

Introduction

What is version control?

Distributed

Snapshots

Storage landscape

Using Git

Getting started...

commit

branch

merge

 Merge conflicts

Exploration

Remote repositories

Industrial workflow

Final tips

Lab 3: Version control and collaboration

Comp Sci 1585
Data Structures Lab:
Tools for Computer Scientists

MISSOURI
S&T

Computer Science

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
 - commit
 - branch
 - merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

1 Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

2 Using Git

- Gitting started...
 - commit
 - branch
 - merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

Introduction

What is version control?

Distributed

Snapshots

Storage landscape

Using Git

Gitting started...

commit

branch

merge

 Merge conflicts

Exploration

Remote repositories

Industrial workflow

Final tips

1 Introduction

What is version control?

Distributed

Snapshots

Storage landscape

2 Using Git

Gitting started...

commit

branch

merge

 Merge conflicts

Exploration

Remote repositories

Industrial workflow

Final tips

Introduction

What is version control?

Distributed

Snapshots

Storage landscape

Using Git

Getting started...

commit

branch

merge

Merge conflicts

Exploration

Remote repositories

Industrial workflow

Final tips

- Keeps track of changes to your code.
- You don't have to worry about accidentally losing or deleting code.
- You can experiment and reset to a known good state.
- Makes collaborating with others easier.

What is Git?

- Git is software - it is not GitHub, which is a website
- Distributed - everything is kept on your local machine.
- 'Repository' - a collection of code and history.
- 'Commit' - a chunk of saved changes.

Introduction

What is version control?

Distributed

Snapshots

Storage landscape

Using Git

Gitting started...

commit

branch

merge

Merge conflicts

Exploration

Remote repositories

Industrial workflow

Final tips

1 Introduction

What is version control?

Distributed

Snapshots

Storage landscape

2 Using Git

Gitting started...

commit

branch

merge

Merge conflicts

Exploration

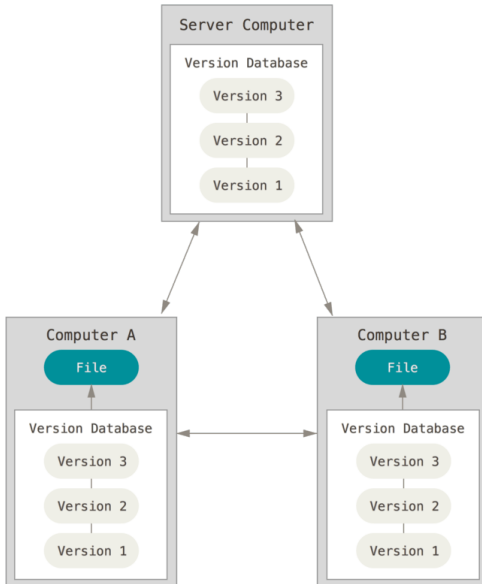
Remote repositories

Industrial workflow

Final tips

Distributed version control

- Introduction
 - What is version control?
 - Distributed**
 - Snapshots
 - Storage landscape
- Using Git
 - Gitting started...
 - commit
 - branch
 - merge
 - Merge conflicts
 - Exploration
 - Remote repositories
 - Industrial workflow
 - Final tips



Introduction

- What is version control?
- Distributed
- Snapshots**
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

1 Introduction

What is version control?

Distributed

Snapshots

Storage landscape

2 Using Git

Gitting started...

commit

branch

merge

- Merge conflicts

Exploration

Remote repositories

Industrial workflow

Final tips

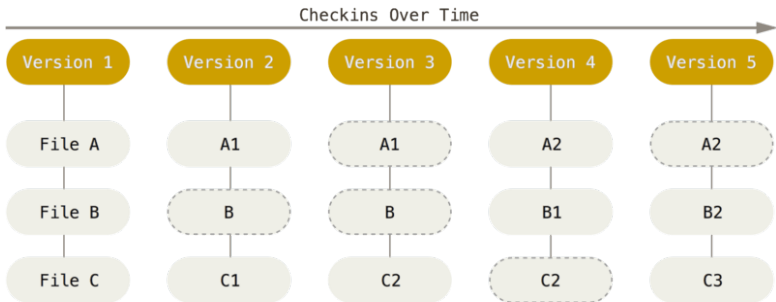
Snapshots (commits) include all files

Introduction

- What is version control?
- Distributed
- Snapshots**
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
- Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips



Introduction

- What is version control?
- Distributed Snapshots
- Storage landscape**

Using Git

- Gitting started...
 - commit
 - branch
 - merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

1 Introduction

- What is version control?
- Distributed Snapshots
- Storage landscape**

2 Using Git

- Gitting started...
 - commit
 - branch
 - merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

Three places where edits exist

Introduction

What is version control?

Distributed

Snapshots

Storage landscape

Using Git

Gitting started...

commit

branch

merge

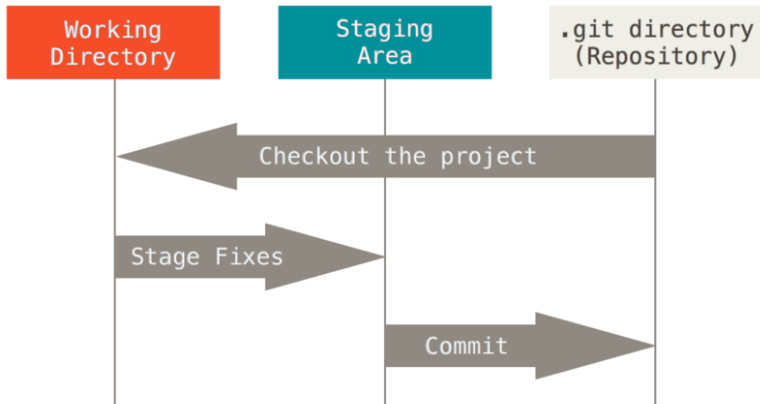
Merge conflicts

Exploration

Remote repositories

Industrial workflow

Final tips



Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

1 Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

2 Using Git

- Gitting started...
- commit
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

Gitting started...

- commit
- branch
- merge
 - Merge conflicts
- Exploration

Remote repositories

- Industrial workflow

Final tips

1 Introduction

What is version control?

Distributed

Snapshots

Storage landscape

2 Using Git

Gitting started...

commit

branch

merge

- Merge conflicts

Exploration

Remote repositories

Industrial workflow

Final tips

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

- `$ git init` Makes a new empty git repository.
- `$ git add <FILENAME>` Adds changes in `FILENAME` to the next commit.
- `$ git status` Shows the status of the repository.
- `$ git config --global user.name "<YOUR FULL NAME>"`
- `$ git config --global user.email <EMAIL>`
- `$ git config --global core.editor vim`

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit**
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

1 Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

2 Using Git

- Gitting started...
- commit**
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

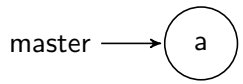
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit**
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git commit -m 'a'
```



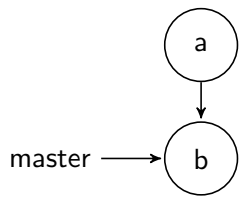
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit**
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git commit -m 'b'
```



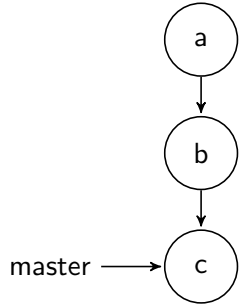
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit**
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git commit -m 'c'
```



Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch**
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

1 Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

2 Using Git

- Gitting started...
- commit
- branch**
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

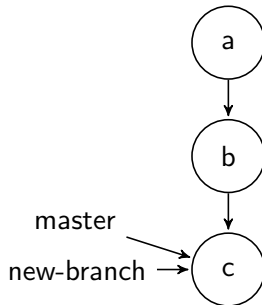
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch**
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git checkout -b new-branch
```



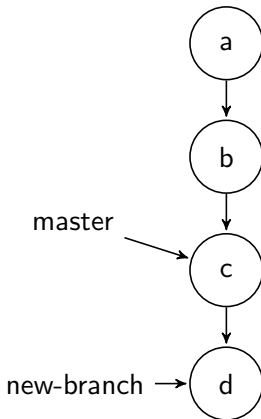
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch**
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git commit -m 'd'
```

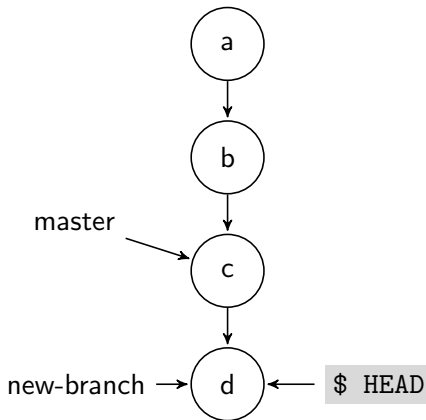


Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch**
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips



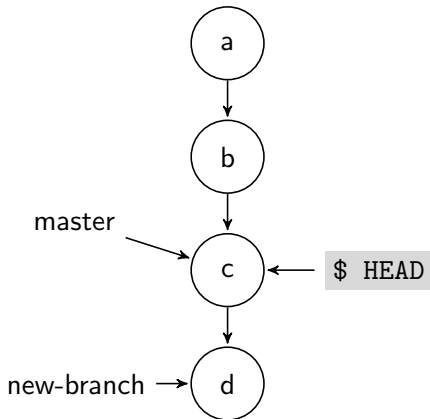
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch**
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git checkout master
```



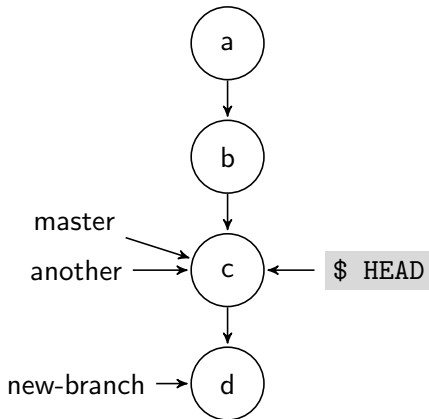
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch**
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git checkout -b another
```



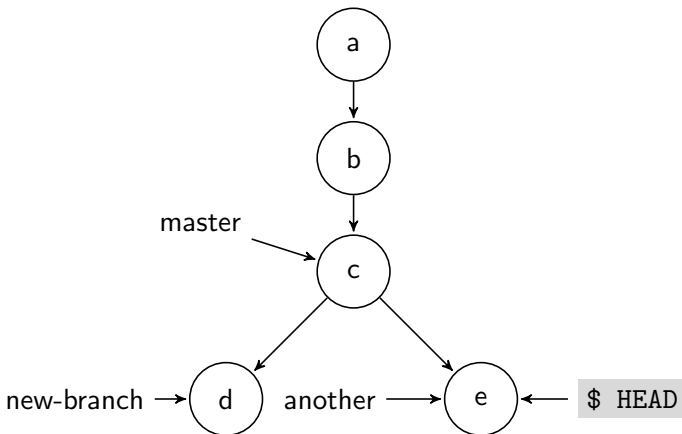
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch**
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git commit -m 'e'
```



Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge**
- Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

1 Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

2 Using Git

- Gitting started...
- commit
- branch
- merge**
- Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

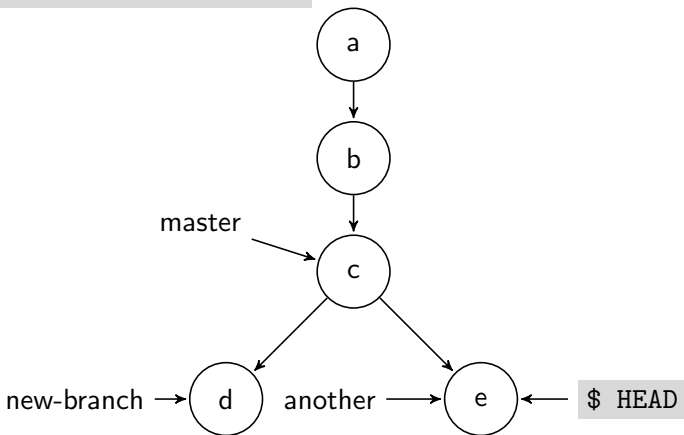
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

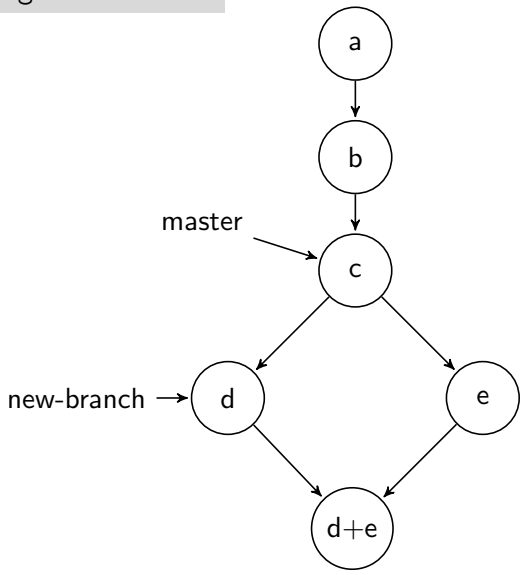
Using Git

- Gitting started...
- commit
- branch
- merge**
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git merge new-branch
```

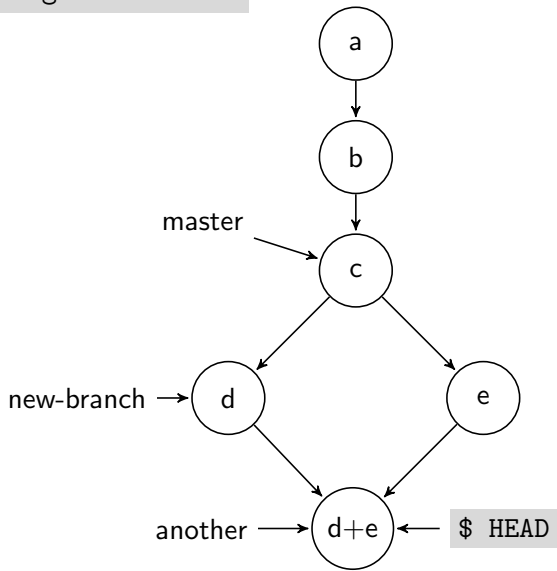


```
$ git merge new-branch
```



- Introduction
- What is version control?
- Distributed
- Snapshots
- Storage landscape
- Using Git
- Getting started...
- commit
- branch
- merge**
- Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

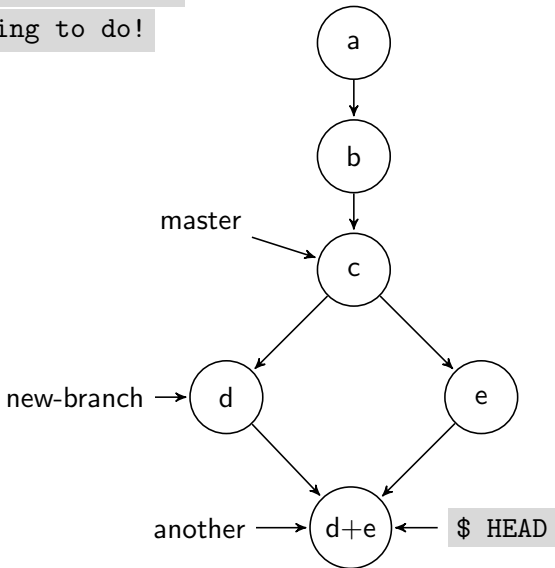
```
$ git merge new-branch
```



- Introduction
- What is version control?
- Distributed
- Snapshots
- Storage landscape
- Using Git
- Gitting started...
- commit
- branch
- merge**
- Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git merge master
```

```
$ Nothing to do!
```



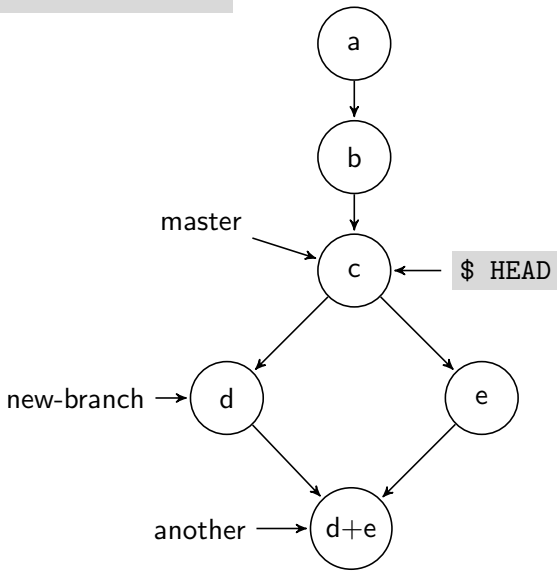
Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

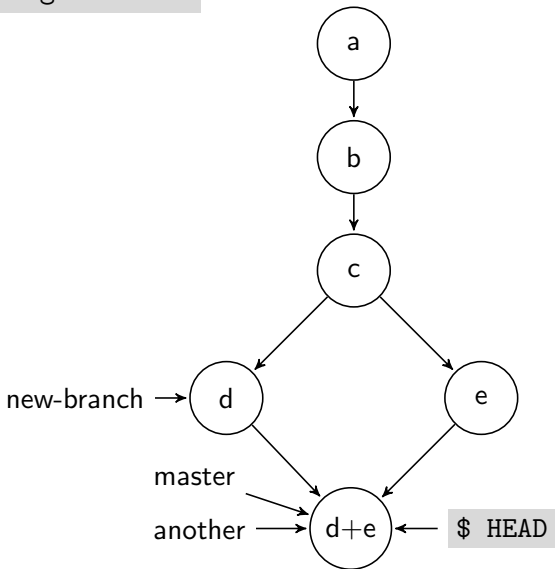
- Gitting started...
- commit
- branch
- merge**
- Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git checkout master
```



- Introduction
- What is version control?
- Distributed
- Snapshots
- Storage landscape
- Using Git
- Getting started...
- commit
- branch
- merge**
- Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

```
$ git merge another
```



Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge**
- Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
- Merge conflicts**

- Exploration
- Remote repositories
- Industrial workflow
- Final tips

CONFLICT (content): Merge conflict in the-file.txt

Automatic merge failed; fix conflicts and then commit the result.

In the-file.txt:

```
<<<<<<< HEAD
```

The current branch's contents

```
=====
```

Stuff from the branch you're merging

```
>>>>>>> new-branch
```

```
$ git add the-file.txt and $ git commit
```


Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
- Merge conflicts

Exploration

- Remote repositories
- Industrial workflow
- Final tips

1 Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

2 Using Git

- Gitting started...
- commit
- branch
- merge
- Merge conflicts

Exploration

- Remote repositories
- Industrial workflow
- Final tips

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
 - Merge conflicts

Exploration

- Remote repositories
- Industrial workflow
- Final tips

- `$ git log` Show a log of commits
 - `--graph` Neat ASCII graph
 - `-p` Show what changed in each commit
- `$ git diff` Show uncommitted changes
- `$ gitk` Graphical log
 - `--all` Show all branches
- `$ git gui` Graphical tool for committing

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
- Merge conflicts
- Exploration

Remote repositories

- Industrial workflow
- Final tips

1 Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

2 Using Git

- Gitting started...
- commit
- branch
- merge
- Merge conflicts
- Exploration
- Remote repositories**
- Industrial workflow
- Final tips

Introduction

What is version control?
 Distributed
 Snapshots
 Storage landscape

Using Git

Getting started...
 commit
 branch
 merge

Merge conflicts
 Exploration

Remote repositories

Industrial workflow

Final tips

- `$ git clone <REPO_URL>` makes a copy of a repository.
- `$ git push` Pushes changes from your current branch to the remote branch it tracks.
 (You may need to run
`$ git config --global push.default simple`.)
- `$ git pull` Pulls changes from the remote branch and merges them into your current branch.
- `$ git remote add <REMOTE_NAME> <REPO_URL>` adds a remote to an existing repository.

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow**
- Final tips

1 Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

2 Using Git

- Gitting started...
- commit
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow**
- Final tips

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
 - commit
 - branch
 - merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips

- You and your co-workers are working on a project simultaneously
- You clone the company's repository:


```
$ git clone https://git.company.com/project.git
```
- `$ git checkout -b doug` to create your own development branch
- Modify files, `$ git add <FILENAME>` to stage them, `$ git commit` when they are in a working state.
- Ready to merge with mainline?


```
$ git checkout master
```

 and

```
$ git merge doug
```

.
- Your work is now merged with your *local* `master` branch (but not on the company's repo).
- Question: which branch is `HEAD` now pointing to?

Introduction

What is version control?
 Distributed
 Snapshots
 Storage landscape

Using Git

Gitting started...
 commit
 branch
 merge
 Merge conflicts

Exploration

Remote repositories

Industrial workflow

Final tips

- Meanwhile, your co-workers might have made changes!
- First, `$ git pull` to fetch and merge their changes
- Rectify merge conflicts, test the code, then `$ git add <FILENAME>` to stage, and then `$ git commit` when in a working state
- Only after pulling and merging the most recent changes should you `$ git push`
- Your work is merged with that of your co-workers, and now resides on the company repo
- Take a coffee break.

Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

Using Git

- Gitting started...
- commit
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips**

1 Introduction

- What is version control?
- Distributed
- Snapshots
- Storage landscape

2 Using Git

- Gitting started...
- commit
- branch
- merge
 - Merge conflicts
- Exploration
- Remote repositories
- Industrial workflow
- Final tips**

Introduction

What is version control?
Distributed
Snapshots
Storage landscape

Using Git

Gitting started...
commit
branch
merge
Merge conflicts
Exploration
Remote repositories
Industrial workflow
Final tips

- Make your commit messages descriptive!
- Only `$ git commit` when the code works!
- Don't add generated files (like `a.out`) to your repo.
- You can ignore certain files by putting their names in a `.gitignore` file in your repo.
- When collaborating, work on separate branches and merge as you go along.
- `$ git help COMMAND` will show you documentation.