LEIDEN UNIVERSITY MEDICAL CENTER

Git and remote repositories

Martijn Vermaat

Leiden Genome Technology Center

Department of Human Genetics

Center for Human and Clinical Genetics

Remote repositories

Table of contents

Remote repositories

Distributed Git

Repositories can reference each other:

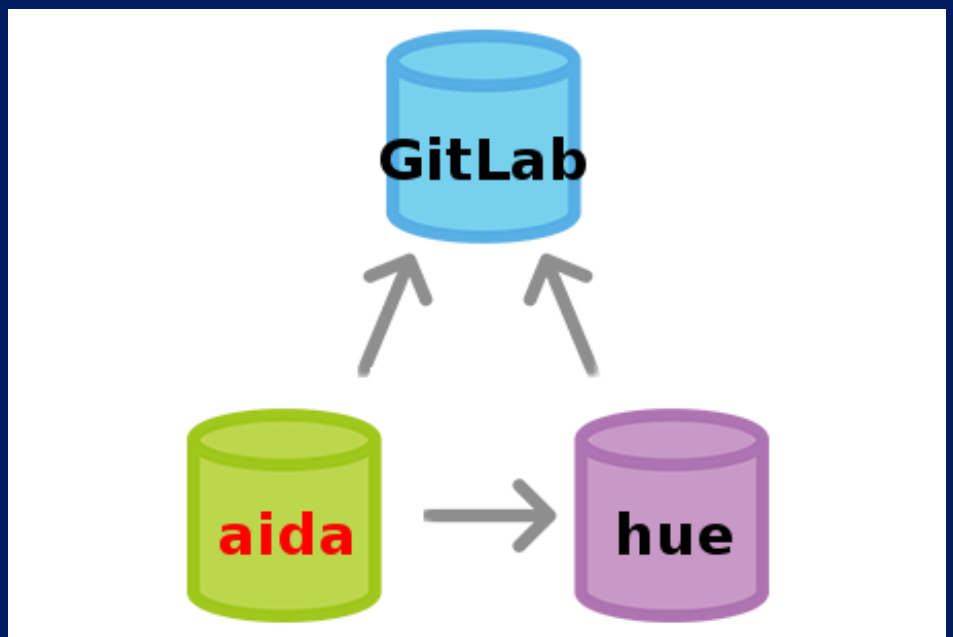
- A repository can be on a server, your desktop, your coworker's laptop, etc.
- Technically, no repository is 'special'.
- We call a reference to another repository a *remote*.

Remote repositories

Distributed Git

Repositories can reference each other:

- A repository can be on a server, your desktop, your coworker's laptop, etc.
- Technically, no repository is 'special'.
- We call a reference to another repository a *remote*.



Remote repositories

Listing remotes

```
$ git remote
gitlab
```

We are on **aida** and have one remote, **gitlab**, defined.

Remote repositories

Listing remotes

```
$ git remote
gitlab
```

We are on **aida** and have one remote, **gitlab**, defined.

```
$ git remote -v
gitlab  https://git.lumc.nl/m.vermaat.hg/tv-series.git (fetch)
gitlab  https://git.lumc.nl/m.vermaat.hg/tv-series.git (push)
```

-v: Include remote location.

We see that communication with **gitlab** is over HTTPS.

Remote repositories

Adding a remote: `git remote add`

```
$ git remote add hue 192.168.0.8:projects/tv-series
```

This adds a reference to the remote repository **`hue`** using communication over SSH.

Remote repositories

Adding a remote: `git remote add`

```
$ git remote add hue 192.168.0.8:projects/tv-series
```

This adds a reference to the remote repository **`hue`** using communication over SSH.

```
$ git remote -v
gitlab  https://git.lumc.nl/m.vermaat.hg/tv-series.git (fetch)
gitlab  https://git.lumc.nl/m.vermaat.hg/tv-series.git (push)
hue     192.168.0.8:projects/tv-series (fetch)
hue     192.168.0.8:projects/tv-series (push)
```

Remote repositories

Remote branches

You see code from remotes as *remote branches*:

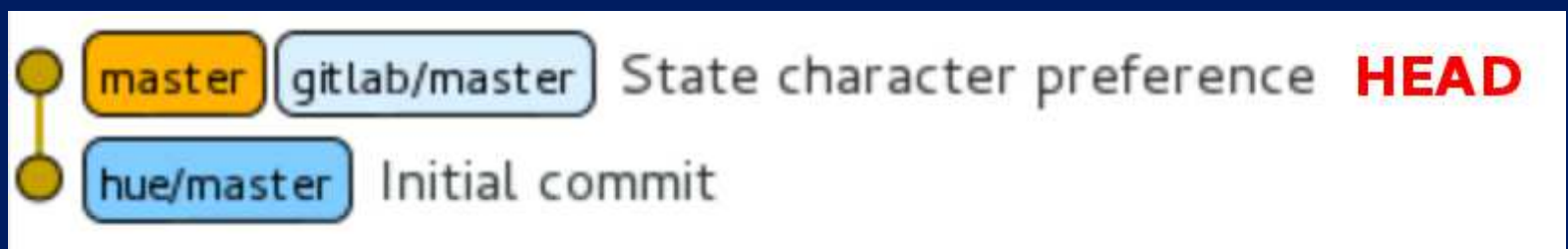
- Remote branches are just branches prefixed with the remote name.
- Communicate with remotes to update the remote branches.
- You can setup a local branch to *track* a remote branch.

Remote repositories

Remote branches

You see code from remotes as *remote branches*:

- Remote branches are just branches prefixed with the remote name.
- Communicate with remotes to update the remote branches.
- You can setup a local branch to *track* a remote branch.



Remote repositories

Listing remote branches: `git branch`

```
$ git branch -v
* master  f1ef19c State character preference
```

This shows only our local branches.

Remote repositories

Listing remote branches: `git branch`

```
$ git branch -v
* master  f1ef19c State character preference
```

This shows only our local branches.

```
$ git branch -v -a
* master                  f1ef19c State character preference
  remotes/gitlab/master f1ef19c State character preference
```

`-a`: Include remote branches.

Remote repositories

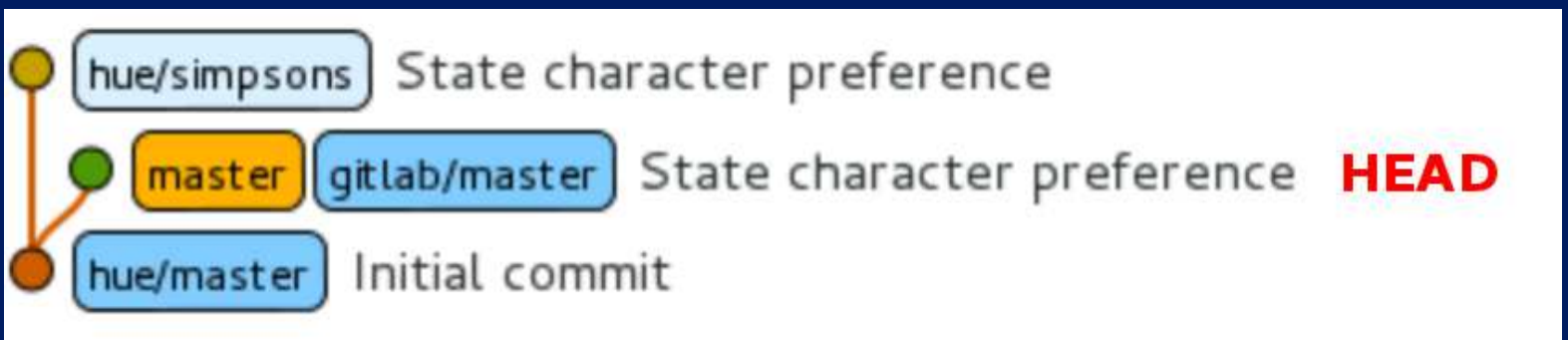
Updating remote branches: `git fetch`

```
$ git fetch hue
remote: Counting objects: 5, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (3/3), done.
From 192.168.0.8:projects/tv-series
 * [new branch]      master     -> hue/master
 * [new branch]      simpsons   -> hue/simpsons
```

Remote repositories

Updating remote branches: `git fetch`

```
$ git fetch hue
remote: Counting objects: 5, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (3/3), done.
From 192.168.0.8:projects/tv-series
 * [new branch]      master     -> hue/master
 * [new branch]      simpsons   -> hue/simpsons
```



Remote repositories

Merging remote branches: `git merge`

We can merge remote branches just like normal branches.

```
$ git merge hue/simpsons
...
```

Remote repositories

Merging remote branches: `git merge`

We can merge remote branches just like normal branches.

```
$ git merge hue/simpsons
...
```

Alternatively, we could continue the work of a remote branch.

Remote repositories

Creating a local tracking branch: `git checkout`

Remote branches are 'read-only':

- We cannot directly continue work on a remote branch.
- But we can setup a local *tracking branch*.

```
$ git checkout simpsons
Branch simpsons set up to track remote branch
simpsons from hue.
Switched to a new branch 'simpsons'
```

What happened here?

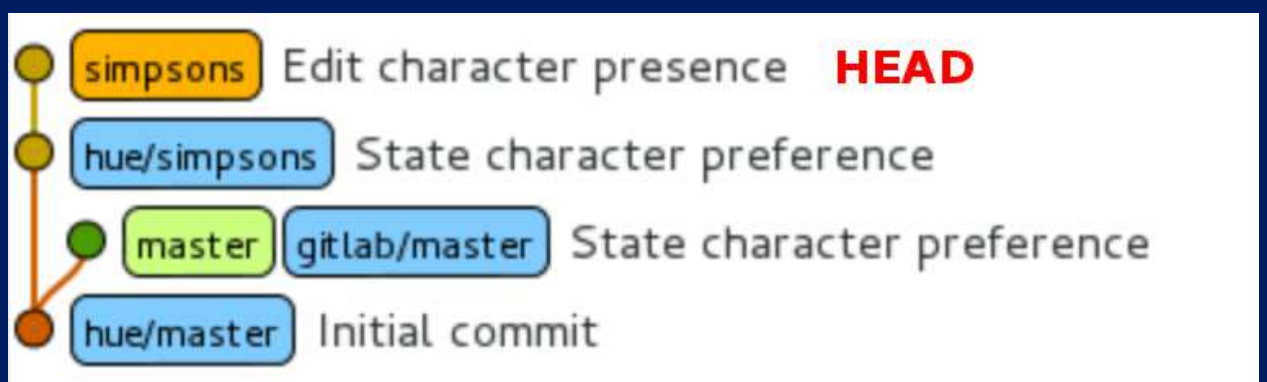
- There was no branch **`simpsons`**.
- But there was a remote branch of the same name.
- So Git creates a new branch based on that (and switches to it).

Remote repositories

Working on a tracking branch

Let's continue working on branch **`simpsons`**.

```
$ emacs FACTS.md
$ git add FACTS.md
$ git commit -m 'Edit character presence'
[simpsons 0676334] Edit character presence
 1 file changed, 1 insertion(+), 1 deletion(-)
$ git status
# On branch simpsons
# Your branch is ahead of 'hue/simpsons' by 1 commit.
#
nothing to commit (working directory clean)
```



Remote repositories

Pushing changes to a remote: `git push`

```
$ git push hue simpsons
Counting objects: 5, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 303 bytes, done.
Total 3 (delta 1), reused 0 (delta 0)
To hue.remote:projects/tv-series
   0535b7e..0676334  simpsons -> simpsons
```

Our work on branch **`simpsons`** is now available on remote **`hue`** too.

Remote repositories

Cloning an existing repository

Instead of creating repositories using **`git init`**, you can create a local *clone* of an existing (remote) repository.

```
$ git clone https://git.lumc.nl/m.vermaat.hg/tv-series.git
Cloning into 'tv-series'...
remote: Counting objects: 6, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 6 (delta 1), reused 0 (delta 0)
Unpacking objects: 100% (6/6), done.
```

Remote repositories

Cloning an existing repository

Instead of creating repositories using **`git init`**, you can create a local *clone* of an existing (remote) repository.

```
$ git clone https://git.lumc.nl/m.vermaat.hg/tv-series.git
Cloning into 'tv-series'...
remote: Counting objects: 6, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 6 (delta 1), reused 0 (delta 0)
Unpacking objects: 100% (6/6), done.
```

A remote called **`origin`** is added for the original repository and branch **`master`** is setup to track the original **`master`**.

```
$ cd tv-series/
$ git remote -v
origin  https://git.lumc.nl/m.vermaat.hg/tv-series.git (fetch)
origin  https://git.lumc.nl/m.vermaat.hg/tv-series.git (push)
```

Remote repositories

Shortcuts for tracking branches

If a local branch is setup to track a remote branch, there are shortcuts for synchronizing them.

Remote repositories

Shortcuts for tracking branches

If a local branch is setup to track a remote branch, there are shortcuts for synchronizing them.

For example, say **`HEAD`** is on **`simpsons`** which is tracking **`origin/simpsons`**.

```
$ git push    ==>    $ git push origin simpsons

$ git pull    ==>    $ git fetch origin
                     $ git merge origin/simpsons
```

Working with GitLab

Table of contents

Working with GitLab

Using a central server

Git can be used by a team completely decentralized.

However, often a central server is used:

- It can be easier to communicate via the server.
- It can be convenient to have a canonical repository.
- Services such as *GitLab* and *GitHub* add many features on top of Git.

Working with GitLab

GitLab

Our GitLab server is at `https://git.lumc.nl/`

- Coupled to your LUMC account.
- All users can create projects.
- Browse repositories and edit files online.
- Control access for other users.
- Track bugs/issues/tickets.
- Create merge requests and do code reviews.

Questions?

Table of contents

Questions?

Acknowledgements:

Jeroen Laros
Zuotian Tatum

http://git-scm.com/book

https://www.atlassian.com/git