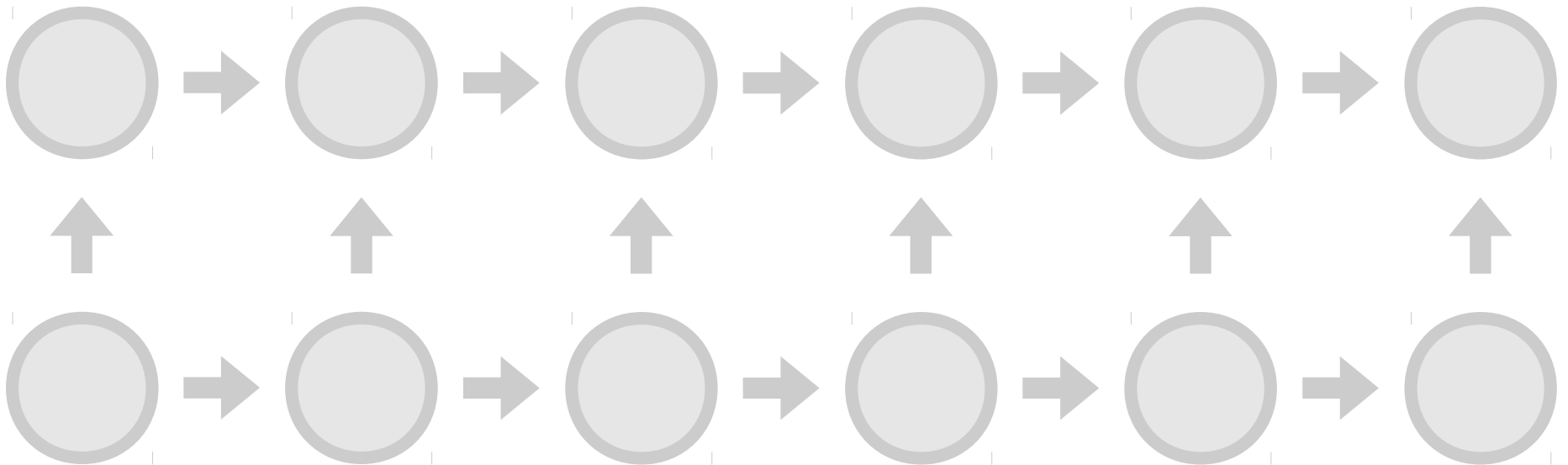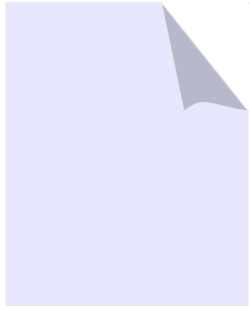# Version Control with Git

Claudius Coenen

What We're *Not* Talking About

- Conceived by Linus Torvalds around 2005

- Distributed Version Control vs. Central Version Control

- Why git is better than <insert favourite vcs>

- Extended Team Workflows

- Internal Workings

Why Version Control?

welcome.html

Why Version Control?

- So, what is the newest version of this file?

- Who changed the title?

- Yesterday, it was working just fine.

- Did you get my mail?

- Who's working on XYZ?

- Is this bugfix already included somewhere?

- Who is working on this project, anyway?

welcome.html

welcome_2.html

welcome_fina

welcome_2.html

e_final.h

welcome2.html

welcome_final2-bernd.html

welcome_final2-berndwelcome2.html

welcome_final2-bernd.html

welcome.html

What *is* Version Control?

NOT Version Control

- Emailing files
- Central FTP / Fileshare
- Periodical Backup of your project directory
- Dropbox

Version Control

- CVS (old)
- SVN
- Mercurial
- Bazaar
- Bitkeeper
- Git
- ...

„Better than nothing"
Award in bronze

„Better than nothing"
Award in silver

Safe-Keeping of Versions

- When you check something into version control, it will remember this exact version for you

- You can return to a previous version

- You get a fancy timeline of your project

06.03.2012 14:10

10.10.2012 14:12

Yesterday, 10:04

Today
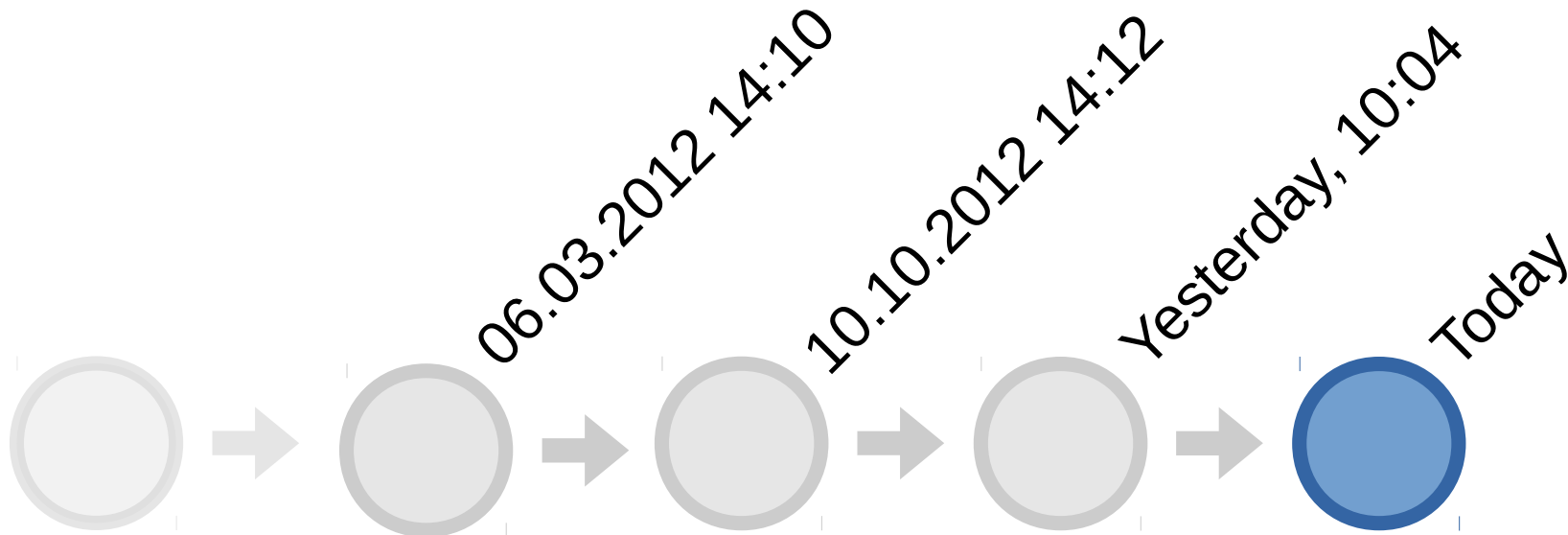
Access Control

- Usually on a per-project basis
- Who can read and/or write a „repository"
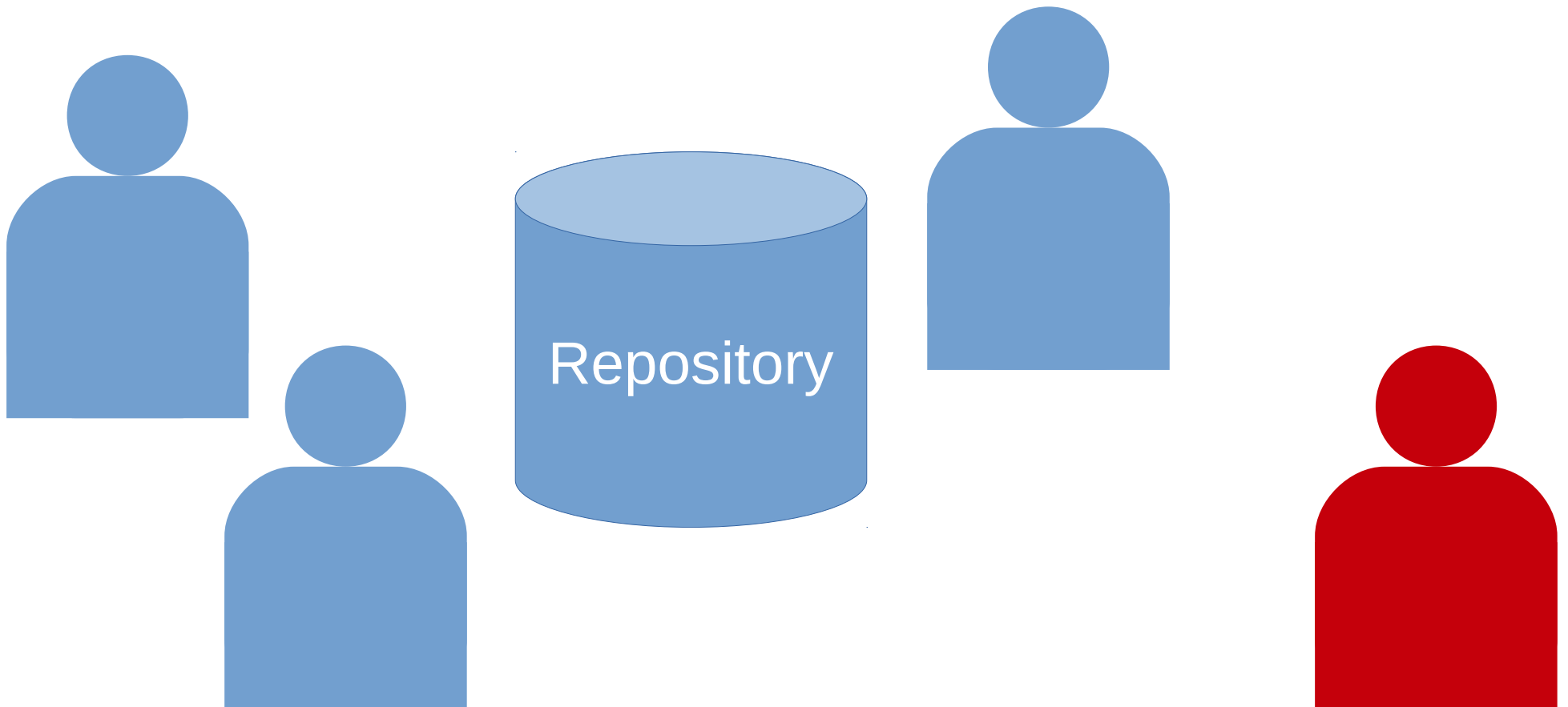
Conflict Resolution

- Files edited by different people can usually be resolved automatically.

welcome.html
(Alice)

welcome.html
(Bob)

welcome.html
(Both changes)
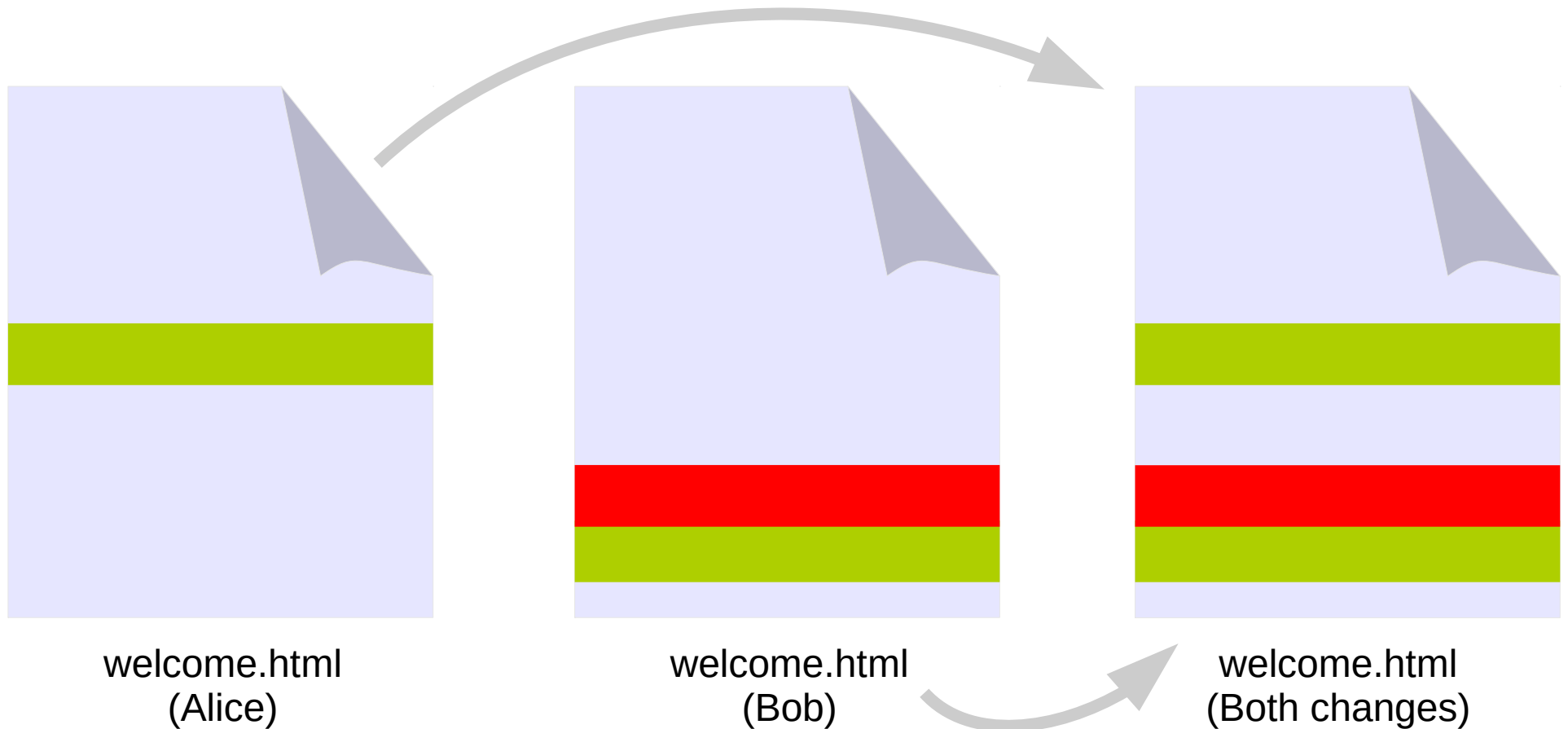
The „Commit"

- An item in the history
- Think of this as a snapshot of your project

Commit-Hash

```
commit f303d887ebf263b466958ca7c83aebb9385936f0
Author: Claudius Coenen <coenen@meso.net>
Date:    Sat Feb 15 02:08:44 2014 +0100

    Ticket #2342: fixing file upload to work with
    animated gif
```
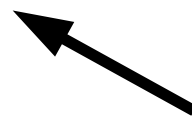
Commit-Message

Full Workflow

git pull

- DO STUFF
  change files the way you normally would

- PREPARE
  git add the changes, so git knows about them

- COMMIT
  git commit, preferably with a nice message

git push

think: „committing to memory"
or „committment"

Branching/Merging
And The History
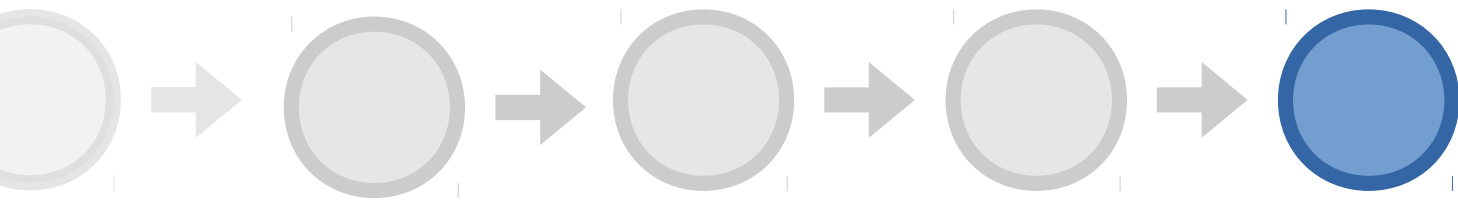
Branching/Merging
And The History

#§%$!~??
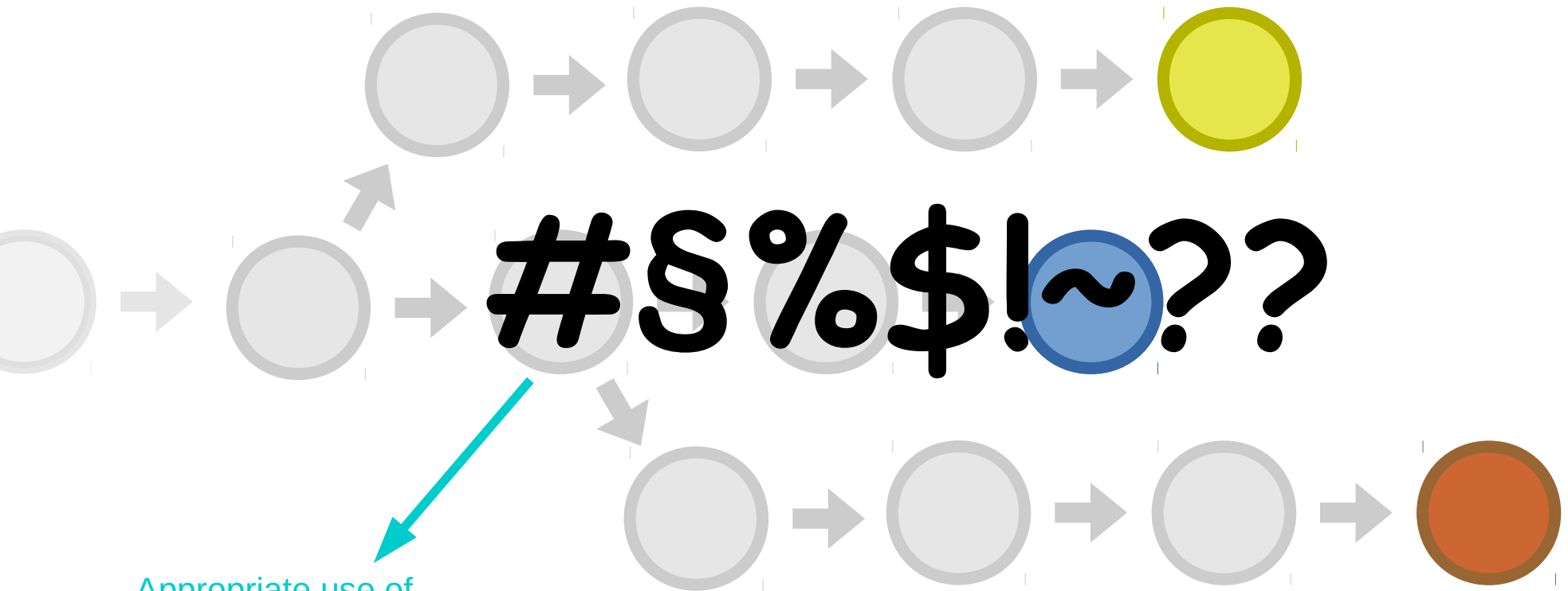
Appropriate use of
Comic Sans!

Branching/Merging
And The History
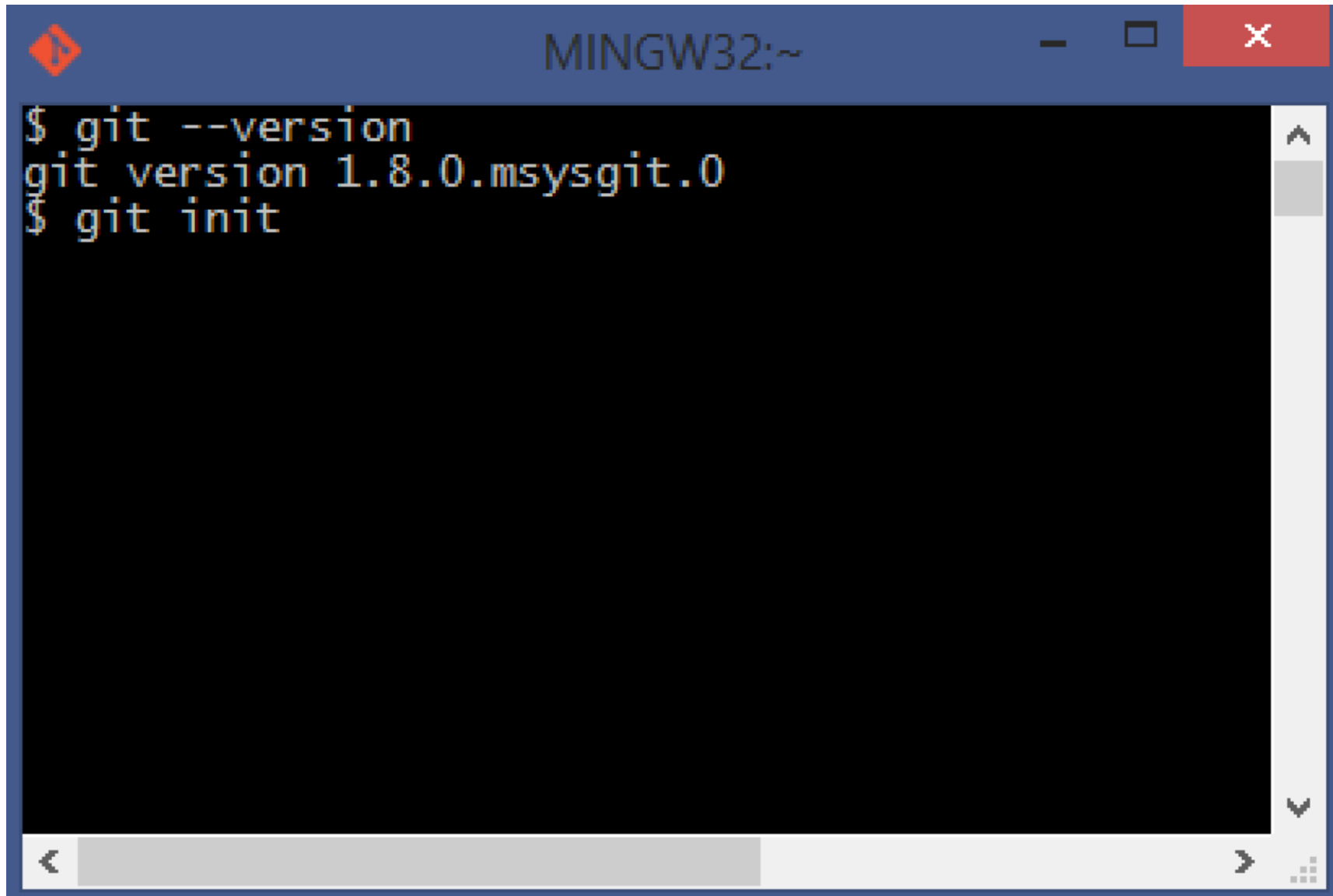
Branch starts here

Merge
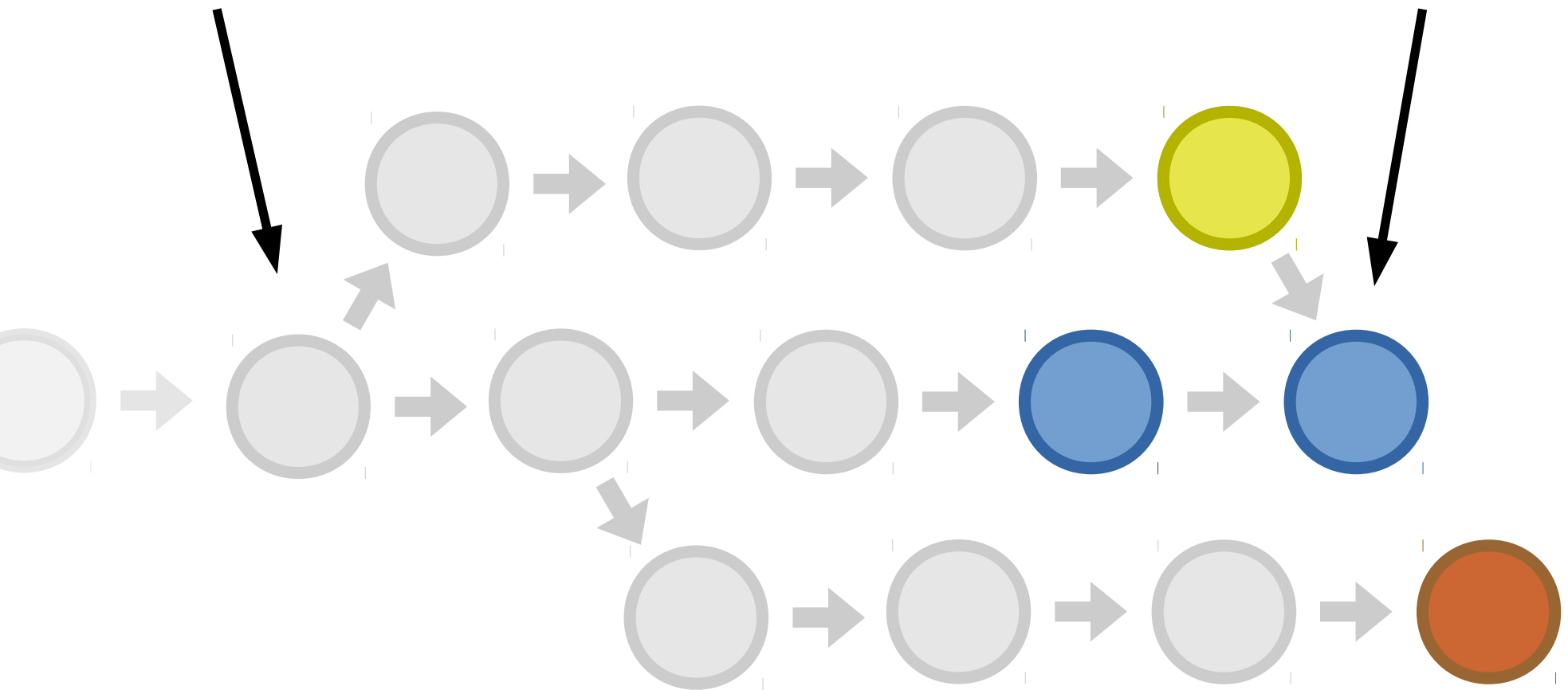
Usually no problem

Git Command Line Interface

Github App

Atlassian SourceTree

References

Command Cheat Sheet (Infrequent Commands)

- git config --global user.name "Jane Doe"
- git config --global user.email "j.d@test.com"
- git config --global color.ui true
- git config --system core.editor <editor>

- git init   (creates a new repository)
- git clone <repository url>   („clones" existing)

Command Cheat Sheet
(Frequent Commands)

- git status   (what's up on my drive)
- git log   (what happened so far)
- git add   (prepare these files for commit)
- git commit   (commit the added files)
- git checkout   (return to a previous version)
- git reset   (reset the git index)

These Commands are Local!

Command Cheat Sheet
(Used often)

- git fetch   (just download)

- git pull   (download AND merge)

- git push   (send my changes out)

These Commands
go over the network*
*) simplification

Where to Start

- Baby steps in your browser
  - http://try.github.io
- Download git for your system
  - http://git-scm.com/downloads
- Maybe add a Git client that suits your taste
  - At least *try* the commandline. Git in its purest form.
  - gitk, GitHub-App, SourceTree, GitTower, TortoiseGit
- Integrate git into your IDE / Editor
  - Eclipse, Sublime, WebStorm, RubyMine, NetBeans

Hand-Picked Resources

- Git has extensive documentation
    - http://git-scm.com/documentation
- There's a lot of Tutorials out there
    - http://marklodato.github.io/visual-git-guide/index-en.html
    - https://www.kernel.org/pub//software/scm/git/docs/everyday.html
    - http://www.teehanlax.com/blog/github-fundamentals/
    - http://mislav.uniqpath.com/2010/07/git-tips/
    - http://nvie.com/posts/a-successful-git-branching-model/
- In-Depth Info
    - http://git-scm.com/book
- Cannon fodder for lengthy debates with your fellow geek
    - http://thkoch2001.github.io/whygitisbetter/