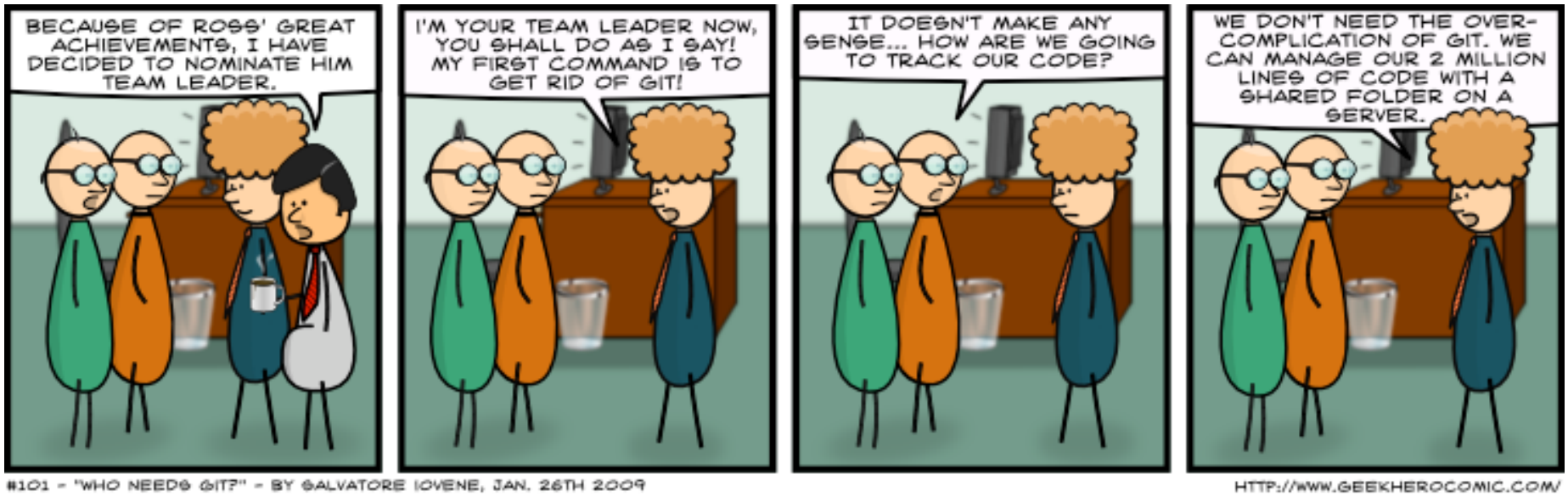
Git: Part 1
Overview & Object Model

These slides were largely cut-and-pasted from http://excess.org/article/2008/07/ogre-git-tutorial/ , with some additions from other sources. I have deleted a lot from the cited tutorial, and recommend that you listen to the entire tutorial on line, if you can.

Who needs Git?



http://www.geekherocomic.com/2009/01/26/who-needs-git/

Topics

- What is Git?
- SCM concepts used in Git
- Git object model
- Representation of Git objects as files
- References

Git

- A collection of tools developed by Linux kernel group for SCM
  - Now used by several other groups, and apparently growing in popularity
- Actually implements a replicated versioned file system
- Can be used to implement a variety of software configuration management models and workflows

Git Flavor

- A collection of many tools
- Evolved from scripts
- Suited to a C programmer's mentality
- Everything is exposed and accessible
- Very flexible
  - You can do anything the model permits
  - Including shooting yourself in the foot
- **Need to understand the underlying model**

Git has a lot of commands

- Learn a core subset of them
- And one of the GUI tools (e.g., gitk)
- Then learn the rest as you need them

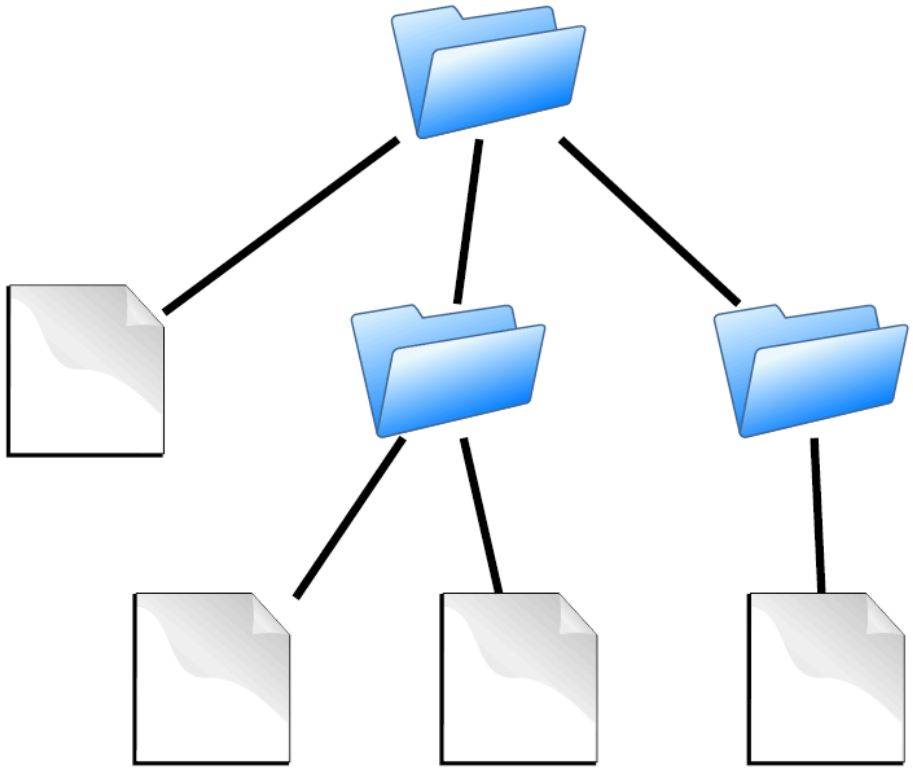
Groups of Git commands

- Setup and branch management
  - init, checkout, branch
- Modify
  - add, delete, rename, commit
- Get information
  - status, diff, log
- Create reference points
  - tag, branch

Source code

contains

- Directories
- Files



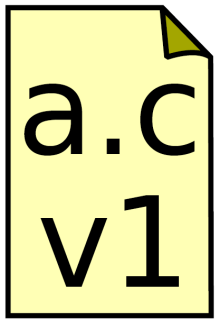
is the substance of a software configuration

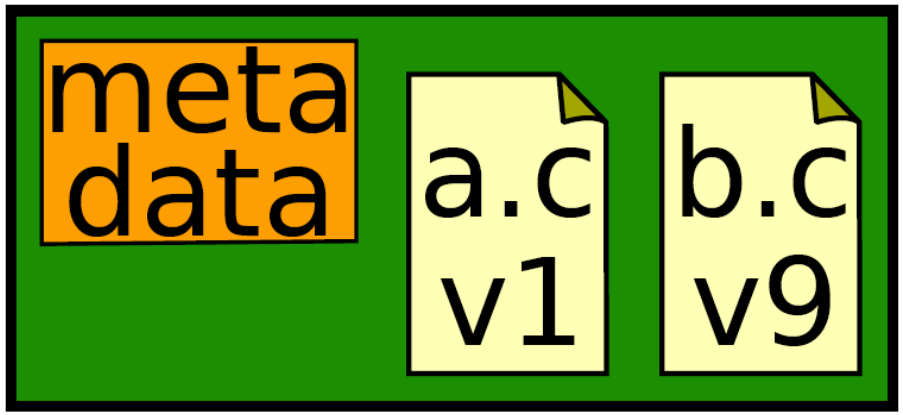
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Repository

Contains

- files
- commits





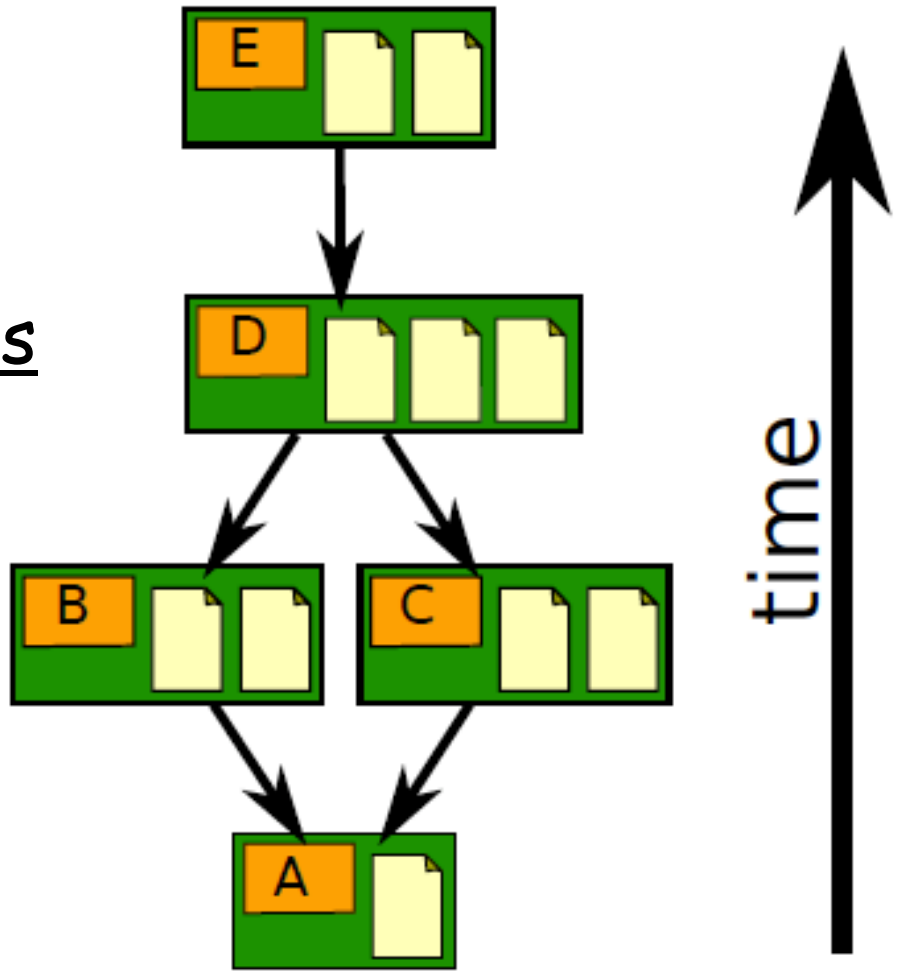
records history of changes to configuration

http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Repository

Contains

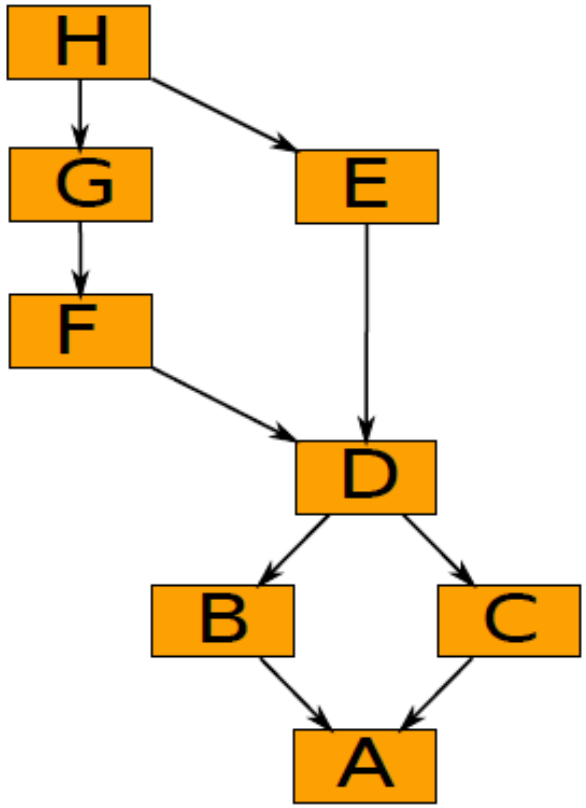
- files
- commits
- ancestry relationships



http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Ancestry relationships

form a directed acyclic graph (DAG)

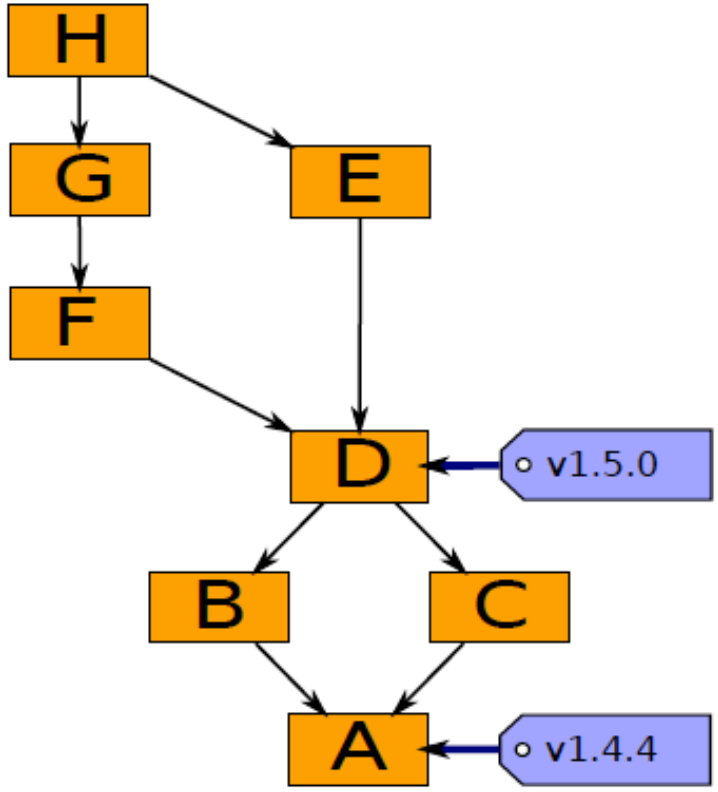


http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Ancestry graph features

Tags

- identify versions of interest
- including "releases"

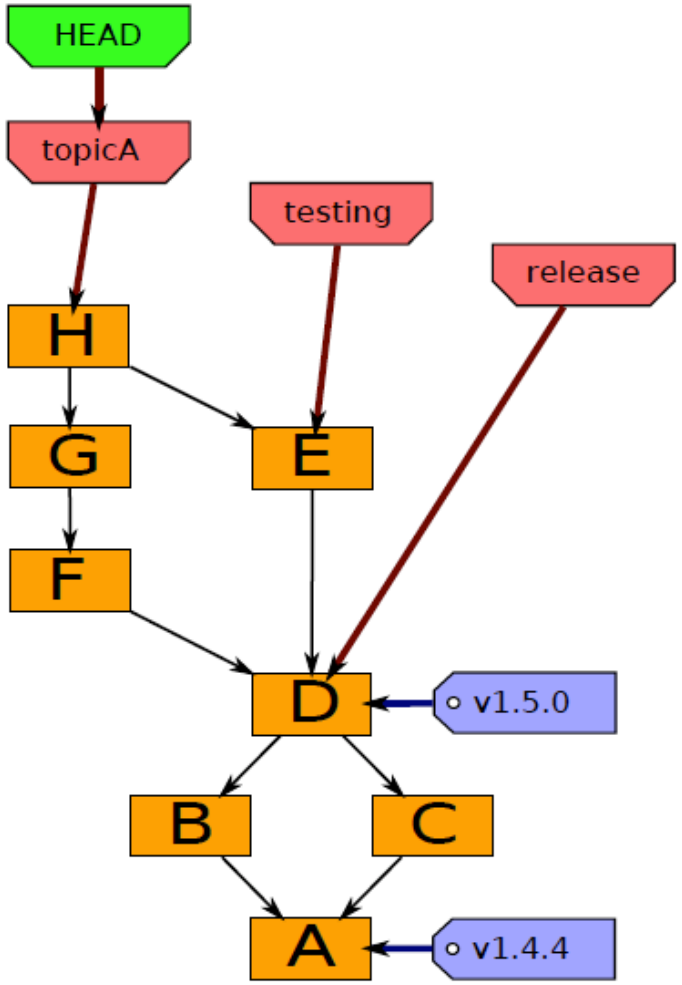


http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Ancestry graph features

HEAD

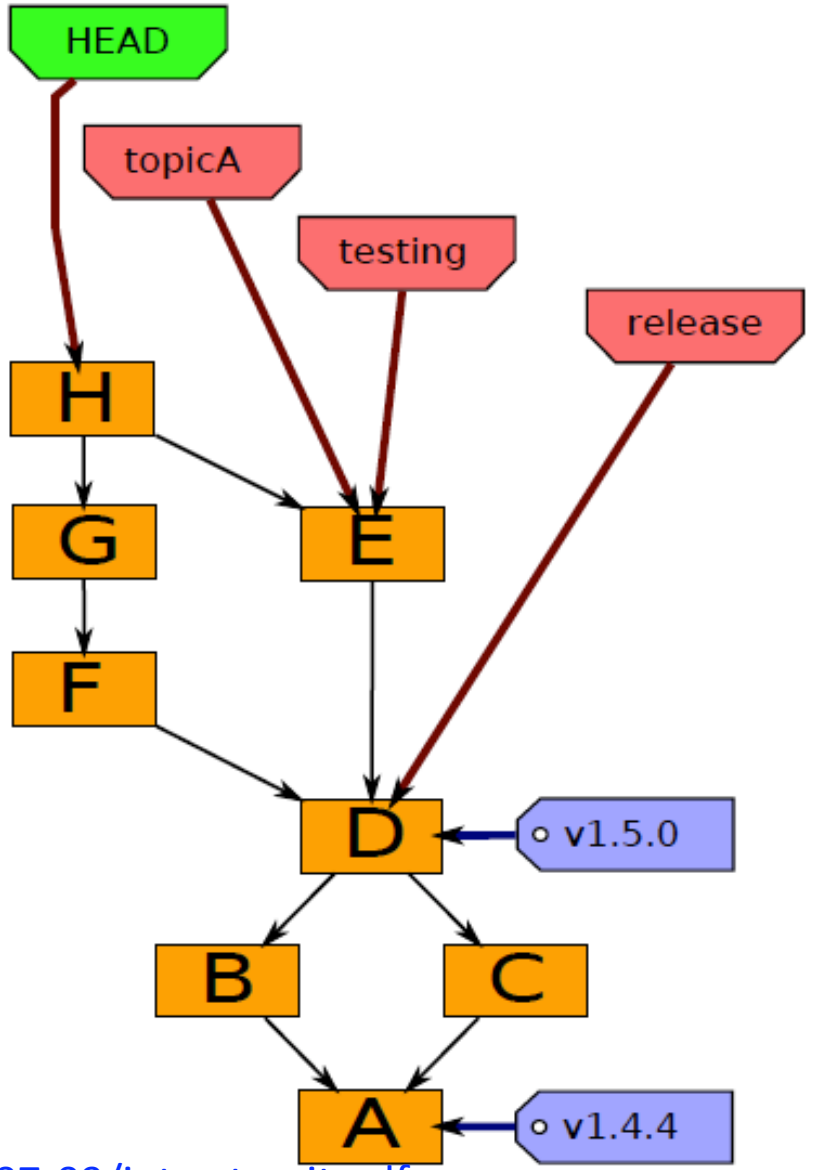
- is current checkout
- usually points to a branch



http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Head may point to any commit

In this case it is said to be <u>detached.</u>

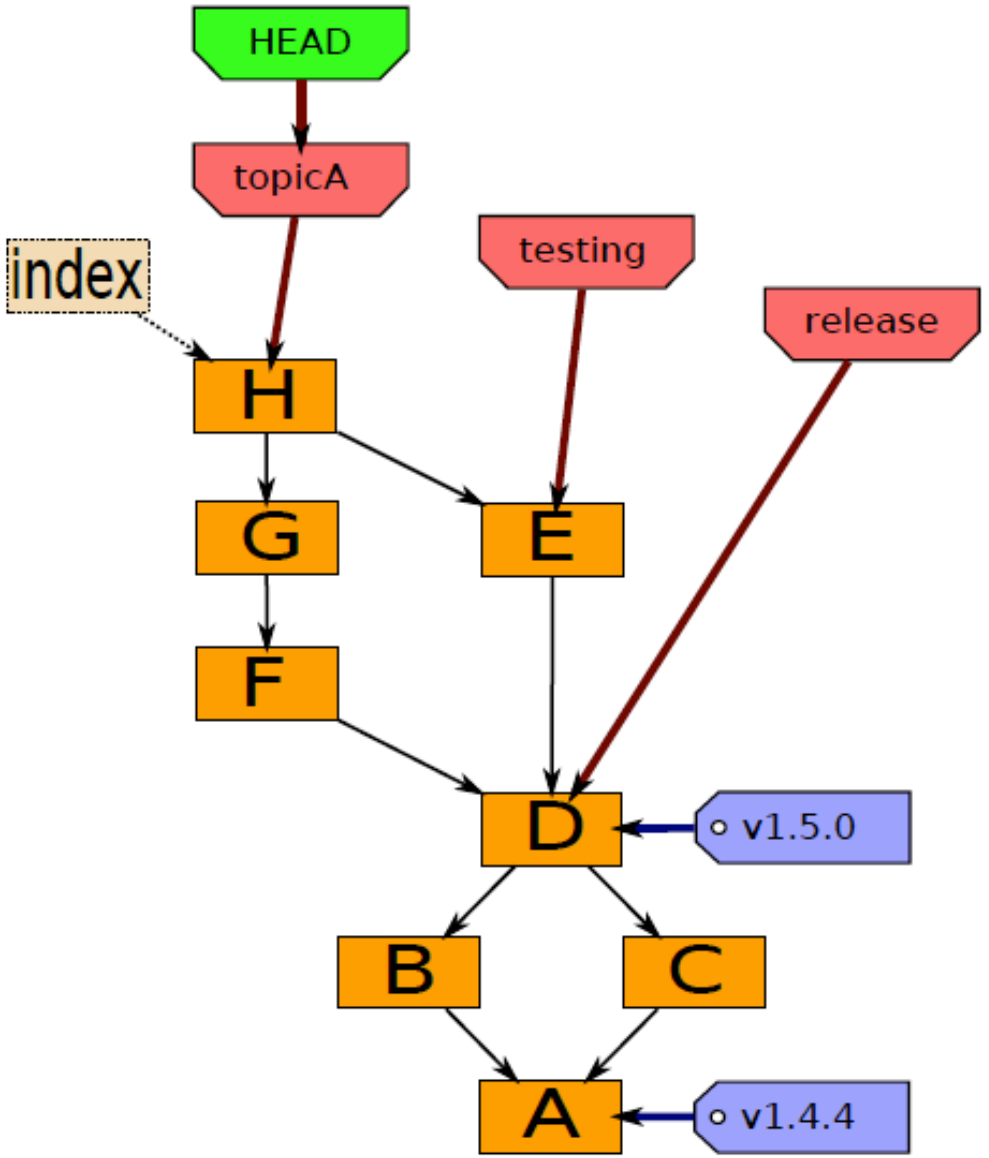


http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Git components

Index

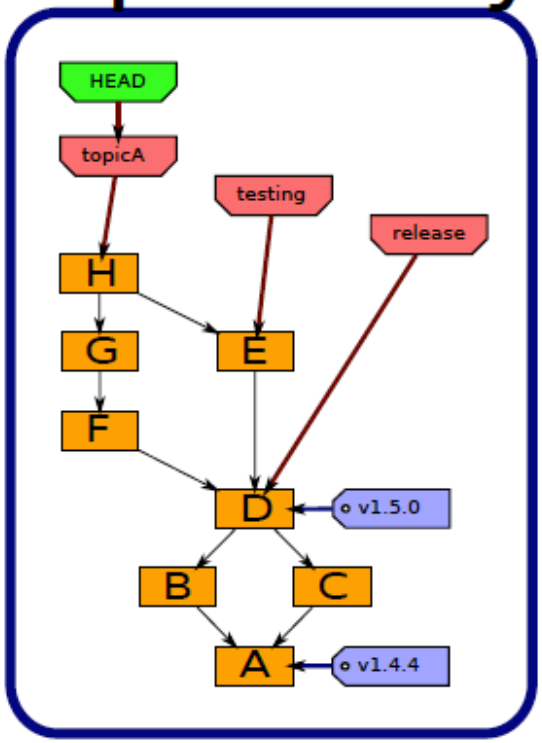
- "staging area"
- what is to be committed



http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

History

repository

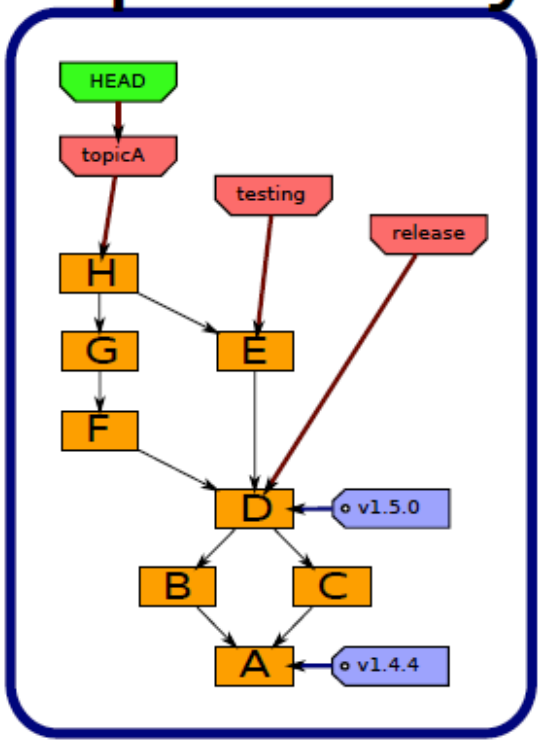


A database, stored in directory ".git".

http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Staging area

repository



index

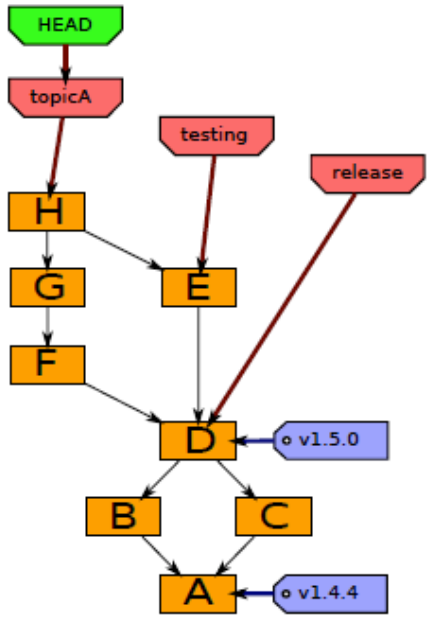
```
100644 20b024 0   bar
100644 1d52a6 0   baz
100644 20b024 0   sub/fi
100644 43dbe0 0   sub/foo
```

Also stored in directory ".git".

http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Files you edit

repository



index

```
100644 20b024 0   bar
100644 1d52a6 0   baz
100644 20b024 0   sub/fi
100644 43dbe0 0   sub/foo
```

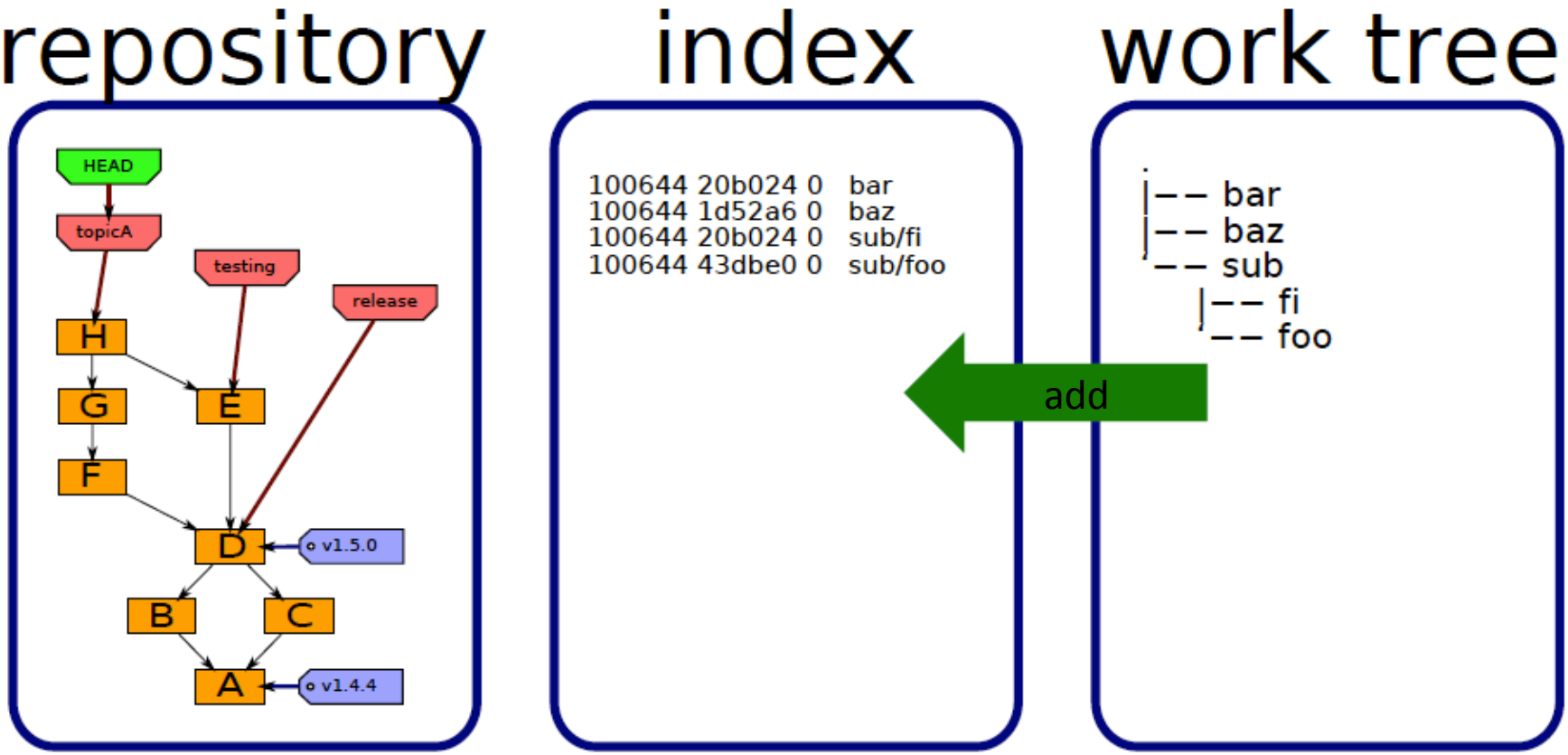
work tree

```
.
|-- bar
|-- baz
`-- sub
    |-- fi
    `-- foo
```

Stored in the directory containing directory ".git".

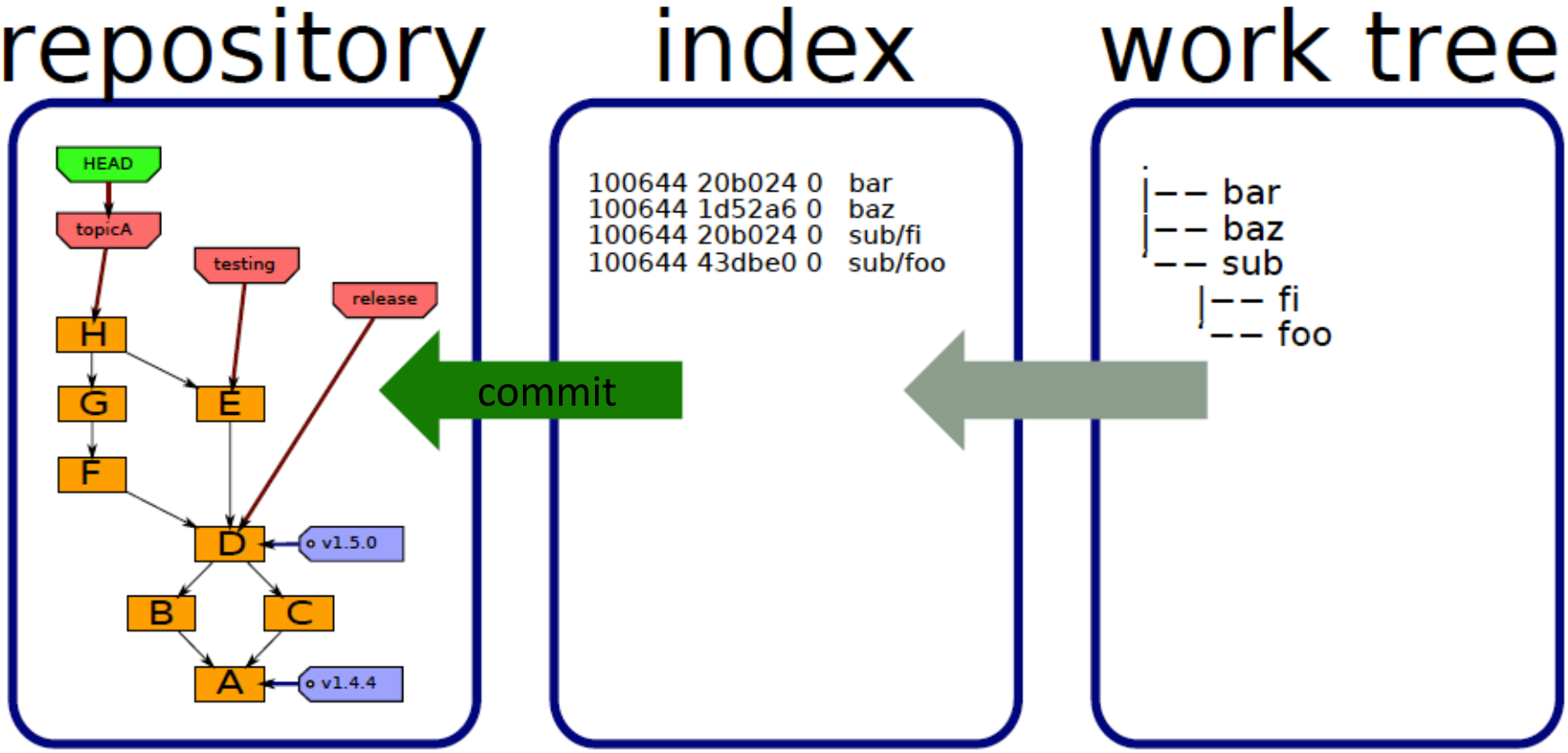
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Staging



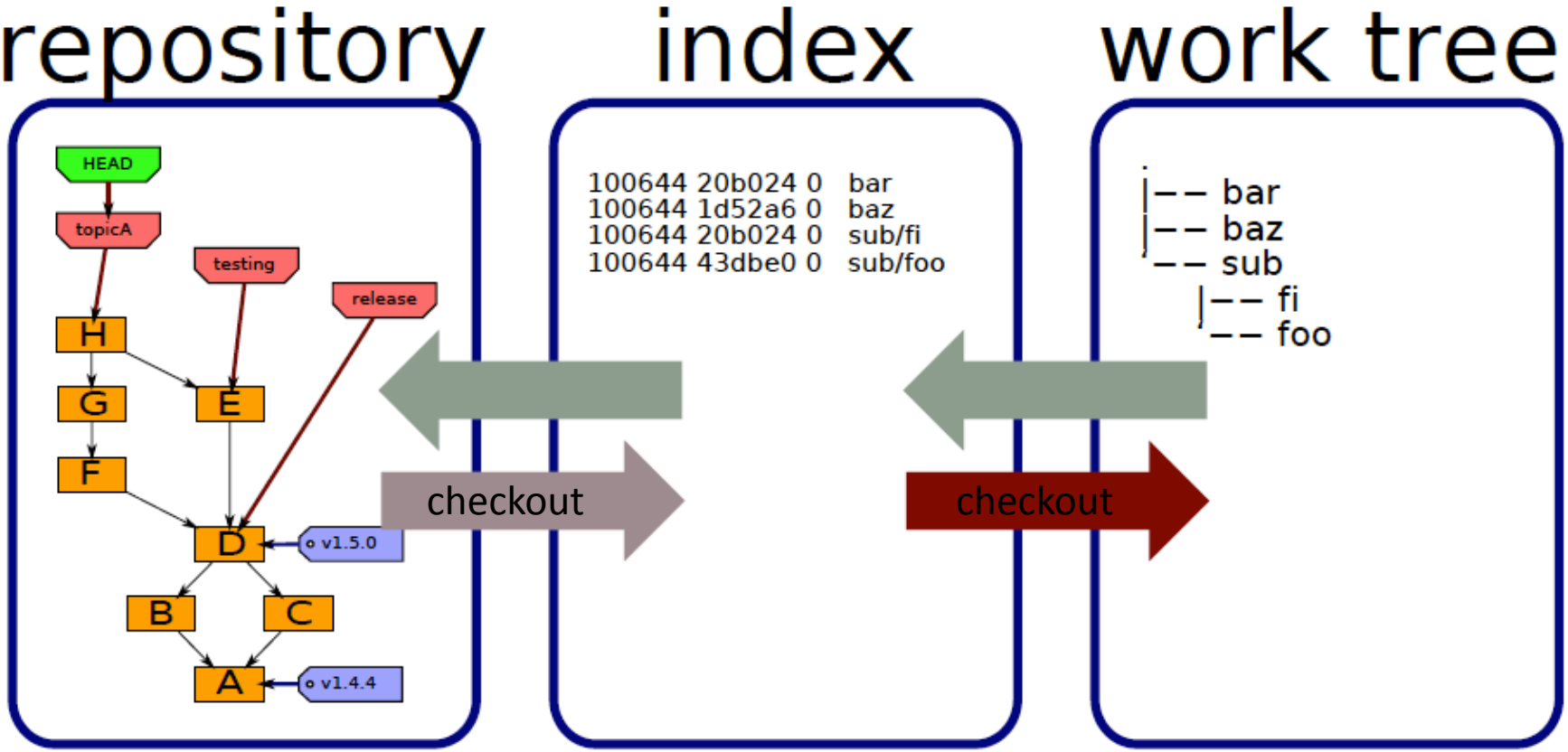
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Committing



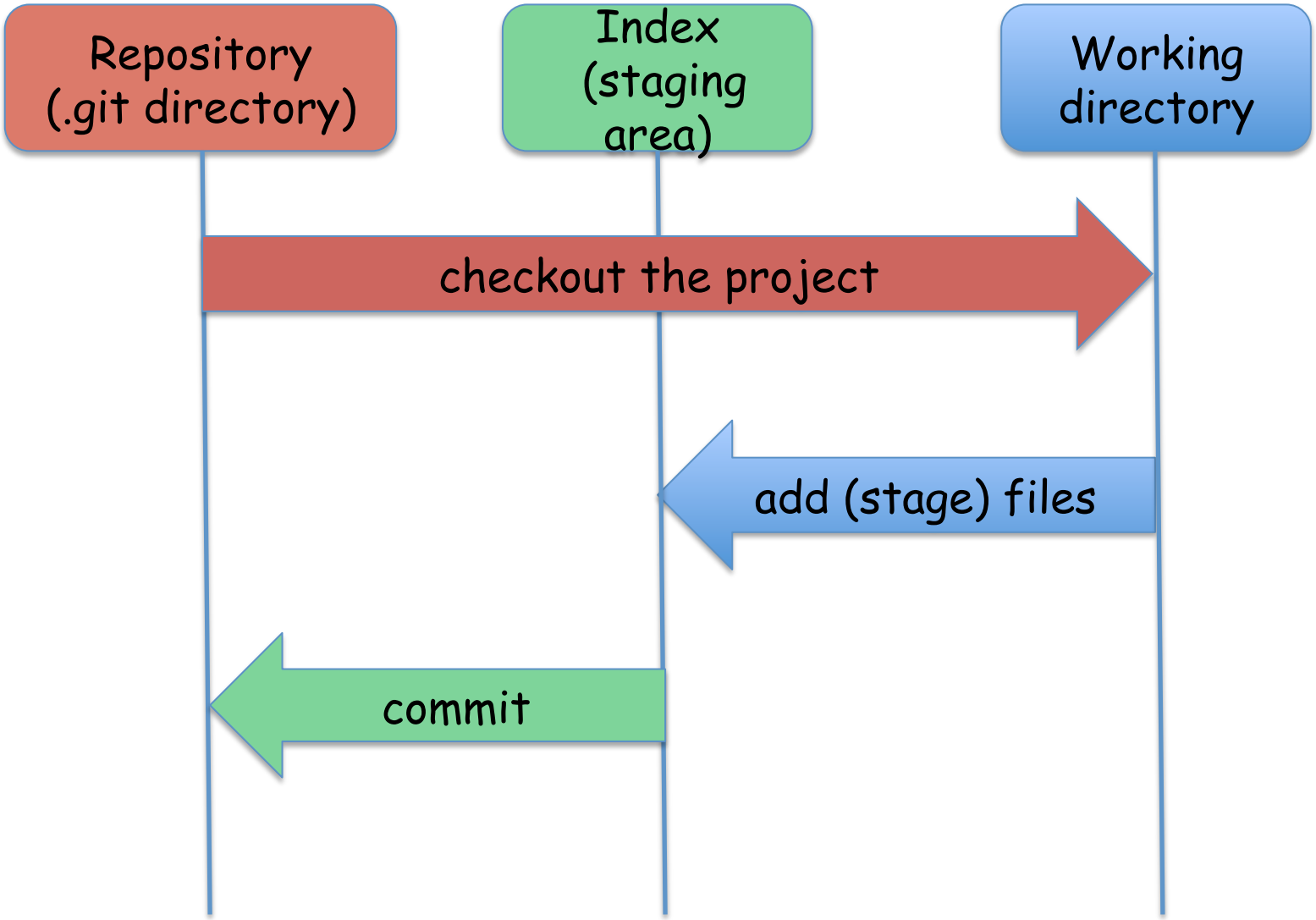
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Checking out



http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

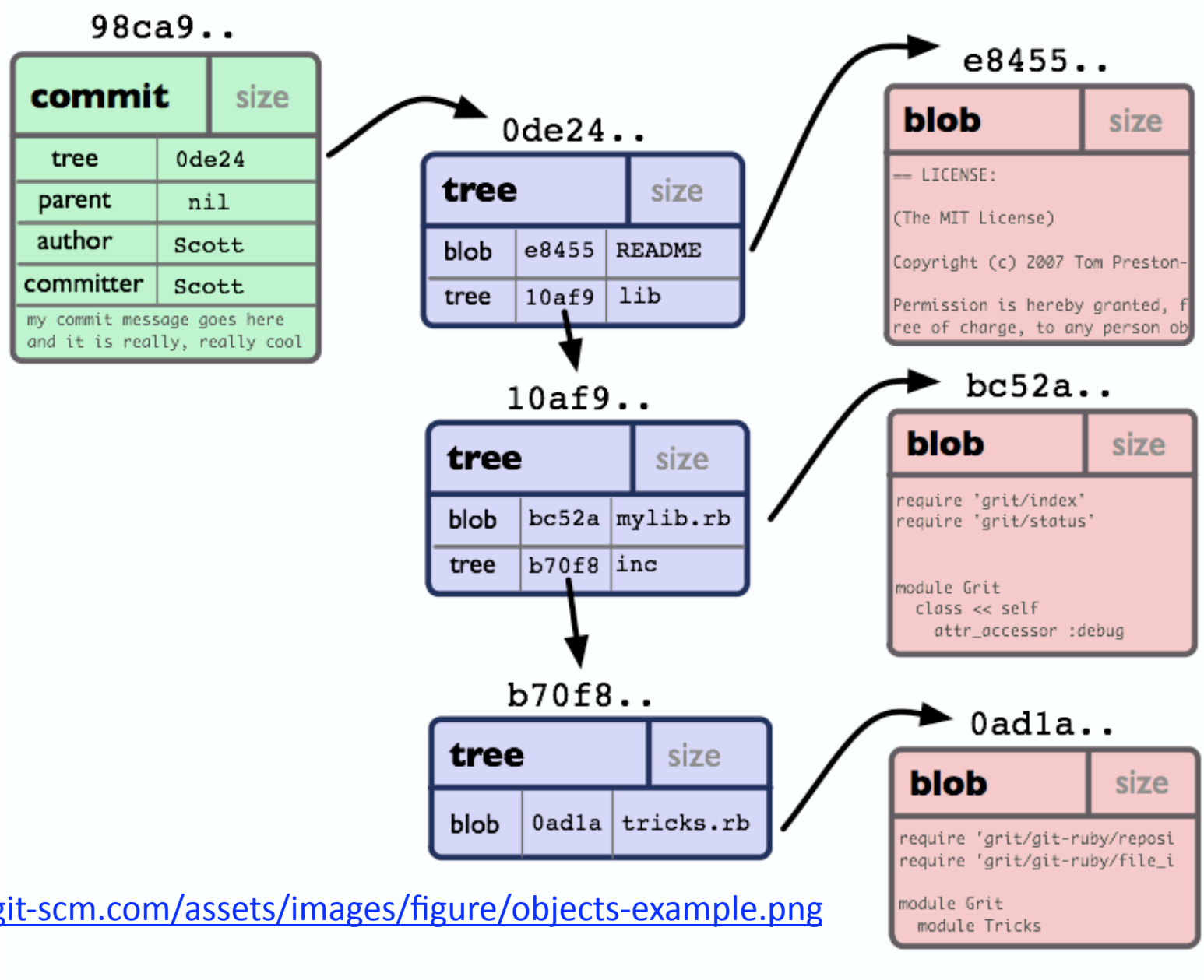
Local Operations



Object types

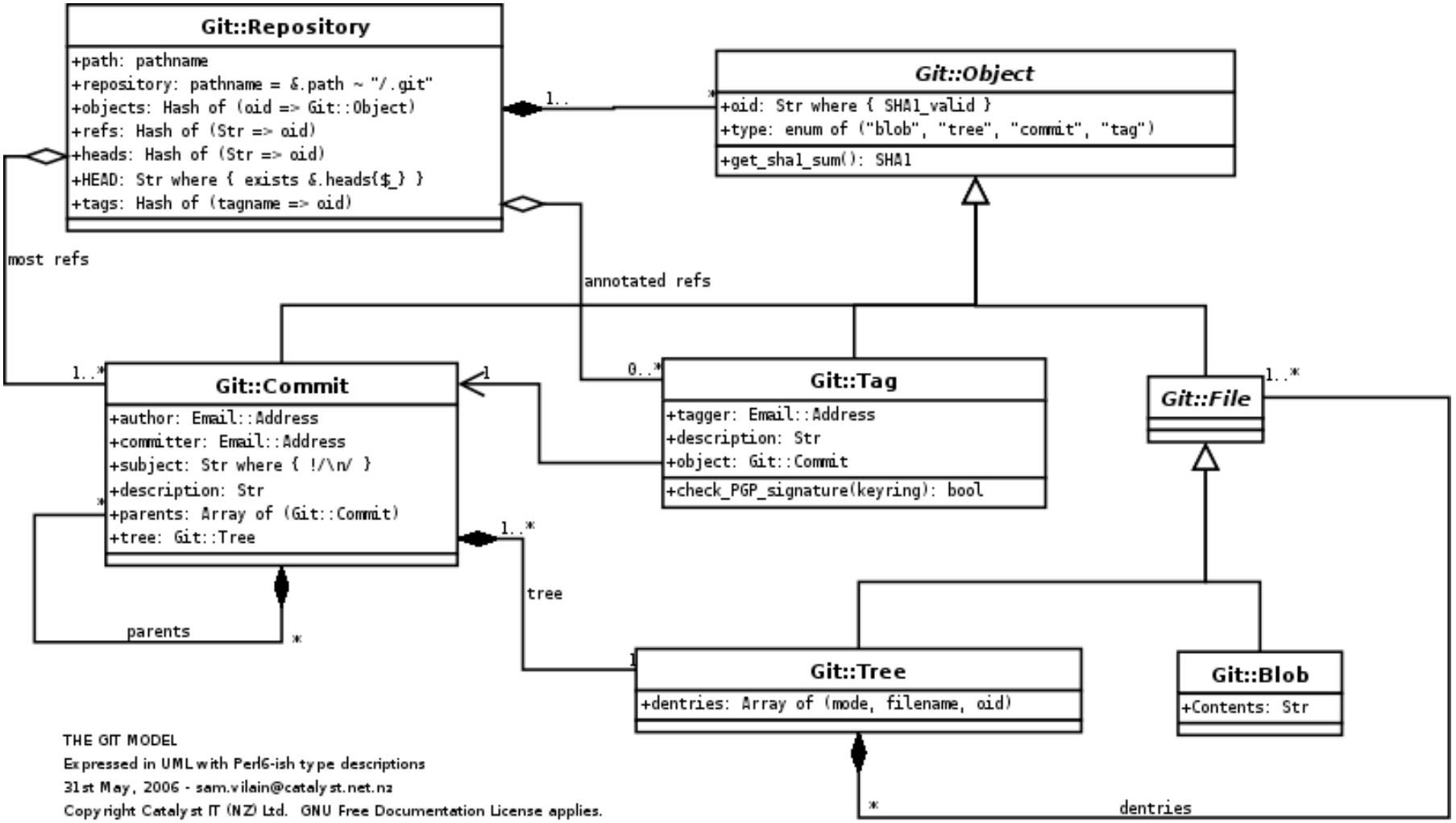
- Blobs
- Trees
- Commits
- Tags

Git Object Model



http://book.git-scm.com/assets/images/figure/objects-example.png

As UML class diagram



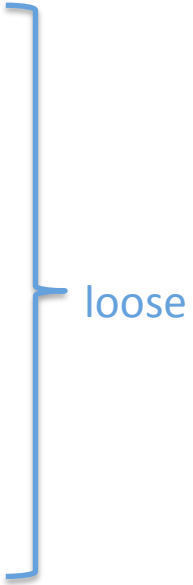
http://utsl.gen.nz/talks/git-svn/git-model.png

Repository

```
.git
|-- HEAD                current checkout reference
|-- config              repo private config
|-- description         repo description
|-- hooks
|   `-- ...             hooking scripts
|-- index               changes to commit
|-- info
|   |-- exclude         repo private
|   `-- refs            refs?
|-- logs
|   `-- ...             "reflog" data
|-- objects
|   |-- XX
|   |   `-- ...         loose objects
|   |-- info
|   |   `-- packs       info about packs
|   `-- pack
|       `-- ...         packs and indexes
`-- refs
    |-- heads
    |   `-- master      master branch
    `-- tags
        `-- ...         tags
```

.git/objects

```
|-- 23
|   '-- d4bd826aba9e29aaace9411cc175b784edc399
|-- 76
|   '-- 49f82d40a98b1ba59057798e47aab2a99a11d3
|-- c4
|   '-- aaefaa8a48ad4ad379dc1002b78f1a3e4ceabc
|-- e7
|   '-- 4be61128eef713459ca4e32398d689fe80864e
|-- info
|   '-- packs
'-- pack
    |-- pack-b7b026b1a0b0f193db9dea0b0d7367d25d3a68cc.idx
    '-- pack-b7b026b1a0b0f193db9dea0b0d7367d25d3a68cc.pack
```

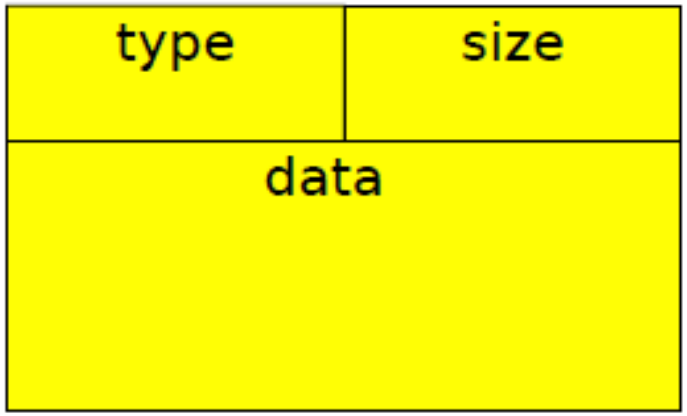


Some other repository files

- .git/config
- .git/description – used by gitweb
- .git/info/exclude – files to ignore
- …

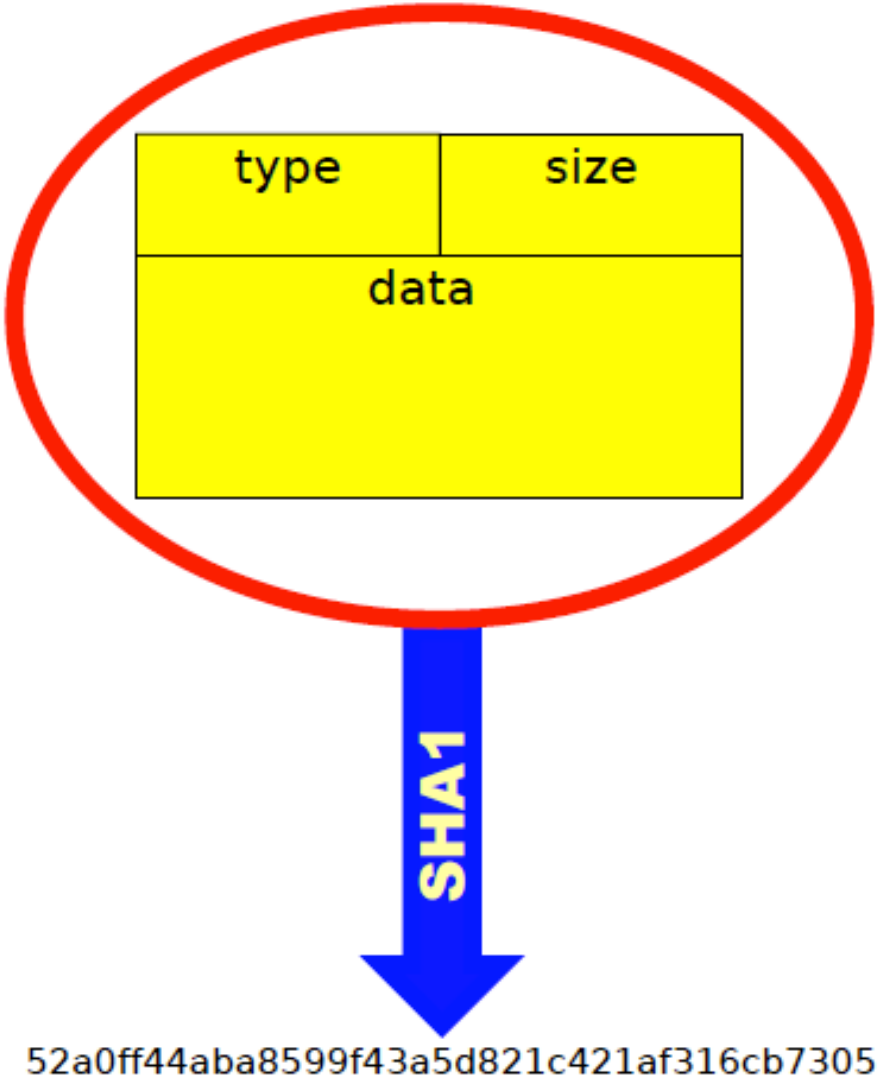
Repository object naming convention

"content addressable" (hashed)



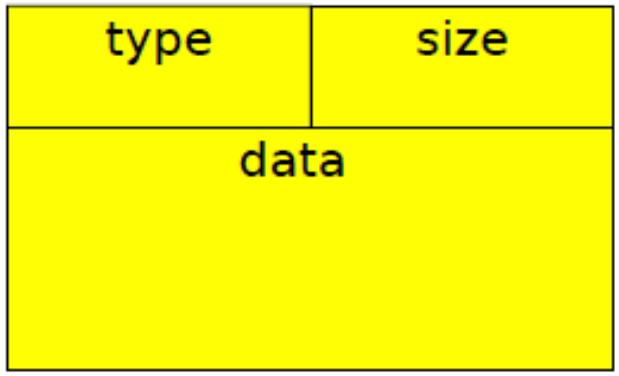
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

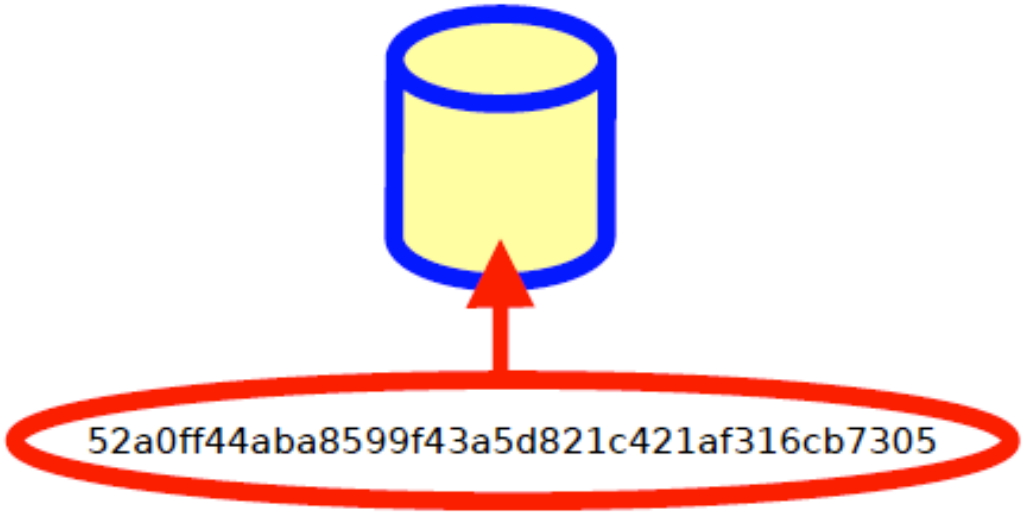
Data values determine hash



http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

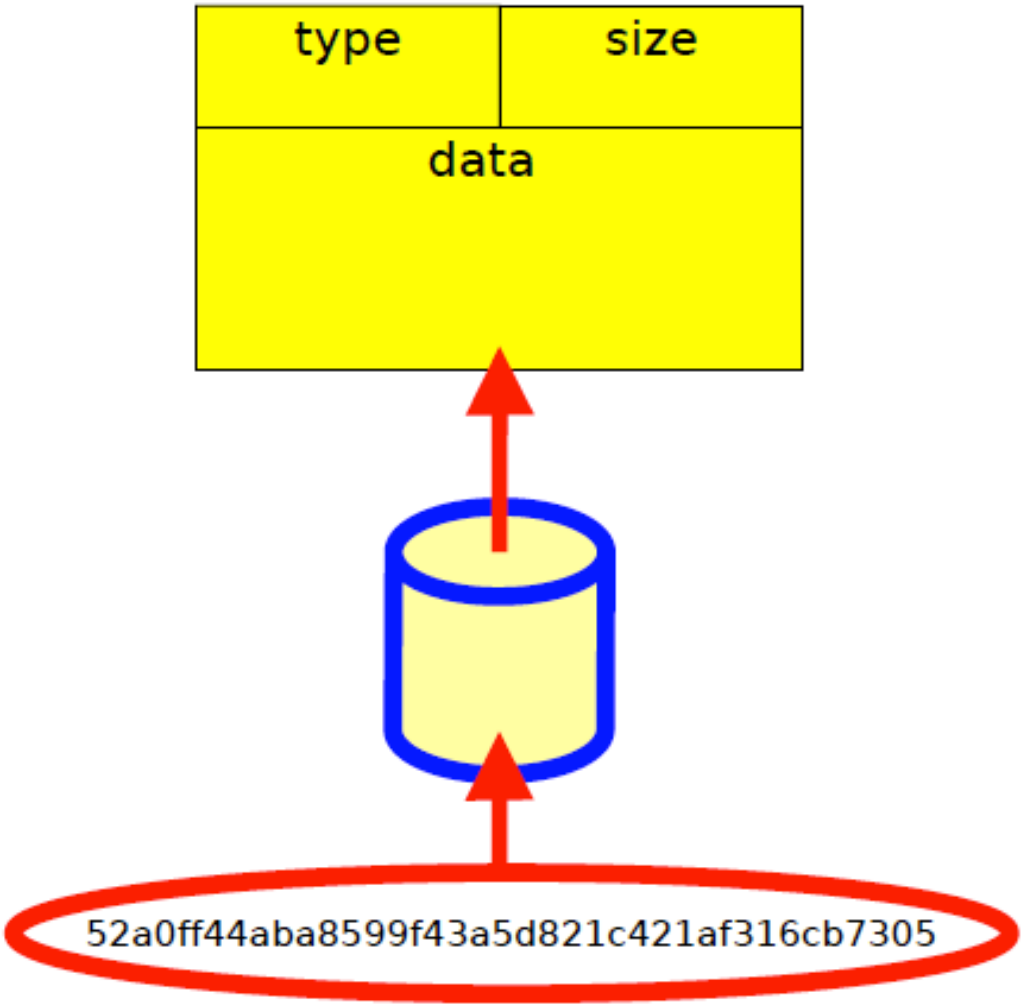
Hash value is filename





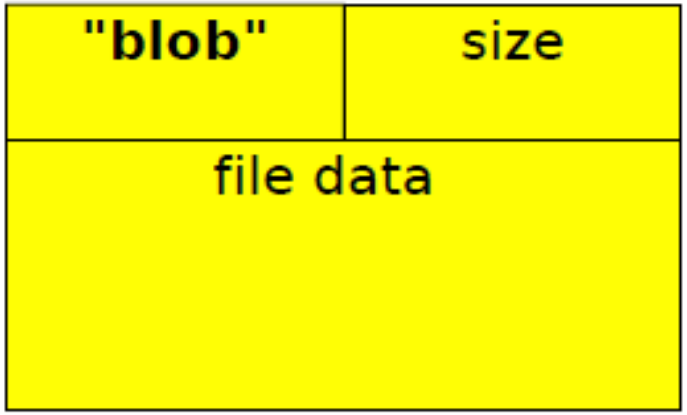
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

File contains data



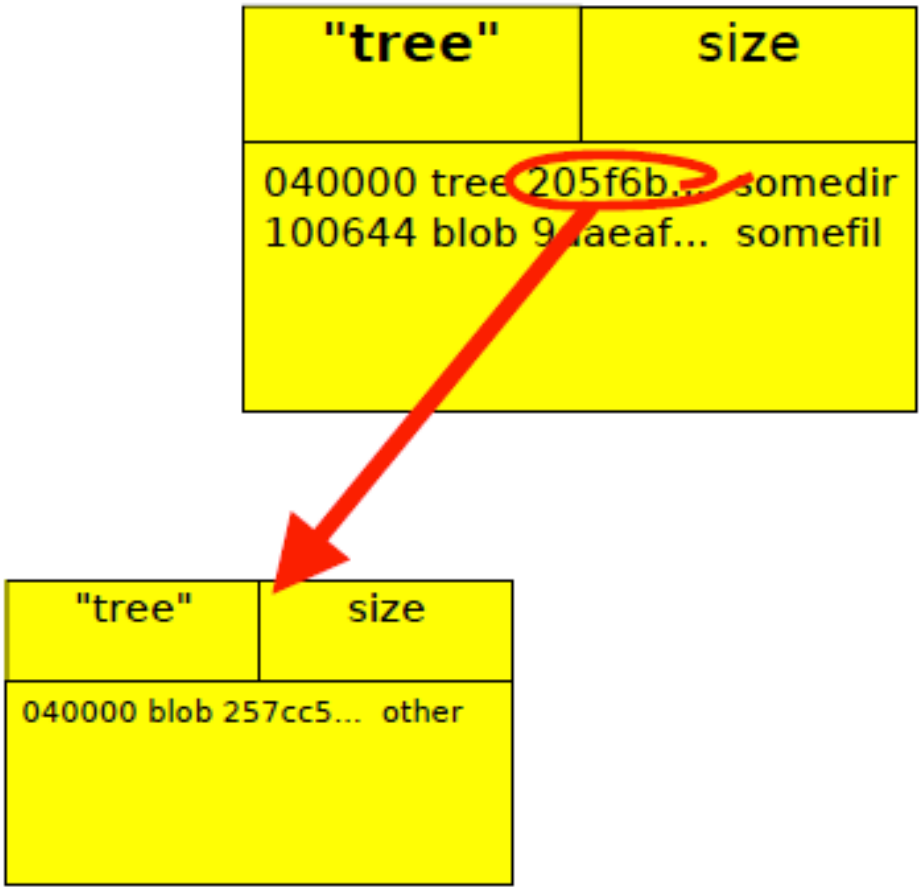
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Blobs



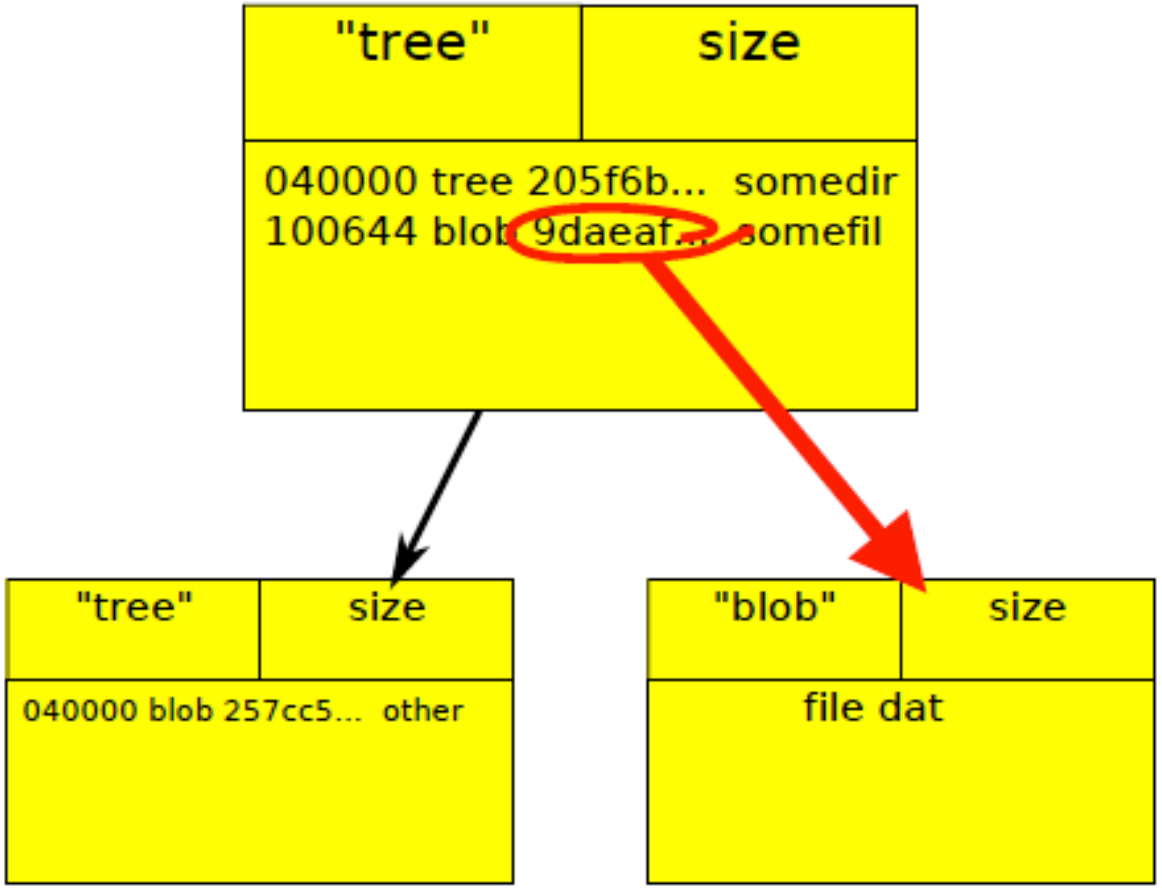
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Trees



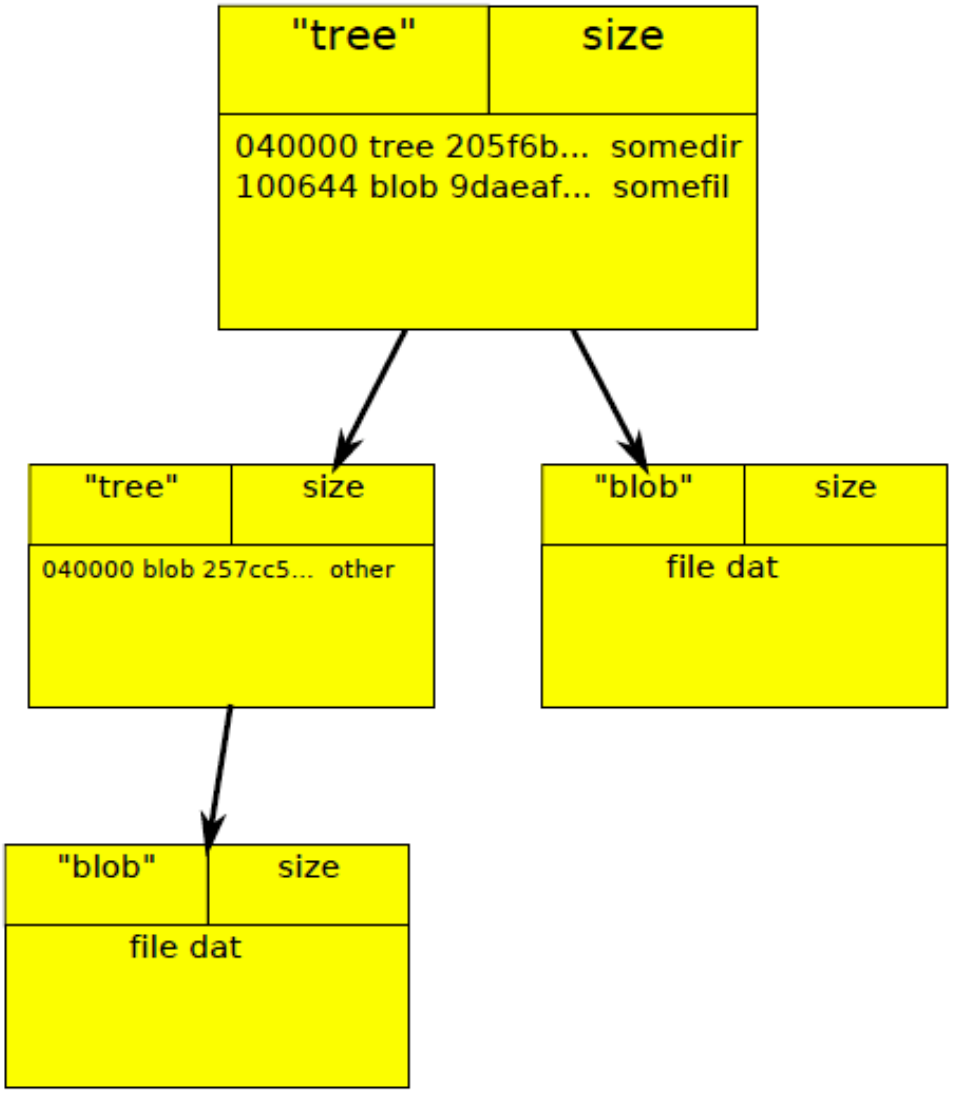
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Trees



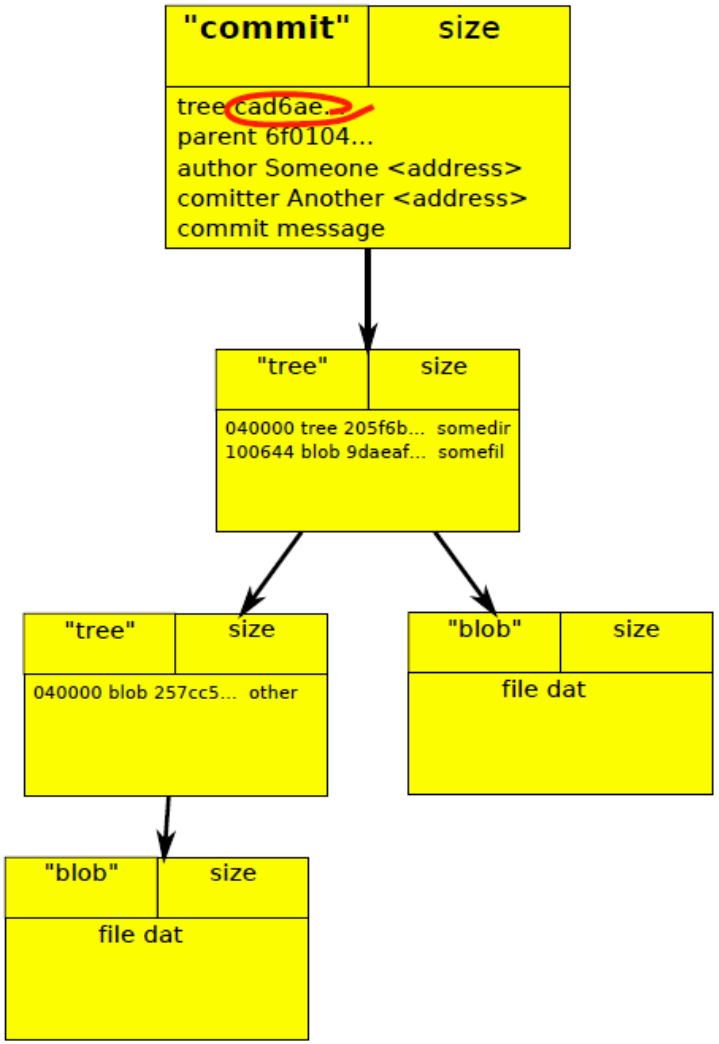
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Trees



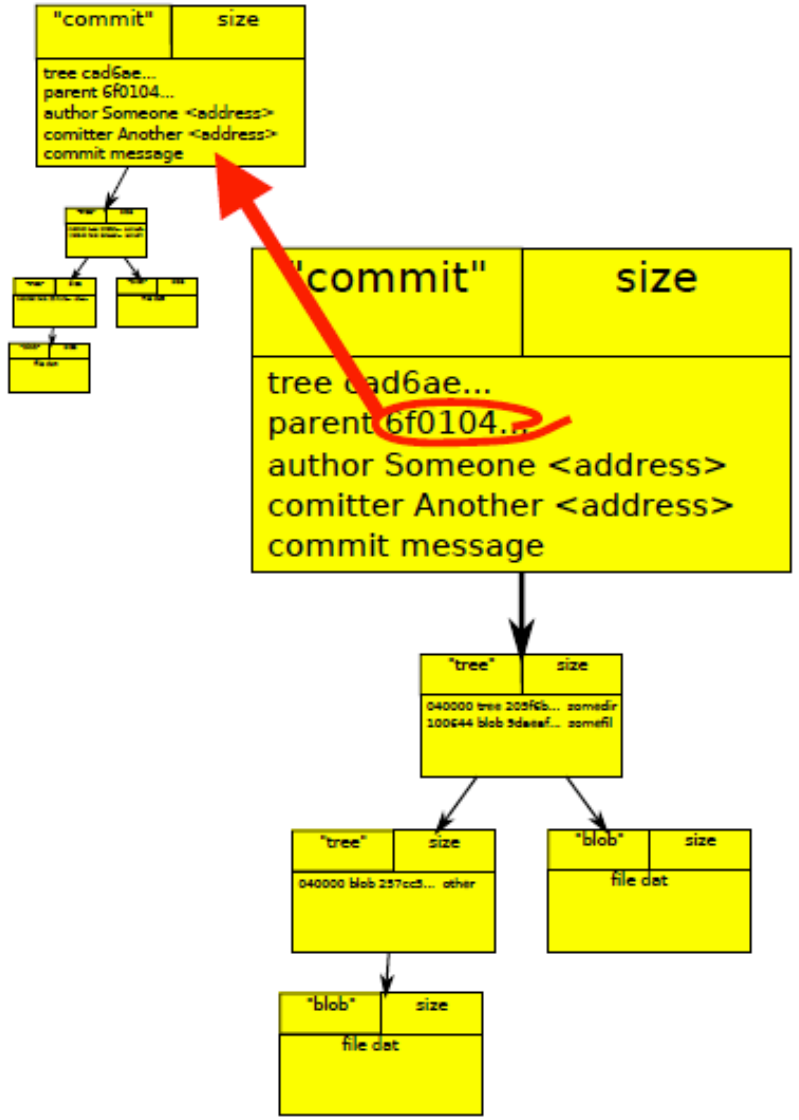
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Commits



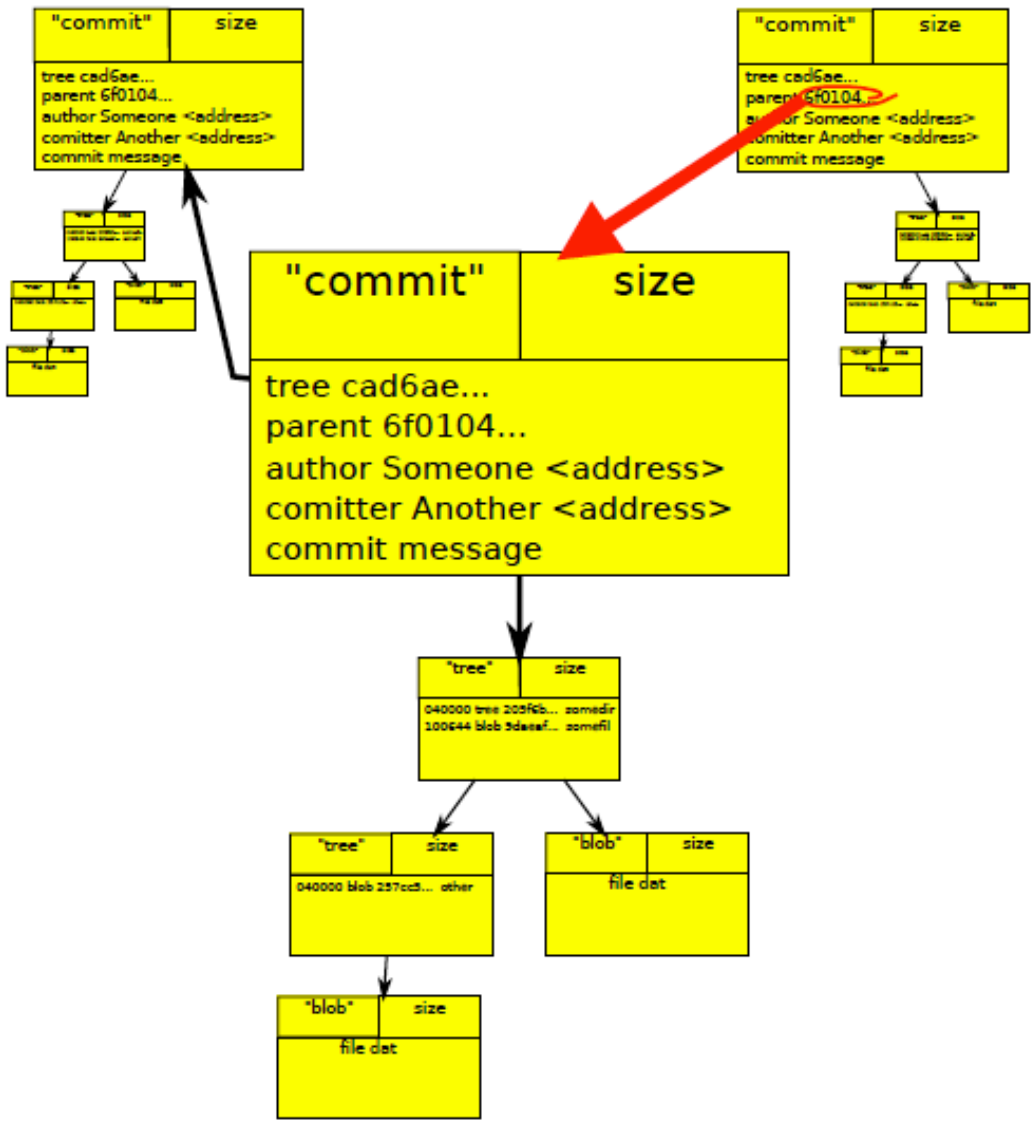
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Commits



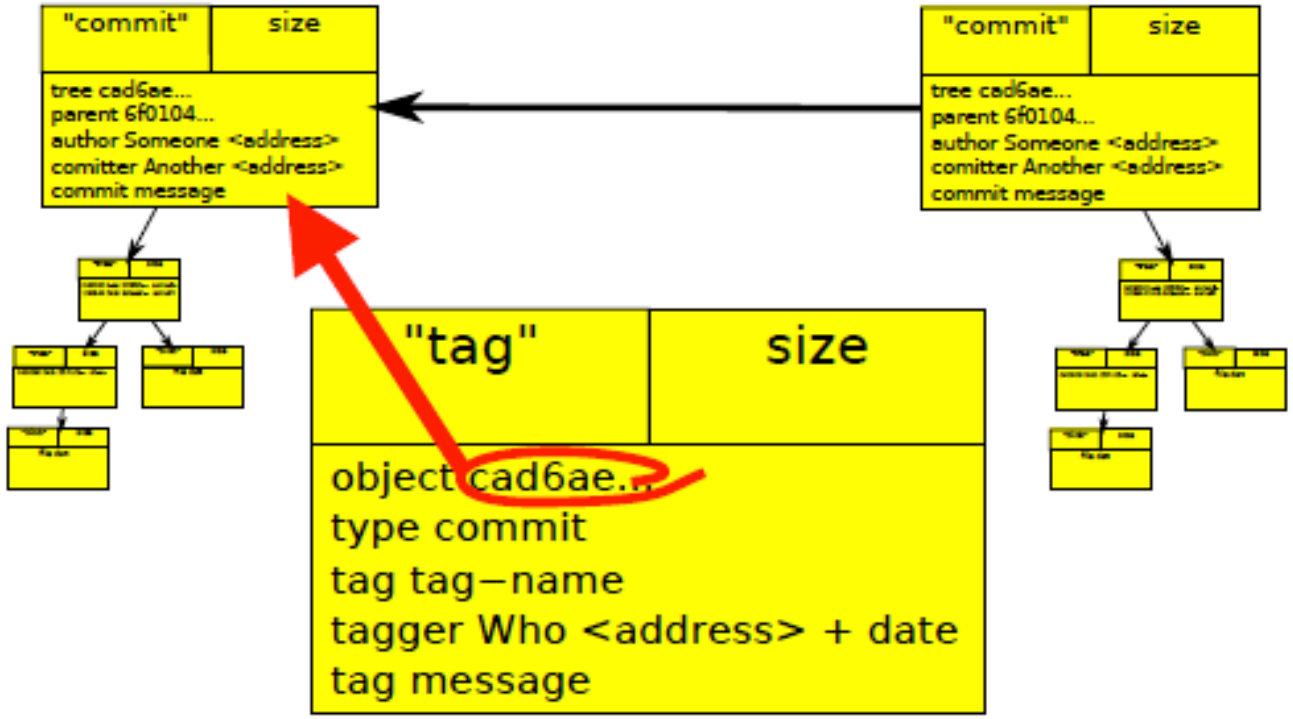
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Commits



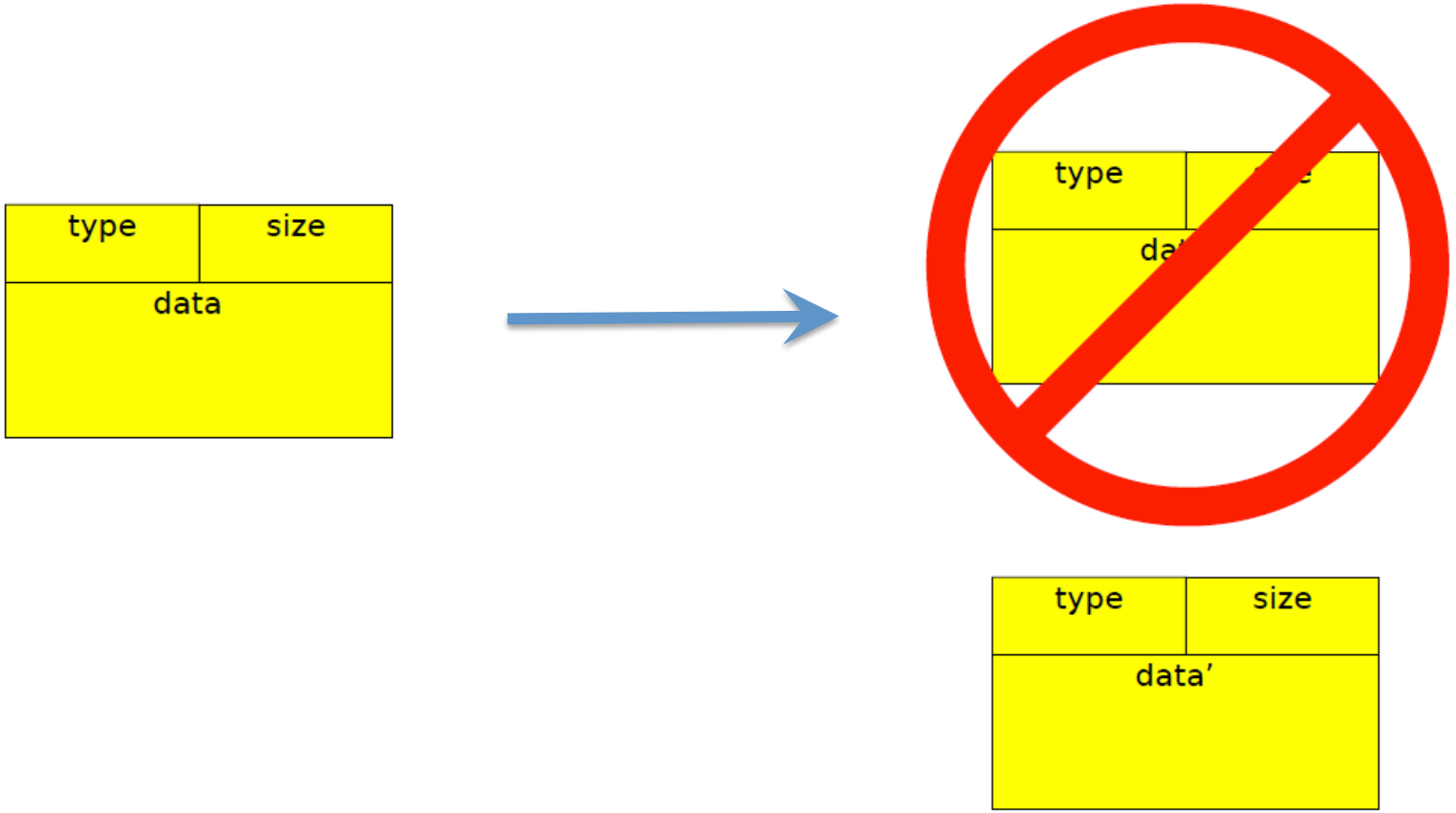
http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Commits



http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

Objects are immutable



http://edgyu.excess.org/git-tutorial/2008-07-09/intro-to-git.pdf

References

- http://book.git-scm.com/index.html
- http://excess.org/article/2008/07/ogre-git-tutorial/
- http://www-cs-students.stanford.edu/~blynn/gitmagic/
- http://progit.org/book/
- http://www.geekherocomic.com/2009/01/26/who-needs-git/