

INTRODUCTION TO GIT

A LECTURE ARRANGED BY LUDD AND ARCTIC GROUP

Anders Engström

7 April 2016



OVERVIEW

INTRODUCTION

- In a world without version control

- Types of version control

- This is git

GETTING STARTED

- Preparations

- Your own repository

- First time with a repository

- Connect more users

USING GIT

- Everyday use

- Tags

- Branching

- Undo

- A closer look at the history

- Ignoring files

- Git on Windows

CONCLUSION

- Summary

- Bibliography and links

- Questions and comments

OVERVIEW

INTRODUCTION

In a world without version control

Types of version control

This is git

GETTING STARTED

Preparations

Your own repository

First time with a repository

Connect more users

USING GIT

Everyday use

Tags

Branching

Undo

A closer look at the history

Ignoring files

Git on Windows

CONCLUSION

Summary

Bibliography and links

Questions and comments

IN A WORLD WITHOUT VERSION CONTROL

HOW DO YOU WORK IN A GROUP?

- ▶ Shared folder.
- ▶ Mail files back and fourth. (All files or only changed ones?)
- ▶ Instant messaging.

HOW DO YOU UNDO CHANGES?

- ▶ Rewrite parts of the code so that it looks the same as before.
- ▶ Create a copy in advance according to a naming convention.
- ▶ Search through old mails.

CAN YOU COME UP WITH ANY MORE WAYS?

TYPES OF VERSION CONTROL

HOW DO YOU HANDLE MULTIPLE USERS?

- ▶ File locks
- ▶ Merge

WHERE IS THE INFORMATION STORED?

- ▶ Centralized
- ▶ Distributed

THIS IS GIT

GIT IS:

- ▶ A fast distributed version control system.
- ▶ Written for the Linux kernel (by Linus Torvalds et al.)
- ▶ Lots of other projects also use it.
- ▶ Git is open source, GPL etc.

WE WILL LOOK AT:

- ▶ Mostly the official command-line client. (Also available for Windows.)
- ▶ A little bit of a graphical client. (The concepts are the same.)

OVERVIEW

INTRODUCTION

In a world without version control

Types of version control

This is git

GETTING STARTED

Preparations

Your own repository

First time with a repository

Connect more users

USING GIT

Everyday use

Tags

Branching

Undo

A closer look at the history

Ignoring files

Git on Windows

CONCLUSION

Summary

Bibliography and links

Questions and comments

PREPARATIONS

WHO ARE YOU?

```
1 git config --global user.name "Anders Engström"  
2 git config --global user.email ankan@ludd.ltu.se
```

CREATE A COMMON STORAGE AREA

- ▶ Many sites offer free hosting. (Github, Bitbucket, Ludd)
Create an account and a repository there.
- ▶ Or create your own on a server. (All you need is a directory all users can access.)

YOUR OWN REPOSITORY

CREATE YOUR OWN REPOSITORY (ON A SERVER):

```
1 git init --bare --shared test.git
2 groupadd testproject
3 chgrp -R testproject test.git
4 chmod -R 770 test.git
```

TO BEGIN PUTTING STUFF IN THERE

PREPARE A SIMPLE REPOSITORY TO UPLOAD (ON A CLIENT MACHINE.):

```
1 git init test
2 cd test
3 touch README
4 git add README
5 git commit -m 'first commit'
```

CONNECT WITH THE SERVER FOR THE FIRST TIME

BITBUCKET:

```
1 git remote add origin https://ankaan@bitbucket.org/  
   ankaan/test.git
```

GITHUB:

```
1 git remote add origin git@github.com:ankaan/test.git
```

YOUR OWN SERVER:

```
1 git remote add origin ankan@localhost:test.git
```

UPLOAD:

```
1 git push -u origin master
```

CONNECT MORE USERS

TO CONNECT MORE USERS BEYOND THE FIRST, USE:

```
1 git clone ankan@localhost:test.git
```

OVERVIEW

INTRODUCTION

In a world without version control

Types of version control

This is git

GETTING STARTED

Preparations

Your own repository

First time with a repository

Connect more users

USING GIT

Everyday use

Tags

Branching

Undo

A closer look at the history

Ignoring files

Git on Windows

CONCLUSION

Summary

Bibliography and links

Questions and comments

WHICH COMMANDS TO YOU ACTUALLY NEED TO LEARN?

ADD Mark file for saving.

RESET Unmark file for saving.

COMMIT Save changes to the log.

STATUS How does the git-repo look? What files have been changed?

PUSH Send all committed changes to the shared repository.

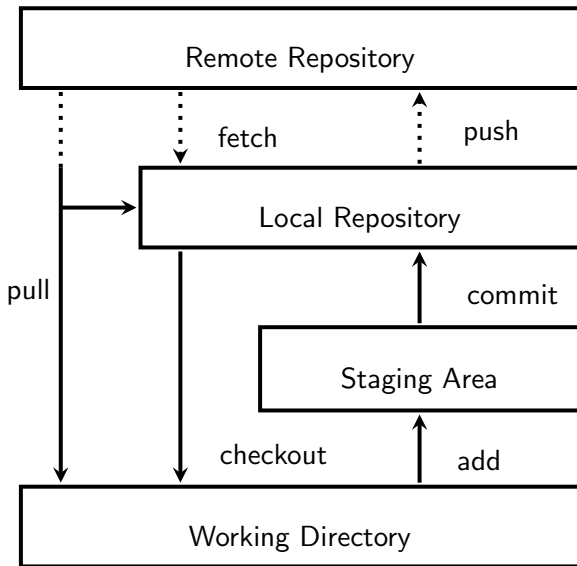
PULL Download all changes from the shared repository.

CHECKOUT Get an old version of a file.

LOG Show a log over all commits made.

HELP Show the help for a certain command.

HOW IS DATA STORED IN GIT?



DEMONSTRATION OF THE BASIC CONCEPTS.

- ▶ Change files.
- ▶ Commit.
- ▶ Handle conflicts.
- ▶ Look in the log.

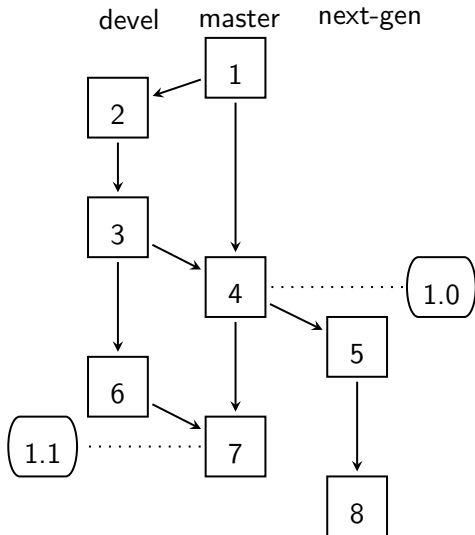
WHAT IS A TAG?

- ▶ A named bookmark that points to a specific commit.

COMMANDS TO HANDLE TAGS:

```
1 git tag <name>
2 git tag
3 git push --tags
```

BRANCHES AND COMMITS



HOW TO USE BRANCHES

COMMANDS:

BRANCH List or create a new branch. (-r to list remote branches.)

CHECKOUT Is used to change branches.

MERGE Bring in changes from another branch.

BRANCHES ARE LOCAL BY DEFAULT:

```
1 # Upload and follow:  
2 git push -u origin <branch>  
3  
4 # Follow a remote branch:  
5 git branch <local_branch> <remote_branch>
```

UNDO

UNDO CAN MEAN DIFFERENT THINGS:

```
1 # While still coding:  
2 git checkout <file >  
3 git checkout 502fcb1 <file >  
4  
5 # Prepared for a commit, but not yet committed:  
6 git reset HEAD <file >  
7  
8 # Already committed:  
9 git revert 502fcb1
```

LEAVE YOUR COMMITTED ERRORS IN THE HISTORY!

A CLOSER LOOK AT THE HISTORY

DIFFERENT WAYS OF LOOKING AT THE HISTORY:

- LOG** Show a log of all commits with their comments.
- SHOW** Show details for a specific commit.
- DIFF** Compare different versions of files.
- BLAME** Show who last touched a certain row.

HOW DO YOU GET GIT TO STOP LISTING CERTAIN FILES?

- ▶ Create a `.gitignore` file and list what should be ignored (wildcards are available.)

WHAT IF I USE WINDOWS?

- ▶ msygit (installs a shell and works just like on Linux.)
- ▶ Git Extensions (a GUI-application.)
- ▶ Built into your IDE (Eclipse, IntelliJ Idea, etc.)
- ▶ There are GUI programs for other operating systems also.

OVERVIEW

INTRODUCTION

In a world without version control

Types of version control

This is git

GETTING STARTED

Preparations

Your own repository

First time with a repository

Connect more users

USING GIT

Everyday use

Tags

Branching

Undo

A closer look at the history

Ignoring files

Git on Windows

CONCLUSION

Summary

Bibliography and links

Questions and comments

WE HAVE SEEN:

- ▶ What you need version control for.
- ▶ What git is.
- ▶ How to use git.
- ▶ Some of the most common problems you can encounter.

BIBLIOGRAPHY AND LINKS

GUIDES:

- ▶ <http://schacon.github.com/git/gittutorial.html>
- ▶ <http://ricroberts.com/articles/getting-to-grips-with-git-part-1-the-basics>

ONLINE HOSTING:

- ▶ <http://buck.ludd.ltu.se>
- ▶ <http://gitlab.com>
- ▶ <http://github.com>
- ▶ <http://bitbucket.org>

MORE INFORMATION:

- ▶ http://en.wikipedia.org/wiki/Revision_control
- ▶ [http://en.wikipedia.org/wiki/Git_\(software\)](http://en.wikipedia.org/wiki/Git_(software))

QUESTIONS AND COMMENTS

?