

Distributed Versioning with git

Jonas Weber

HFU::UnFUG

27. 10. 2011

Table of Contents

1 Versioning

2 Using git

3 Workshop

Table of Contents

1 Versioning

2 Using git

3 Workshop

Why?

- previous versions
 - ▶ accessible
 - ▶ archived
- collaborative working

Approaches

Position of Repository

- local
- centralized
- distributed

Approaches

Position of Repository

- local
- centralized
- distributed

Storage Method

- Directed Acyclic Graph (DAG)
- Delta Storage

DAG vs Δ

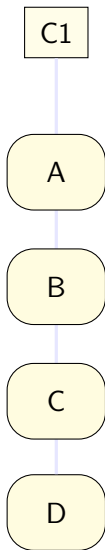
Δ

store differences made in this commit

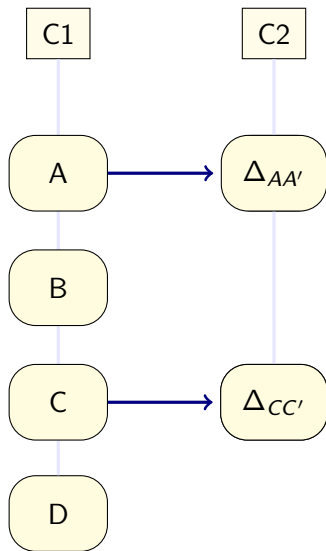
DAG

store file contents

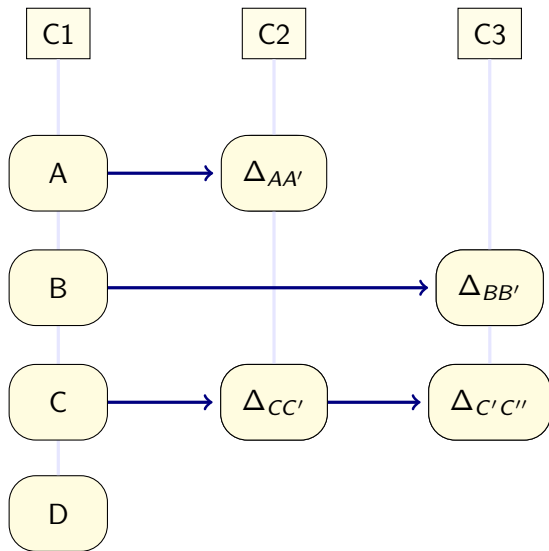
△ storage



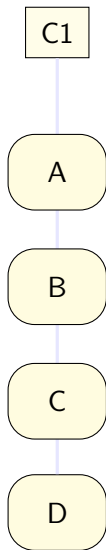
Δ storage



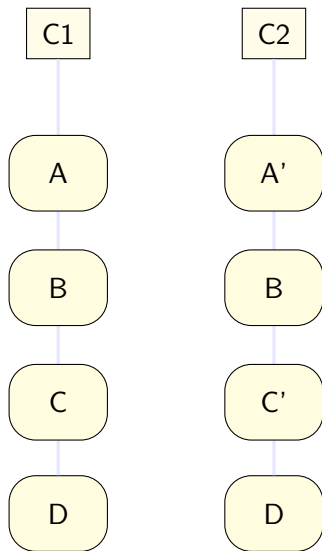
Δ storage



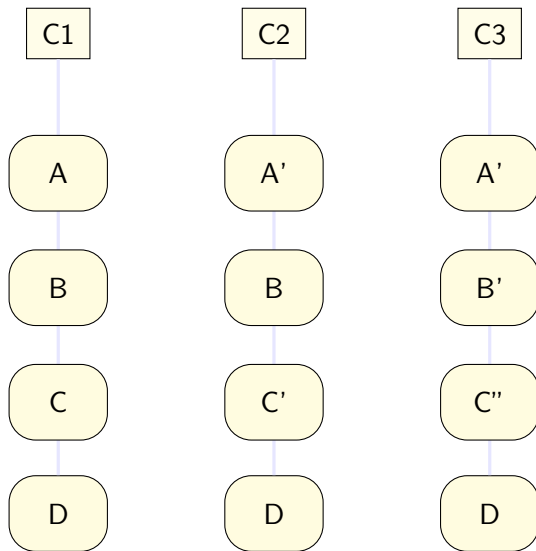
DAG storage



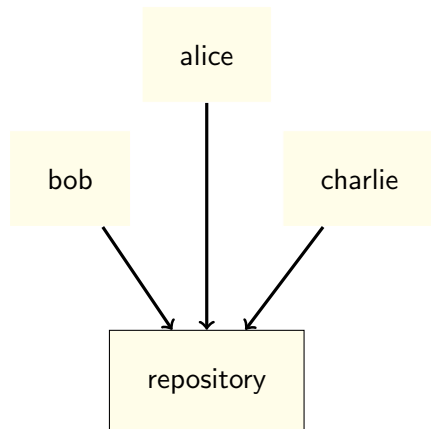
DAG storage



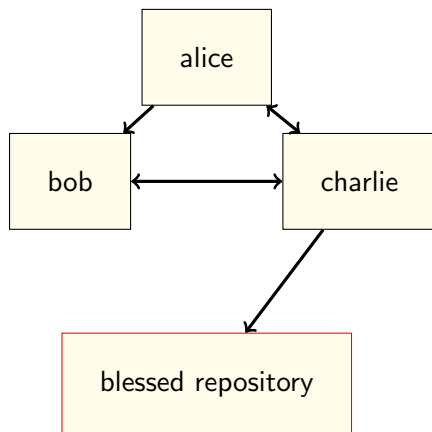
DAG storage



Centralized



Decentralized



VCS comparison

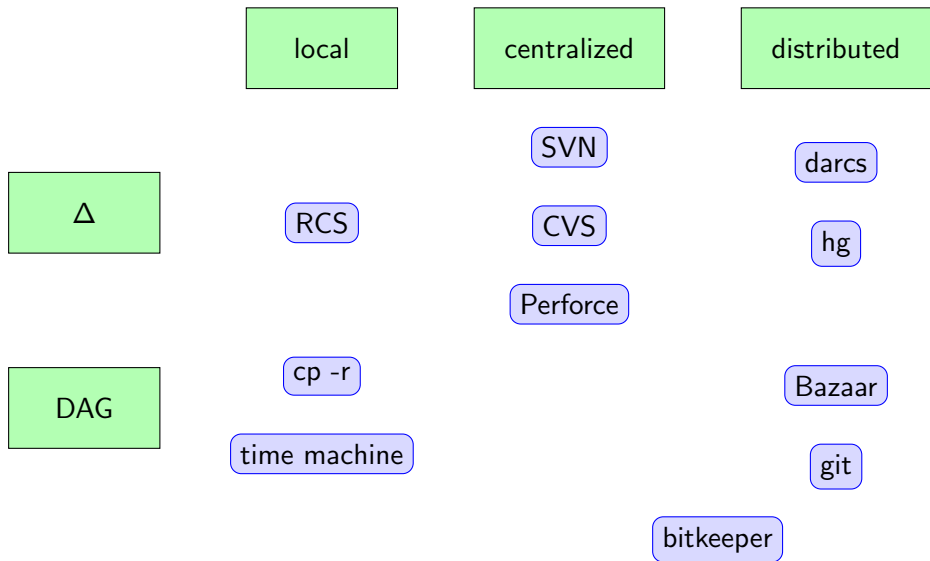


Table of Contents

1 Versioning

2 Using git

3 Workshop

git ?

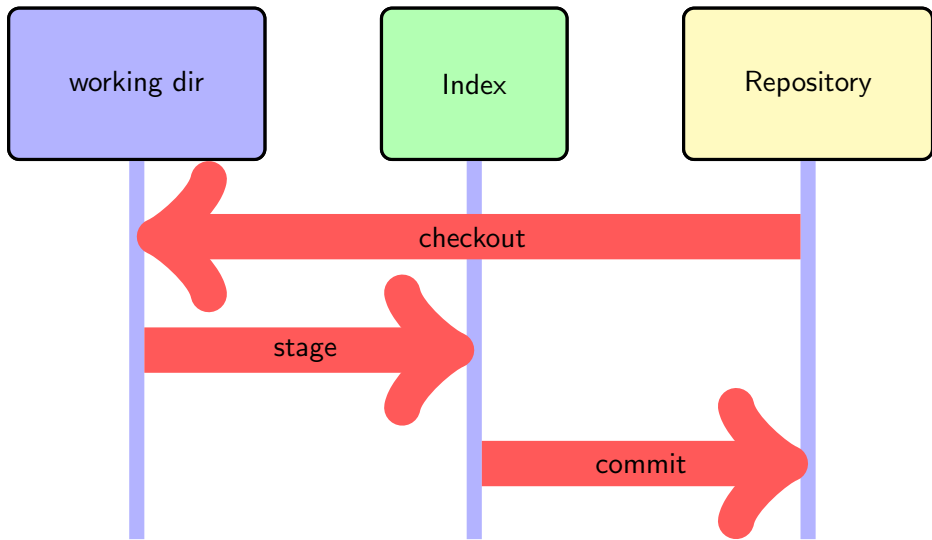
- bitkeeper: no longer free (as in beer)
- started by Linus Torvalds for the Linux Kernel
- now: developed by community

classification

- DAG storage
- decentralized

Commands: SVN vs Git

<code>svn checkout</code>	<code>git clone</code>
<code>svn update</code>	<code>git pull</code>
<code>svn commit</code>	<code>git commit -a</code>
<code>svn revert</code>	<code>git checkout --</code>



Working with a Repository

Getting a repository

```
git clone URL # copies complete history
```

doint some changes to file README ...

Committing Changes

```
git add README # adds README to INDEX  
git commit -m 'changed README' # commits
```

Pushing back to origin

```
git push
```

Branching

Creating a branch

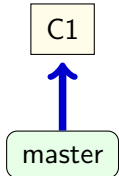
```
git checkout -b crazyidea
```

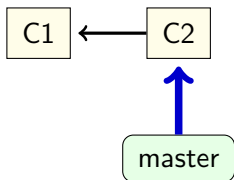
yep, it's that easy!

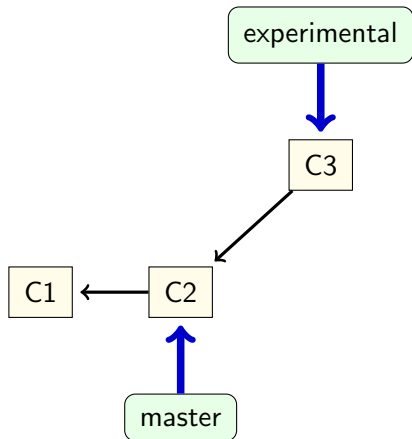
Integrate Changes

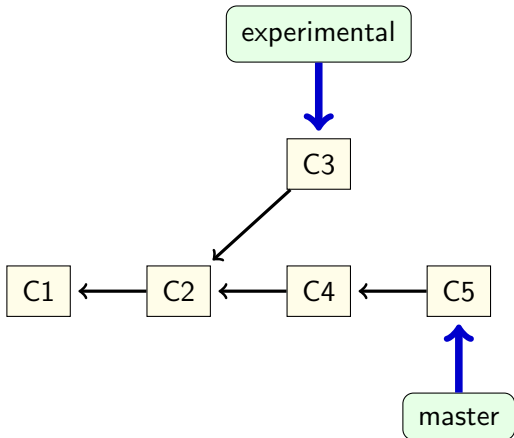
Merge

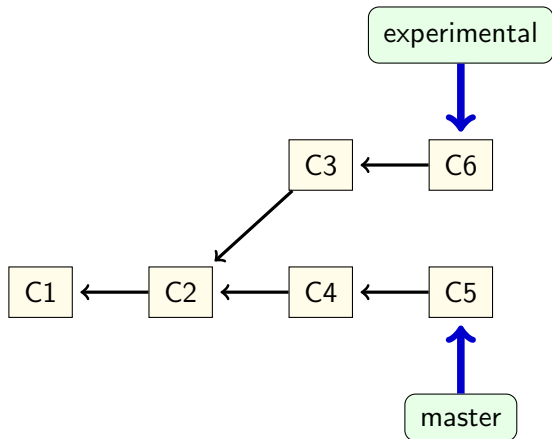
```
git checkout master  
git merge <branch> # creates commit
```

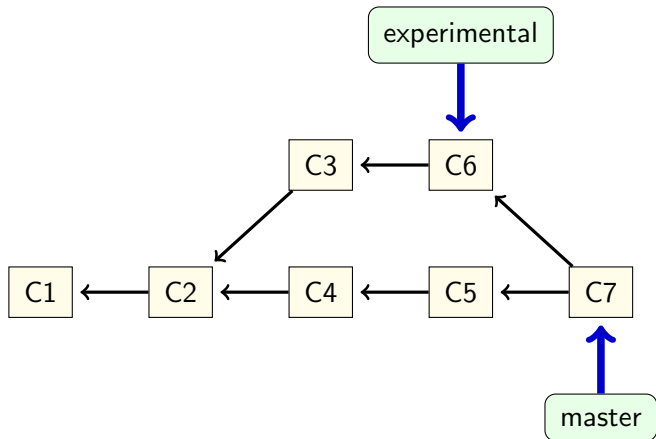


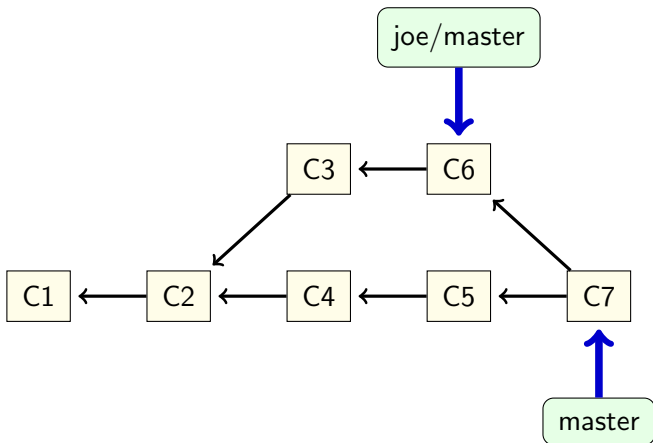












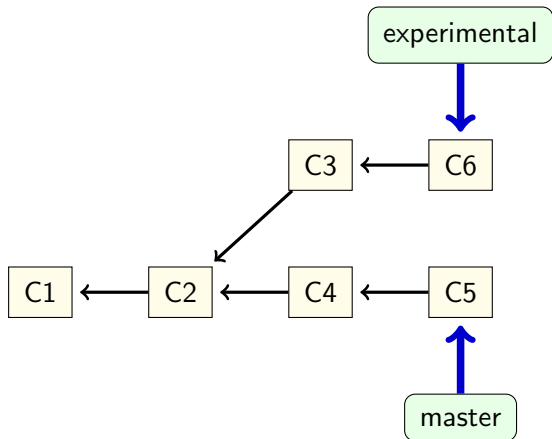
Integrate Changes

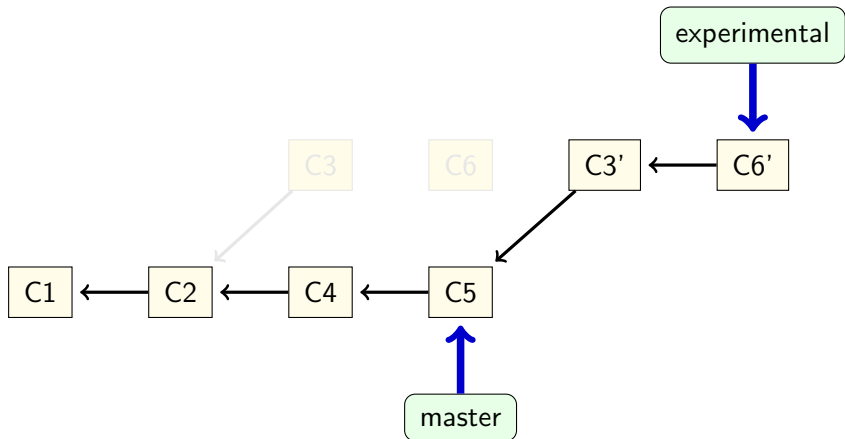
Merge

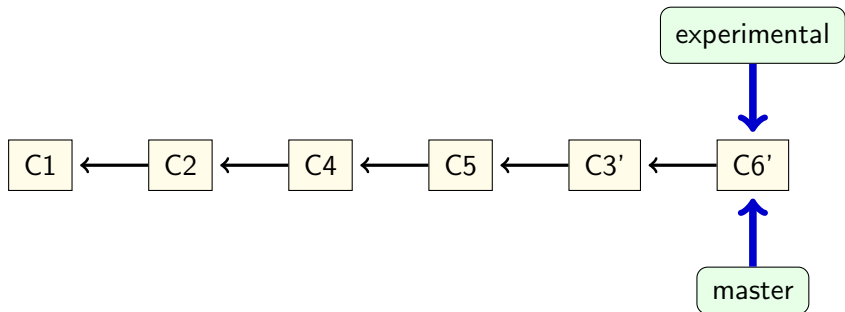
```
git checkout master  
git merge <branch> # creates commit
```

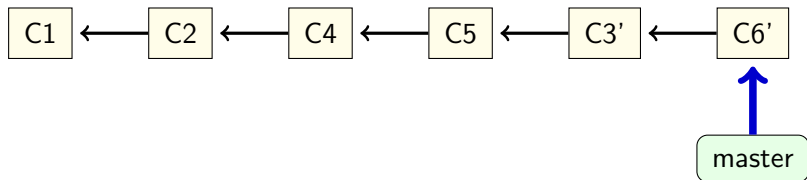
Rebase

```
git checkout <branch>  
git rebase master # moves branch at master
```









How Things are stored

Where

Filename: SHA1-Hash of content.

Object-Types

- Blob
- Tree
- Commit
- Tag

Object-Types

Blob

content of a file

Tree

list of blobs

Commit

points to tree plus metadata

Tag

points to commit plus metadata

Submodules

importing another repository by checkout id.

adding a submodule

```
git submodule add <URL> <dir>
git add .gitmodules <dir>
git commit -m "adding subproject"
```

Working with submodules

after a clone:

```
git submodule init  
git submodule update
```

to update:

```
cd <dir>  
git fetch  
git checkout <ID>  
cd ..  
  
git add <dir> ; git commit -m "updating submodule"
```

Table of Contents

1 Versioning

2 Using git

3 Workshop

Workshop Instructions

to do the workshop again, please download "regenerate_repo.sh.gz" and execute.

Instructions are in the files called README (in german!)

Any unanswered questions?

Contact

Jonas Weber

jonas.weber@hs-furtwangen.de

Links

<http://www.git-scm.com>

<http://github.com>