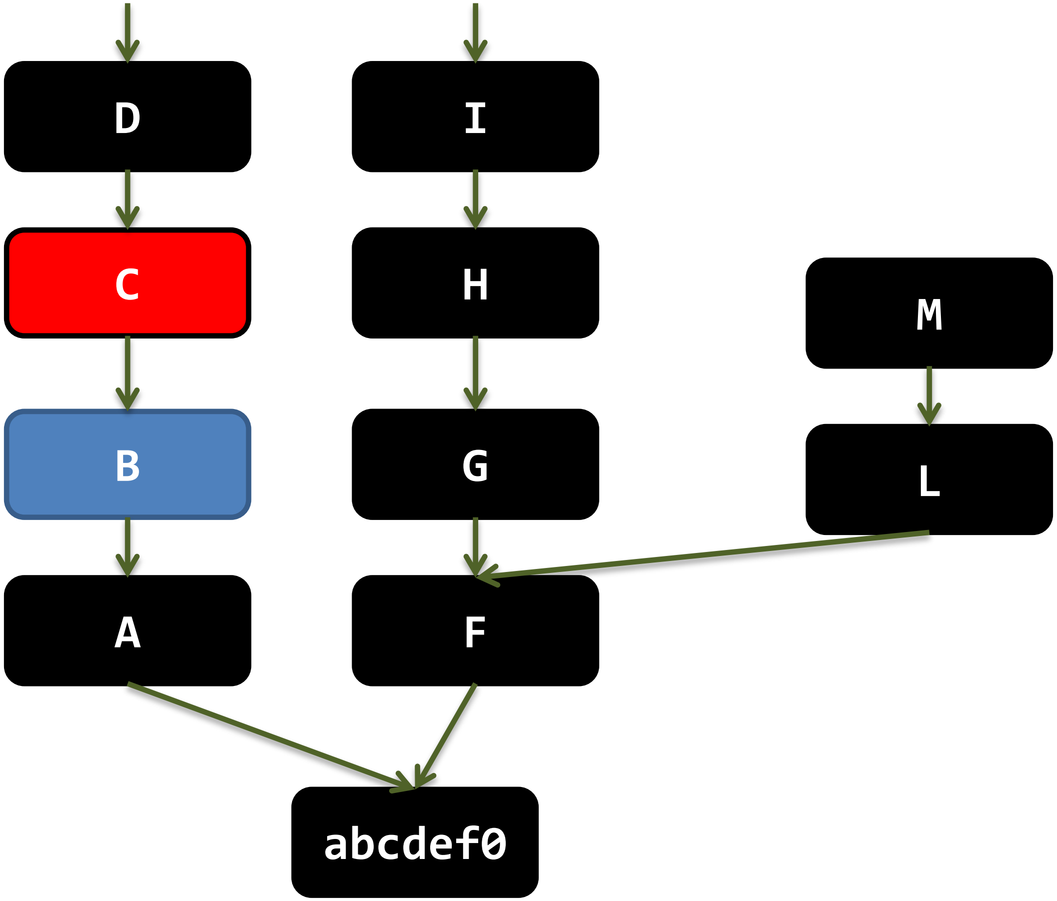
9876543 « Resolves issue #3255 »

8b5b2bd « Resolves issue #2342,
More text describing the fix »

abcdef0

Cherry-Picking

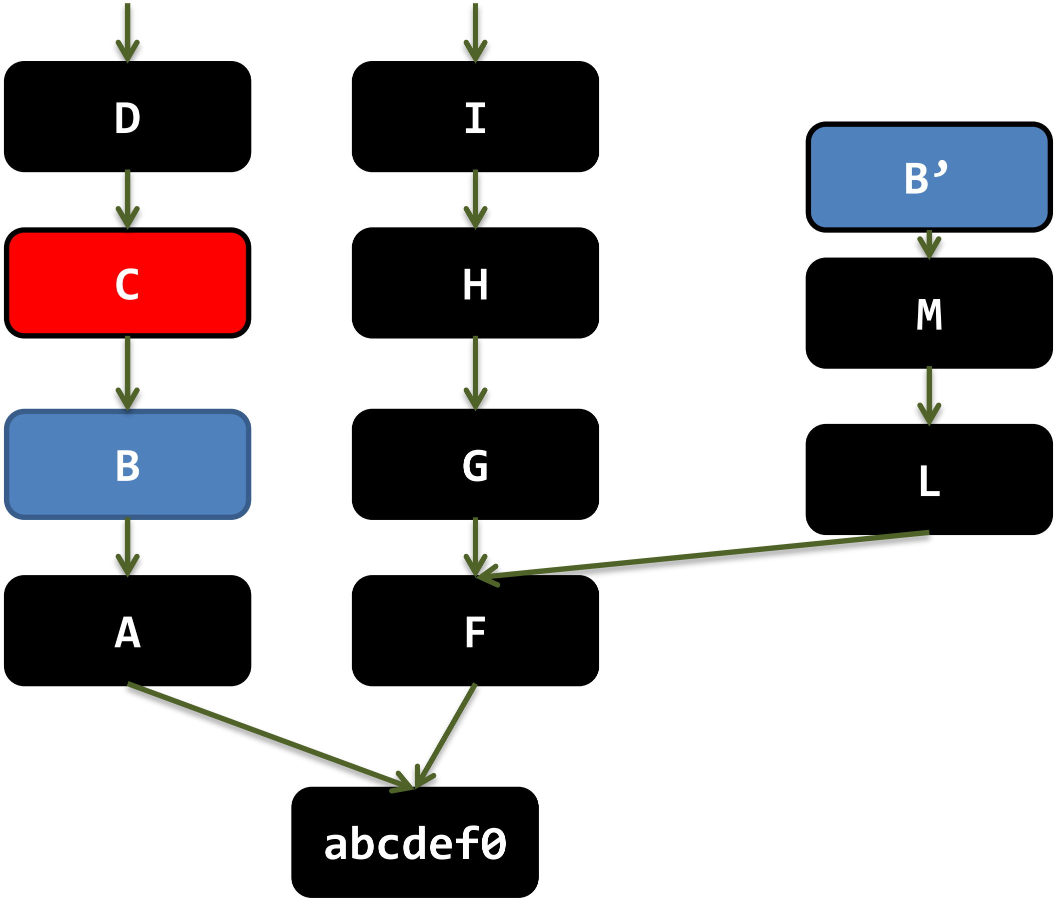
“Given one or more existing commits, apply the change each one introduces, recording a new commit for each.”



```
git cherry-pick B C
```

Cherry-Picking

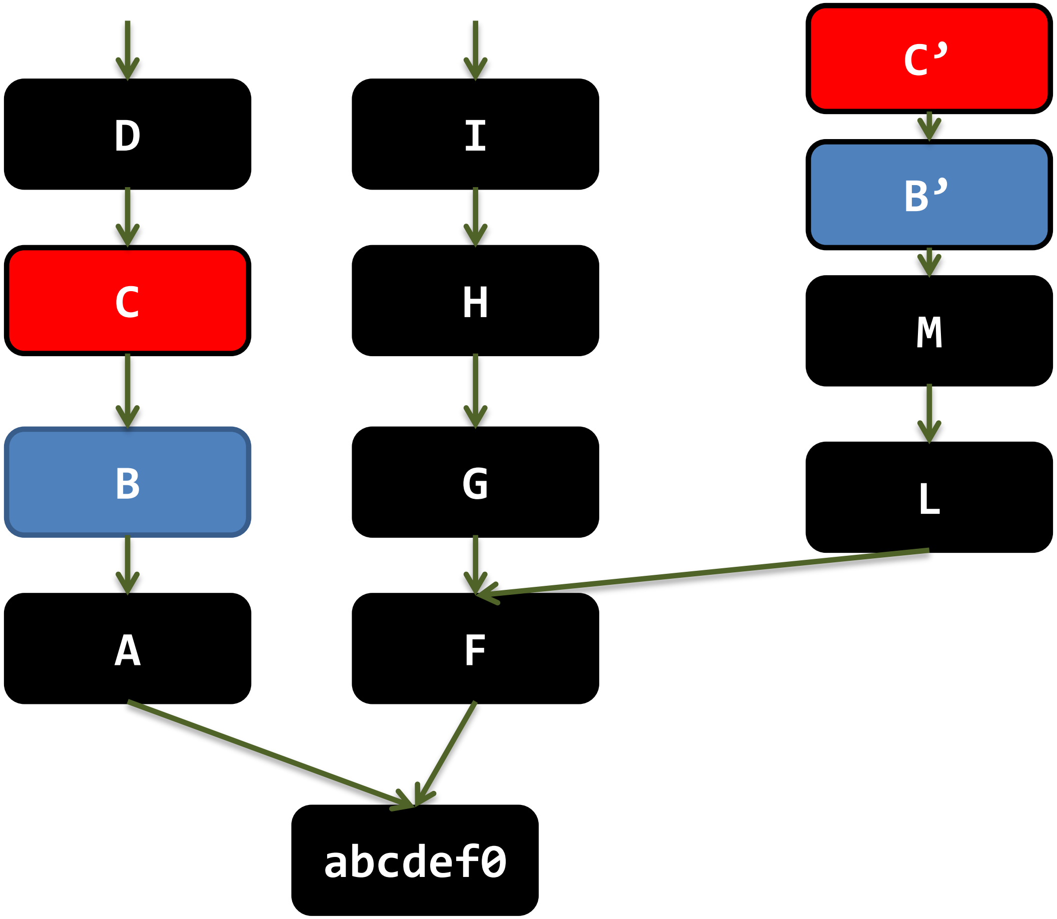
“Given one or more existing commits, apply the change each one introduces, recording a new commit for each.”



`git cherry-pick B C`

Cherry-Picking

“Given one or more existing commits, apply the change each one introduces, recording a new commit for each.”



`git cherry-pick B C`

Autres commandes pratiques pour débuter

- `git add –p`
- `git blame`
- `git diff`

S’organiser avec Git

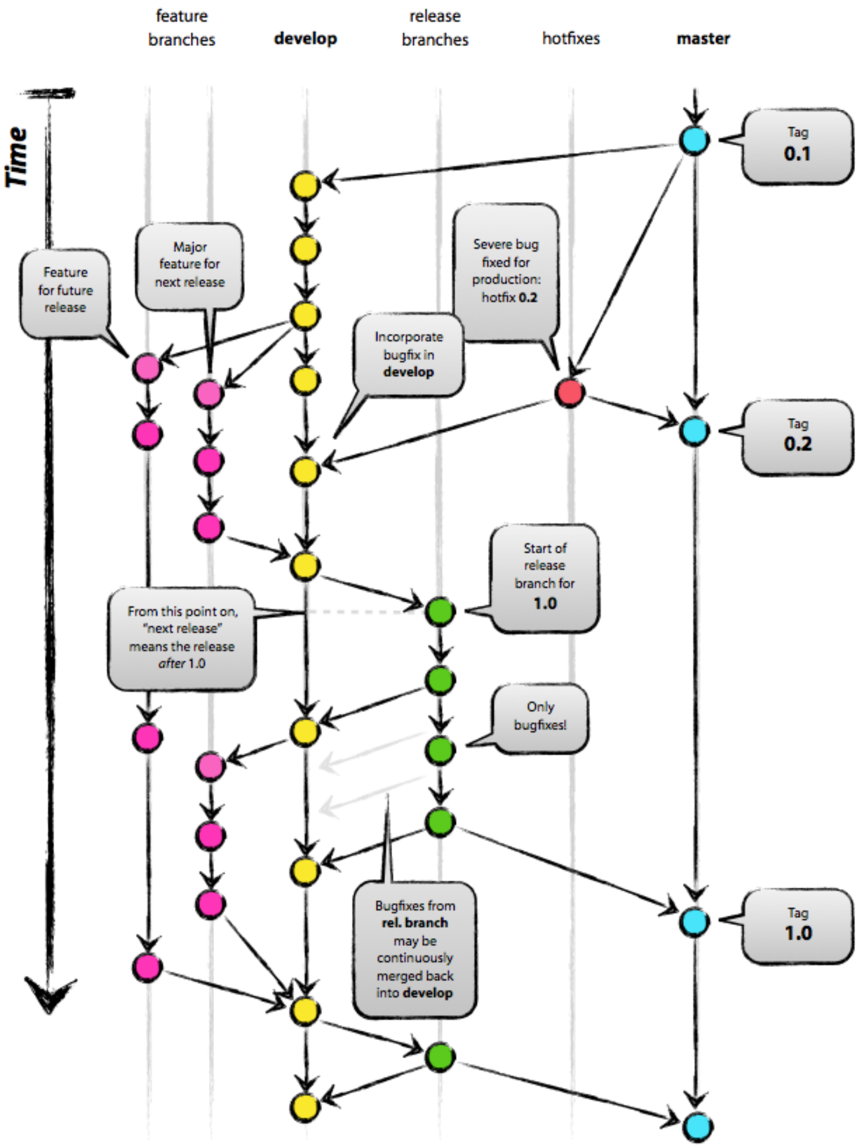
- « Workflow » ?
  - Organisation de répository
    - Branches et leurs intéractions
    - Progression du projet
  - Organisation de personnes
    - Qui a accès à quoi
    - Partage de modifications
    - Responsabilités

S’organiser avec Git

- « Workflow » ?
  - **Organisation de répository**
    - **Branches et leurs intéractions**
    - **Progression du projet**
  - Organisation de personnes
    - Qui a accès à quoi
    - Partage de modifications
    - Responsabilités

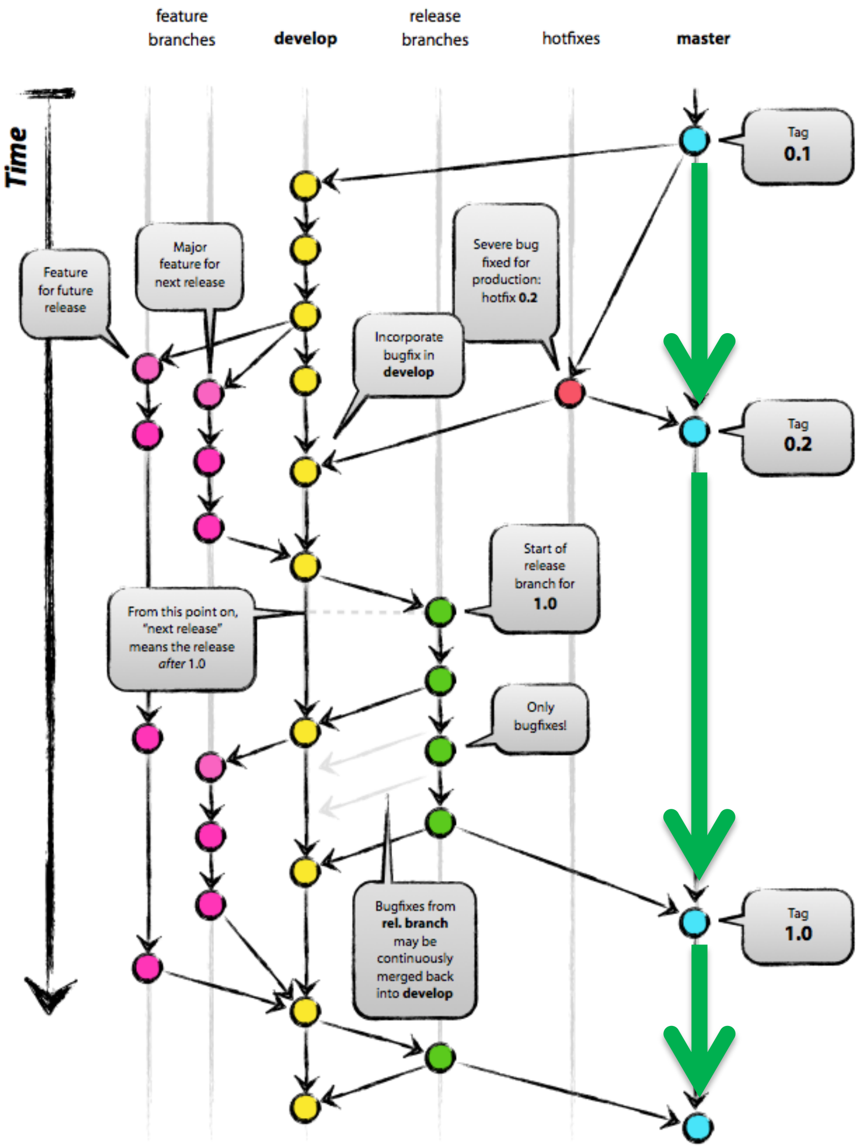
GitFlow

http://nvie.com/posts/a-successful-git-branching-model/


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Major feature for next release
Feature for future release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

GitFlow

http://nvie.com/posts/a-successful-git-branching-model/

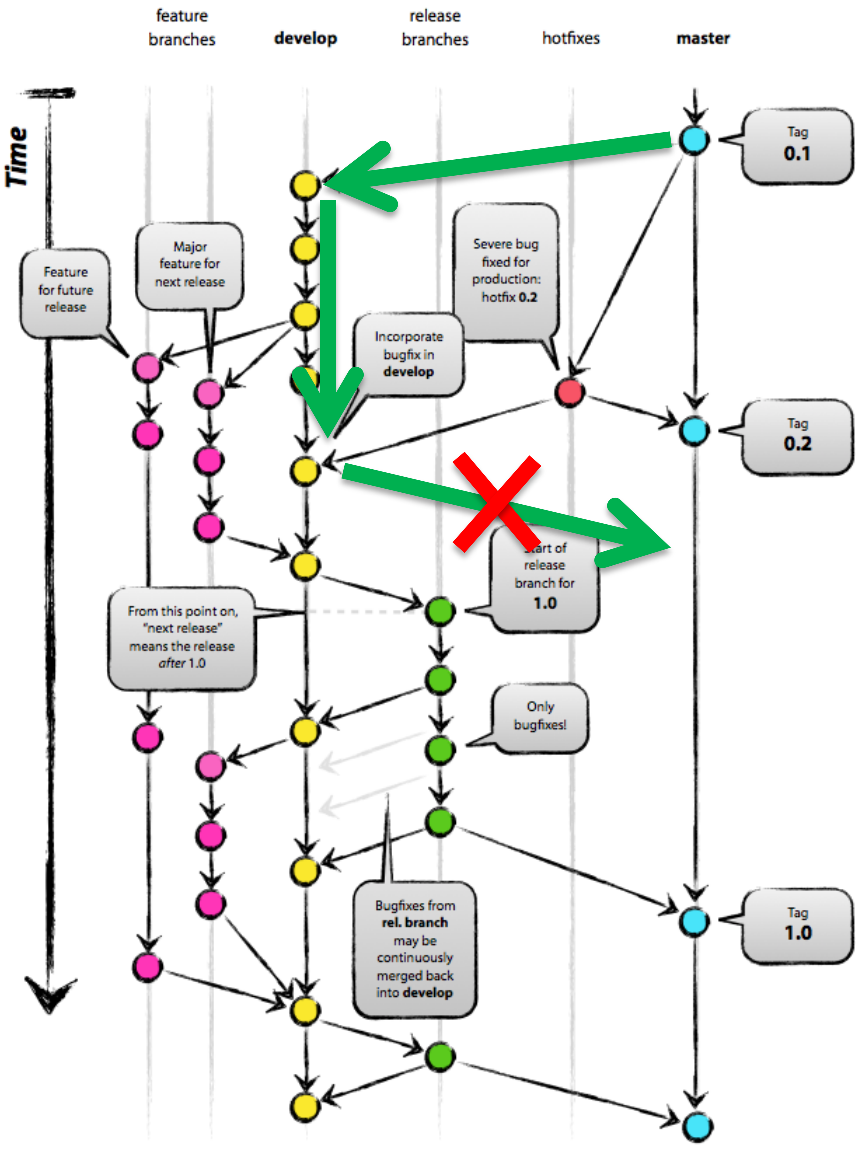


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

- Branche « master » = releases

GitFlow

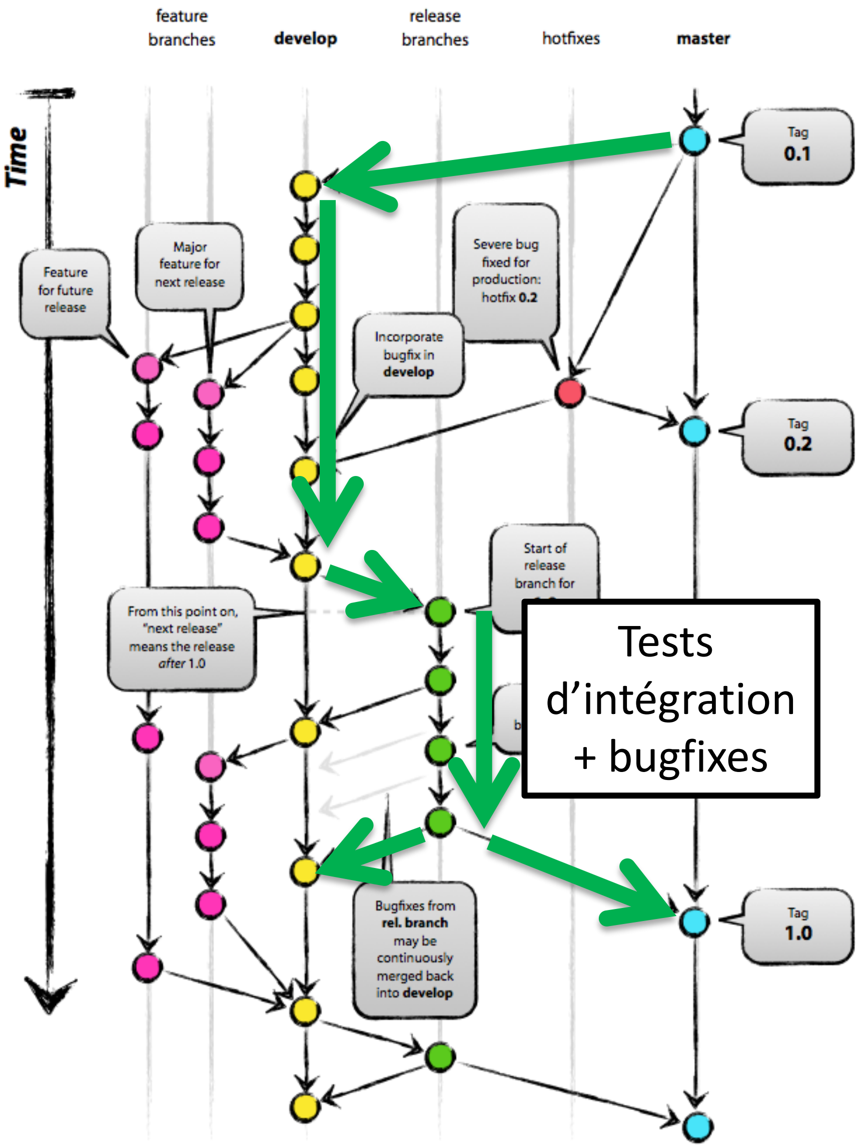
http://nvie.com/posts/a-successful-git-branching-model/


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

- Branche « master » = releases
- Branche « develop » = unstable

GitFlow

http://nvie.com/posts/a-successful-git-branching-model/

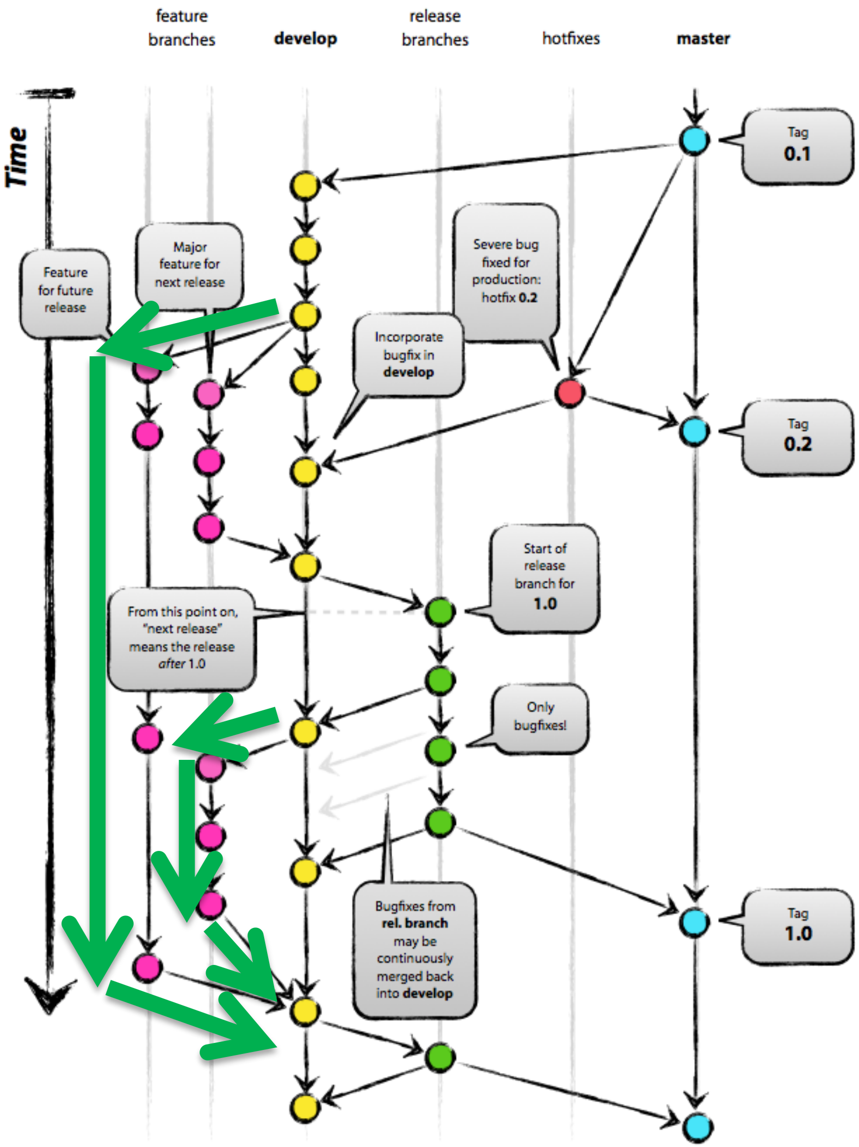

feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for
From this point on, "next release" means the release after 1.0
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

Tests d'intégration + bugfixes

- Branche « master » = releases
- Branche « develop » = unstable
- Branches:
  - Intégration

GitFlow

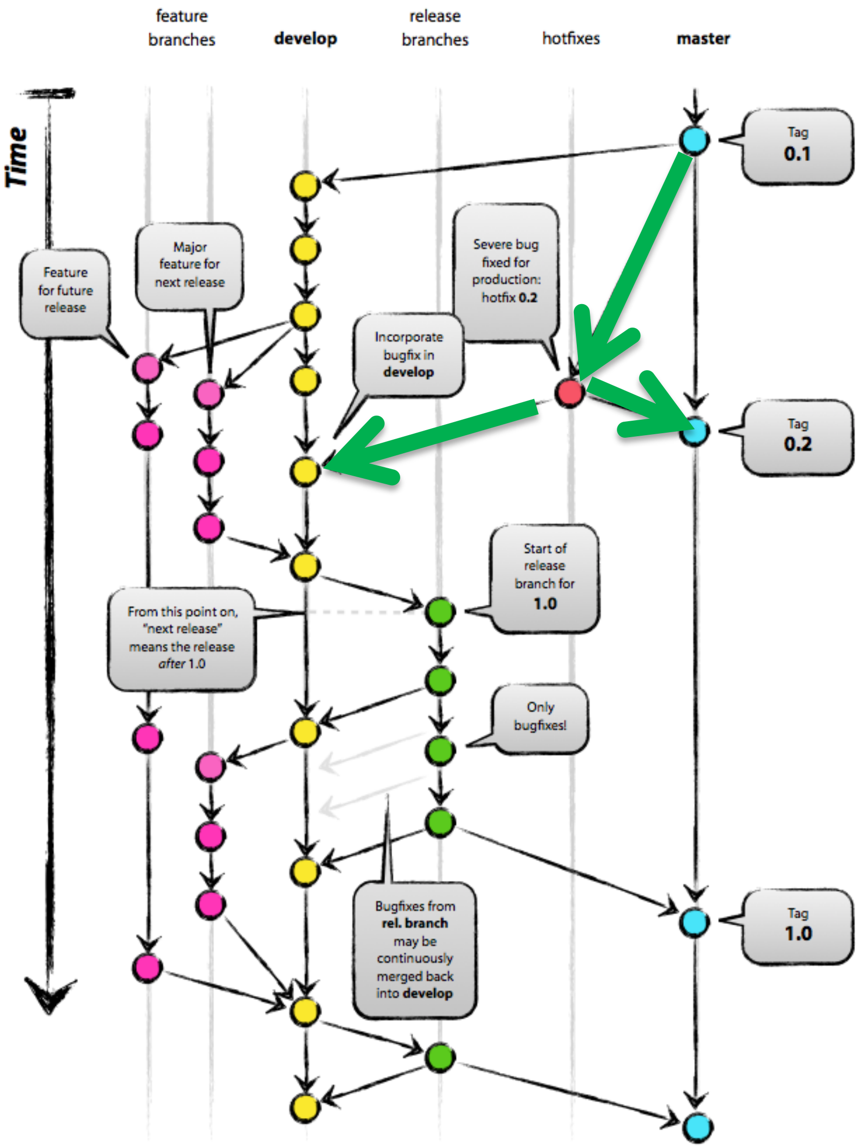
http://nvie.com/posts/a-successful-git-branching-model/


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Major feature for next release
Feature for future release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

- Branche « master » = releases
- Branche « develop » = unstable
- Branches:
  - Intégration
  - Features

GitFlow

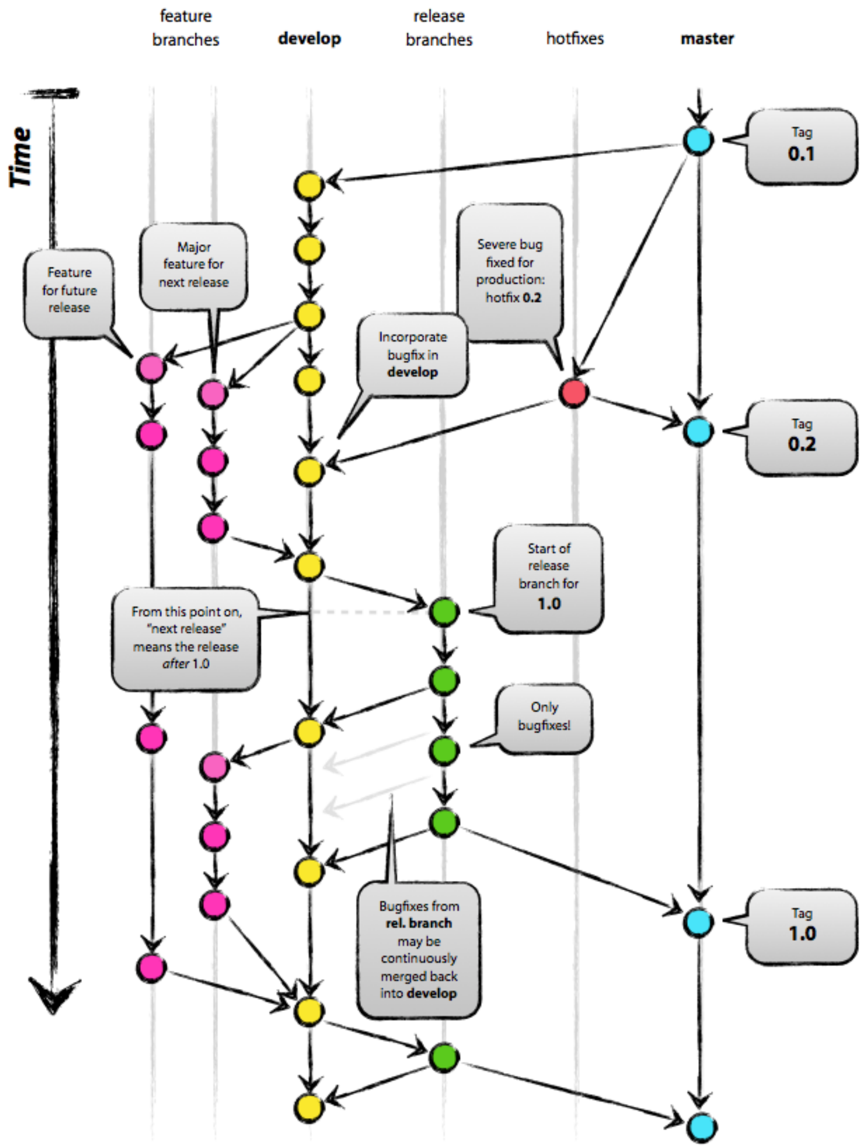
http://nvie.com/posts/a-successful-git-branching-model/


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

- Branche « master » = releases
- Branche « develop » = unstable
- Branches:
  - Intégration
  - Features
  - Hotfixes

GitFlow

http://nvie.com/posts/a-successful-git-branching-model/


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Major feature for next release
Feature for future release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

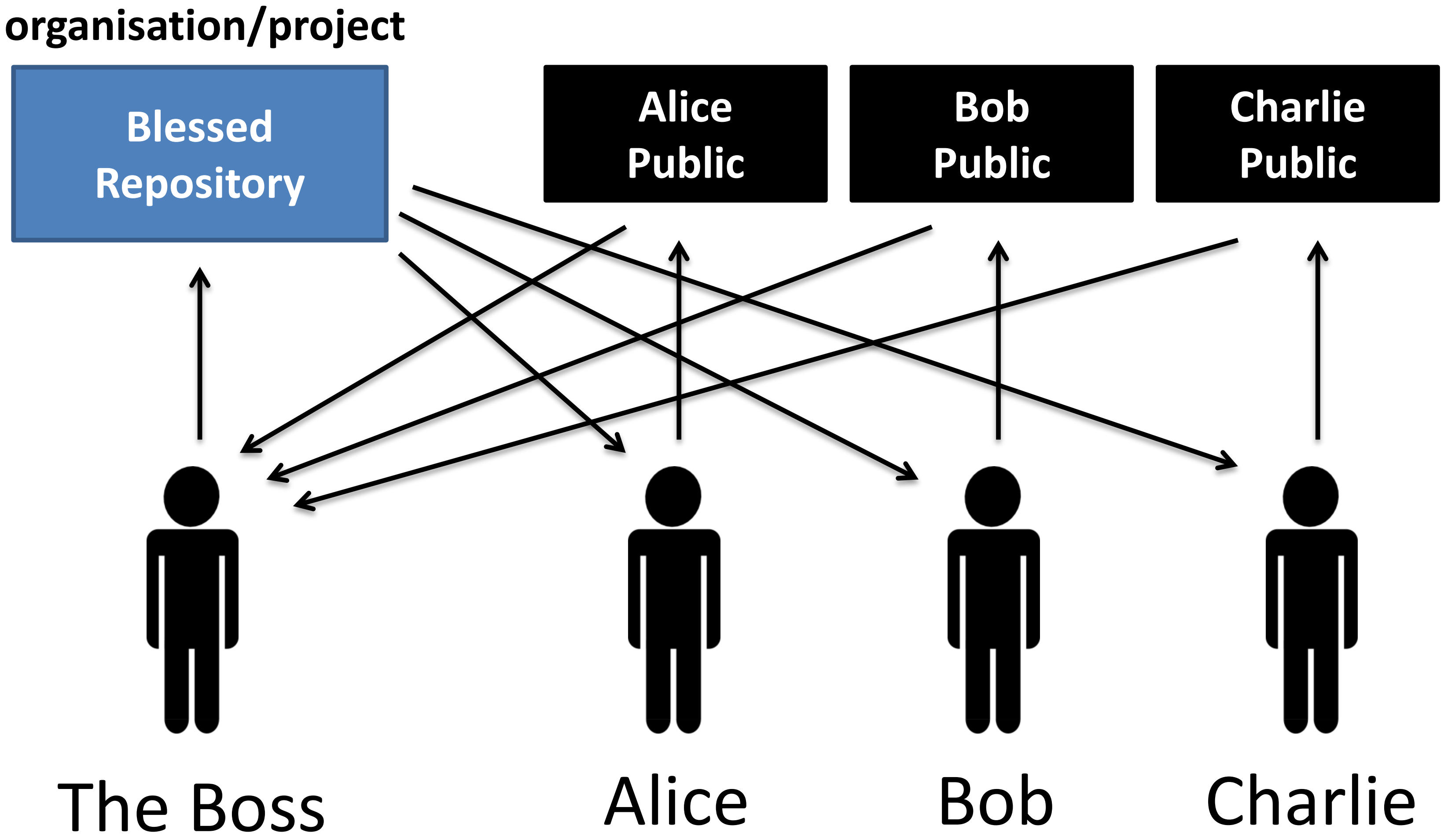
- Branche « master » = releases
- Branche « develop » = unstable
- Branches:
  - Intégration
  - Features
  - Hotfixes

- Intégration au shell
  https://github.com/nvie/gitflow

S’organiser avec Git

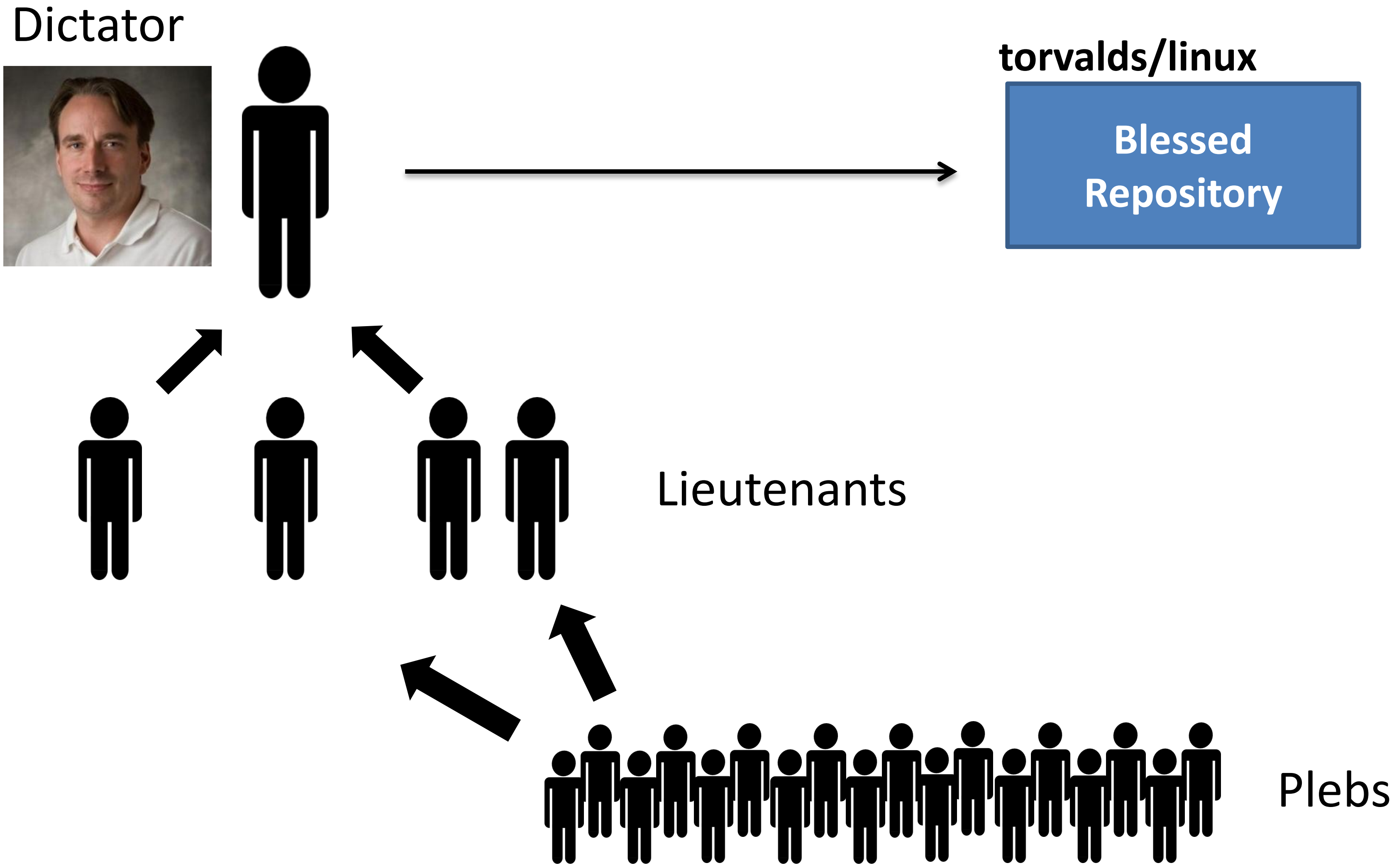
- « Workflow » ?
  - Organisation de répository
    - Branches et leurs intéractions
    - Progression du projet
  - **Organisation de personnes**
    - **Qui a accès à quoi**
    - **Partage de modifications**
    - **Responsabilités**

Integration Manager Workflow



Dictator & Lieutenants Workflow

« Network of Trust » (Linux)



Dictator

torvalds/linux

Blessed Repository

Lieutenants

Plebs

Le serveur git

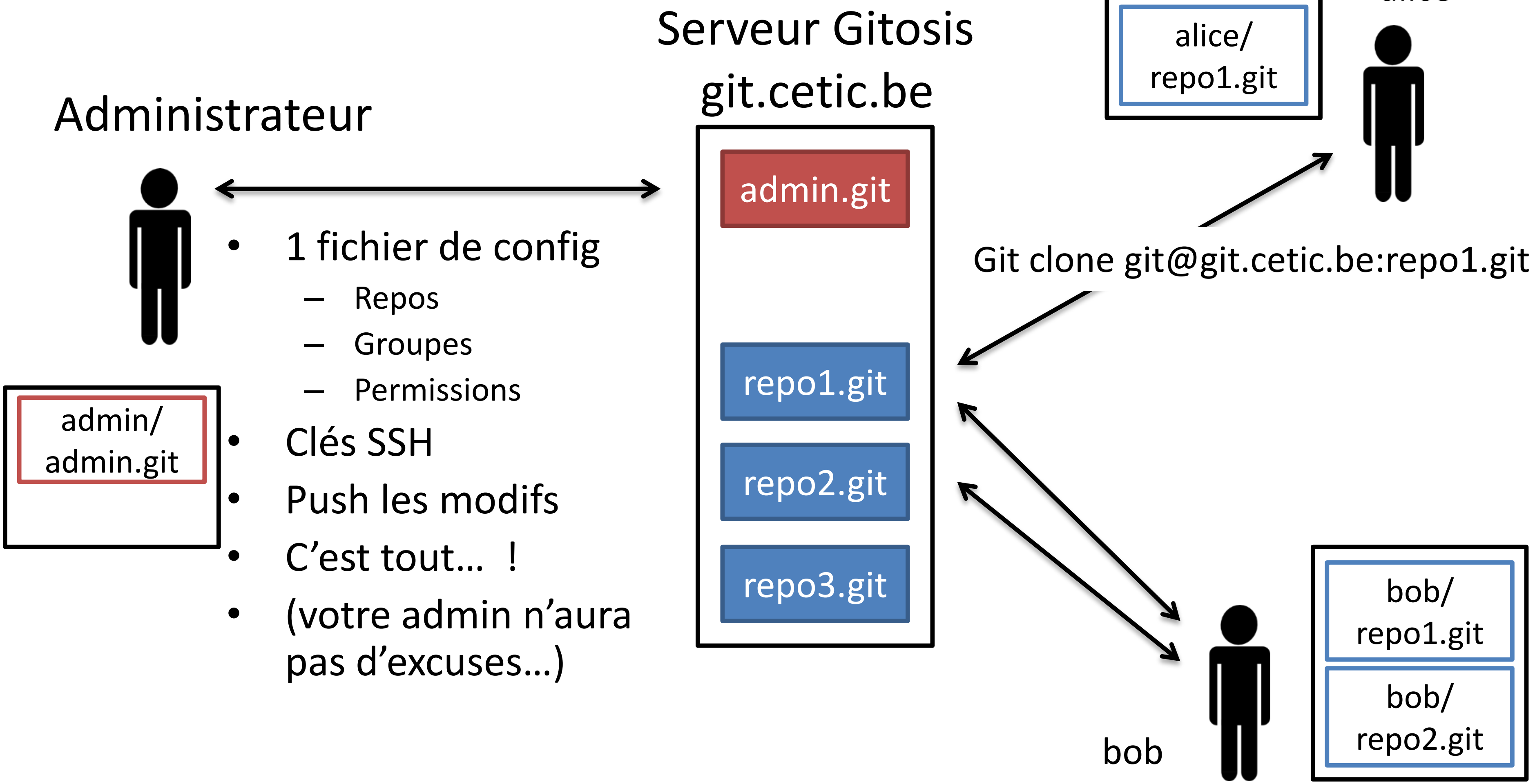
- « Distribué pur » possible mais sans intérêt
- On centralise, mais avec flexibilité:
  - Centraliser en interne ou à l’extérieur?
  - Repo public ou privé?
  - Avec quel outil?
  - Gratuit ou payant?

Le serveur git

- Interne, gratuit: gitosis, gitolite
- Interne, payant: gitorious
- Externe:
  - Github (privé uniquement en payant €€)
  - Bitbucket (repos privés gratuits < 5 devs)
  - Gitorious
- Pourquoi payer?
  - Interface web, forge, bug-tracking
  - Revue de code, commenter des commits

Gitosis/Gitolite

- Administré via git !



Administrateur

admin/
admin.git

Serveur Gitosis
git.cetic.be

admin.git

repo1.git

repo2.git

repo3.git

alice

alice/
repo1.git

Git clone git@git.cetic.be:repo1.git

bob

bob/
repo1.git

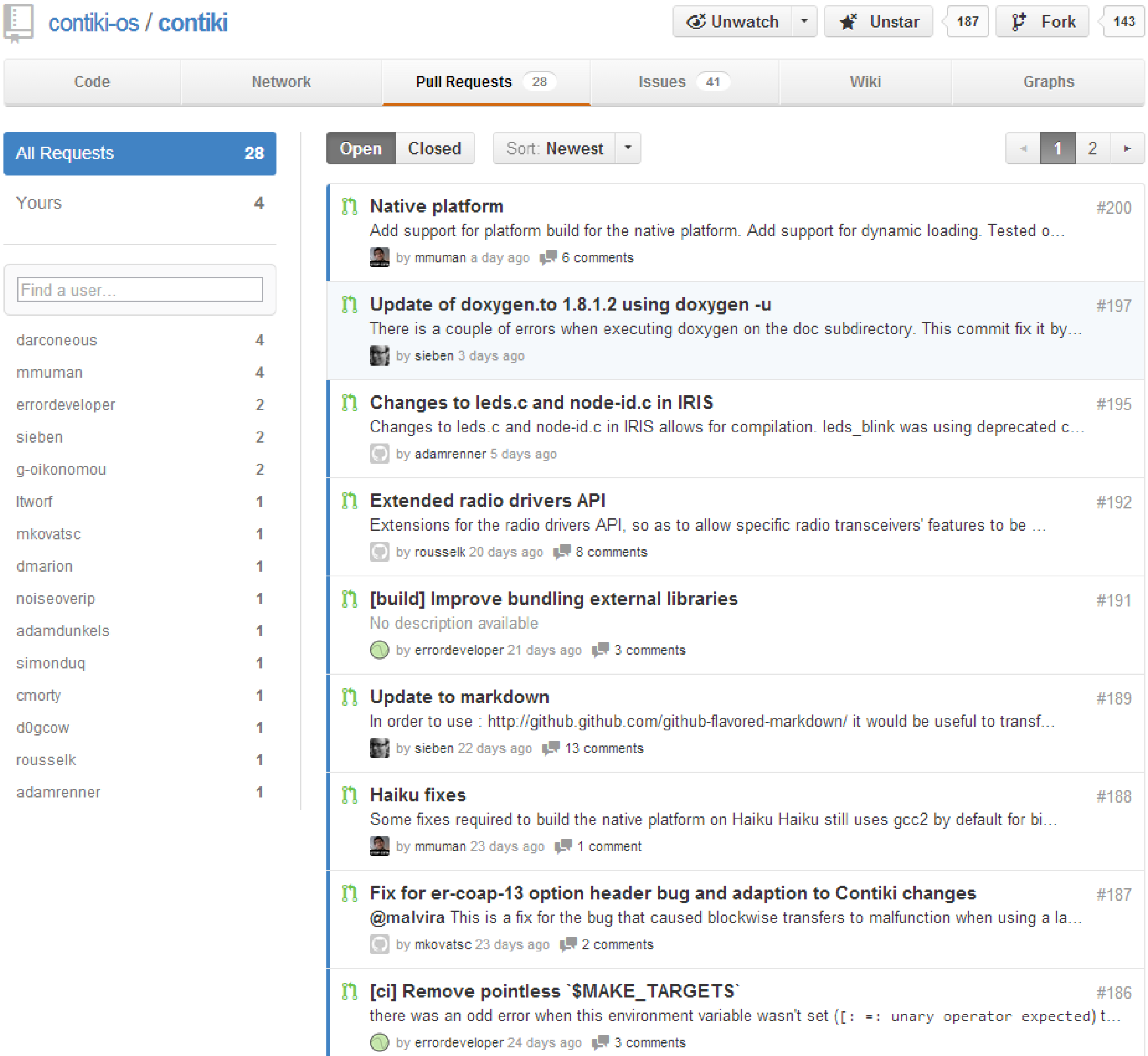
bob/
repo2.git

- 1 fichier de config
  - Repos
  - Groupes
  - Permissions
- Clés SSH
- Push les modifs
- C’est tout… !
- (votre admin n’aura pas d’excuses…)

GitHub

- Services:
  - Hébergement de repo git
  - Issue tracker, wiki, page web de projet
  - Forks & Pull requests
  - Hooks
- « Social Coding »
  - Popularité de repos
  - Visibilité de « forks »
  - Networking

GitHub: Pull Requests



All Requests	28
Yours	4

Find a user…

darconeous	4
mmuman	4
errordeveloper	2
sieben	2
g-oikonomou	2
ltworf	1
mkovatsc	1
dmarion	1
noiseoverip	1
adamdunkels	1
simonduq	1
cmorty	1
d0gcow	1
rousselk	1
adamrenner	1

Native platform #200
Add support for platform build for the native platform. Add support for dynamic loading. Tested o…
by mmuman a day ago 6 comments

Update of doxygen.to 1.8.1.2 using doxygen -u #197
There is a couple of errors when executing doxygen on the doc subdirectory. This commit fix it by…
by sieben 3 days ago

Changes to leds.c and node-id.c in IRIS #195
Changes to leds.c and node-id.c in IRIS allows for compilation. leds_blink was using deprecated c…
by adamrenner 5 days ago

Extended radio drivers API #192
Extensions for the radio drivers API, so as to allow specific radio transceivers' features to be …
by rousselk 20 days ago 8 comments

[build] Improve bundling external libraries #191
No description available
by errordeveloper 21 days ago 3 comments

Update to markdown #189
In order to use : http://github.github.com/github-flavored-markdown/ it would be useful to transf…
by sieben 22 days ago 13 comments

Haiku fixes #188
Some fixes required to build the native platform on Haiku Haiku still uses gcc2 by default for bi…
by mmuman 23 days ago 1 comment

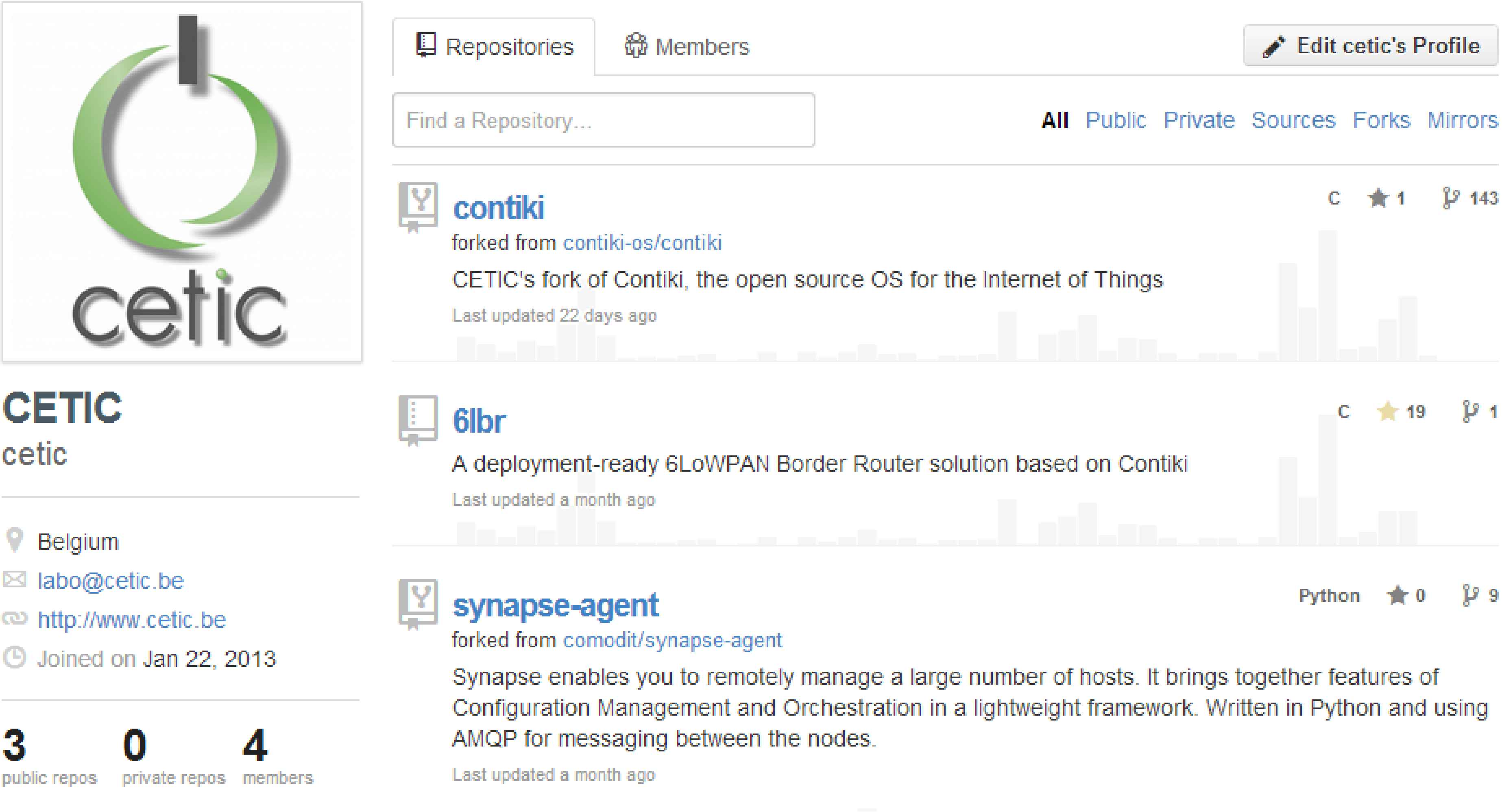
Fix for er-coap-13 option header bug and adaption to Contiki changes #187
@malvira This is a fix for the bug that caused blockwise transfers to malfunction when using a la…
by mkovatsc 23 days ago 2 comments

[ci] Remove pointless `$MAKE_TARGETS` #186
there was an odd error when this environment variable wasn't set (`[: =: unary operator expected`) t…
by errordeveloper 24 days ago 3 comments

GitHub: Organisations

- Regroupement de repos
- Gestion d’accès par équipes
- Visibilité accrue
- Tarifs payants adaptés aux entreprises

GitHub: Organisation CETIC



CETIC
cetic

Belgium
labo@cetic.be
http://www.cetic.be
Joined on Jan 22, 2013

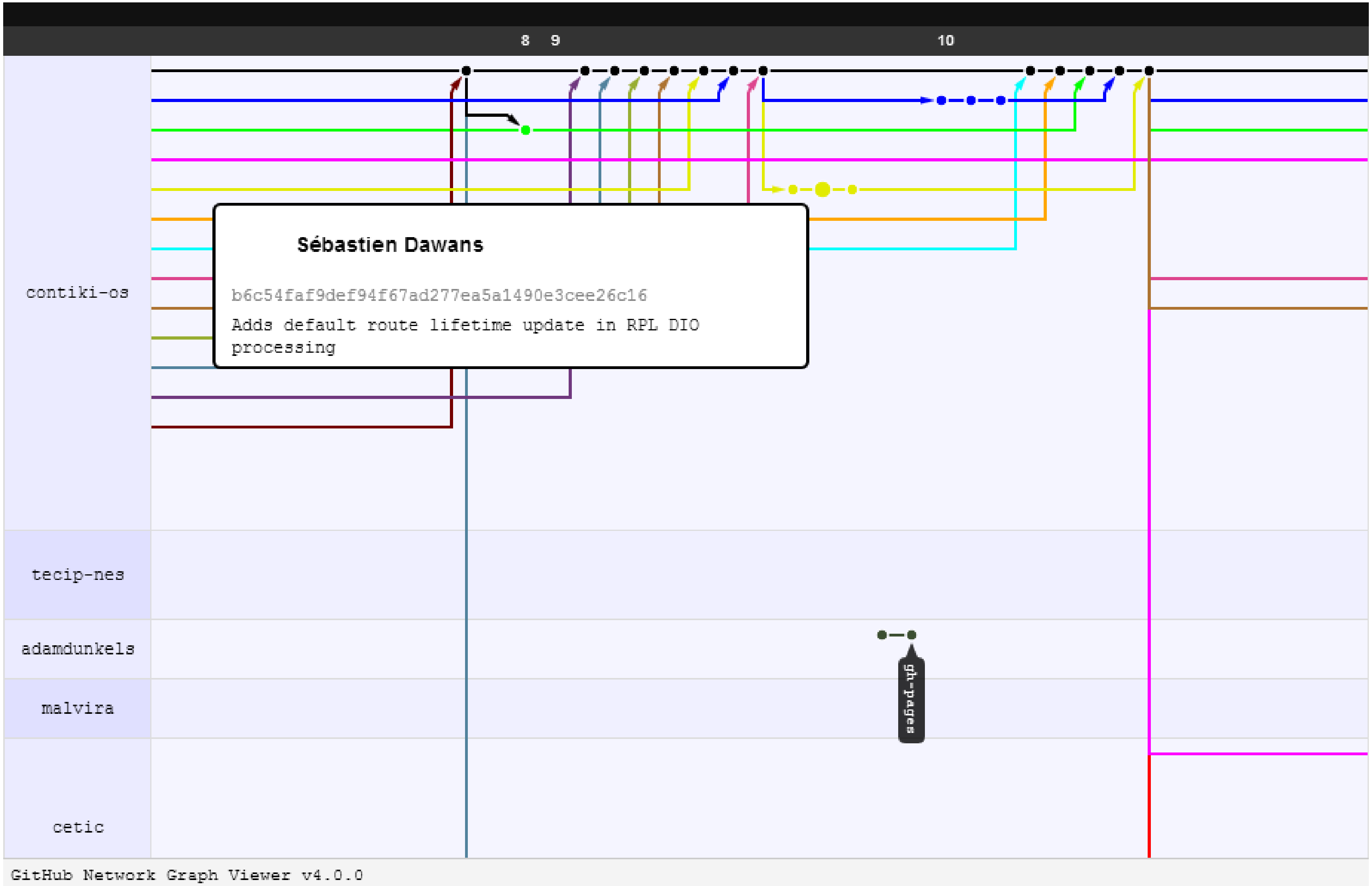
3 public repos **0** private repos **4** members

contiki C ★ 1 143
forked from contiki-os/contiki
CETIC's fork of Contiki, the open source OS for the Internet of Things
Last updated 22 days ago

6lbr C ★ 19 1
A deployment-ready 6LoWPAN Border Router solution based on Contiki
Last updated a month ago

synapse-agent Python ★ 0 9
forked from comodit/synapse-agent
Synapse enables you to remotely manage a large number of hosts. It brings together features of Configuration Management and Orchestration in a lightweight framework. Written in Python and using AMQP for messaging between the nodes.
Last updated a month ago

GitHub: Network Graph



Sébastien Dawans

b6c54faf9def94f67ad277ea5a1490e3cee26c16

Adds default route lifetime update in RPL DIO processing

GitHub Network Graph Viewer v4.0.0

GitHub: Code Review

- Thread de discussion
  - Par Pull Request
  - Par commit (1 ou plusieurs par pull request)
  - Avant ou après le merge
- Annotation de code
- Notifications par mail et dashboard web

Your Connection to ICT Research

MERCI

CETIC

Aéropôle de Charleroi-Gosselies
Rue des Frères Wright, 29/3
6041 Gosselies
info@cetic.be

www.cetic.be

Your Connection to ICT Research

MERCI

Questions?

CETIC

Aéropôle de Charleroi-Gosselies
Rue des Frères Wright, 29/3
6041 Gosselies
info@cetic.be

www.cetic.be

Getting Started with Git

- Book: Pro Git. www.git-scm.com
- Selected Tutorials:
  - Git Immersion http://gitimmersion.com/
  - Learn Git Branching (interactive) http://pcottle.github.io/learnGitBranching/
- More material:
  - teach.github.com
- This presentation + cheat sheets and future stuff
  - https://github.com/cetic/git-slides

Références

- Torvalds, L. **Git.** Google Tech Talk, 14-05-2007
- Shacon, S. **Pro Git**. git-scm.org