

Version control

A server holds the code and every revision in a repository

Clients can pull/push data to the server

Each client has a copy of the full repository

Example systems: CVS, subversion, and git

Can be used for more than coding e.g. latex reports

Example: gitlab.com

Go to gitlab.com and make an account

Create a new project

Copy the link
(not my one, get your own!)



The screenshot shows the GitLab interface for a newly created repository named 'Python lab'. At the top, there is a light blue circular avatar with the letter 'P'. Below the avatar, the project name 'Python lab' is displayed. Underneath the project name, there is a row of buttons: a star icon with '0', 'SSH', 'HTTPS', the repository URL 'git@gitlab.com:tom00/python-lab', a lock icon, and a plus icon. A green arrow points from the text 'Copy the link (not my one, get your own!)' to the repository URL. Below this row, a message states 'The repository for this project is empty'. Further down, it provides instructions: 'If you already have files you can push them using command line instructions below. Otherwise you can start with adding README file to this project.'

★ 0 SSH HTTPS git@gitlab.com:tom00/python-lab 🔒 +

The repository for this project is empty

If you already have files you can push them using command line instructions below.
Otherwise you can start with adding README file to this project.

ssh key

SSH keys are a way to identify trusted computers.

Use “ssh-keygen -t rsa” to generate a key pair

```
tom00@csg007:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/tom00/.ssh/id_rsa): /home/tom00/.ssh/id_rsa_new
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/tom00/.ssh/id_rsa_new.
Your public key has been saved in /home/tom00/.ssh/id_rsa_new.pub.
```

```
tom00@csg007:~$ ls ~/.ssh/id_rsa_new*
/home/tom00/.ssh/id_rsa_new  /home/tom00/.ssh/id_rsa_new.pub
tom00@csg007:~$ cat /home/tom00/.ssh/id_rsa_new.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCA4lymgxVHPkB3QKzLYFvdbf2MacgjEJaDwEsSjn6V
ciVgaInhx37xrXB8a63lqueK52KoNkzGpy2m82YLZ2zk0rDRjZmFcUZt8NtC0U759Q5r0eI7Dr6xC/J
vrGA4mcXi9xTGX0xLgkgrsP20wcnd4y2etkfSAXH57E6LrLQWrk4h2LtPvbNEaI5TEgcP2ceCwuMAv7
TsfqlxfrmXRIMu3SpAin4FQvmMqIW7selb6iAIrjM5yFt+gmbShUEY6lA6KXbem+XViKdK08/09R3/R
faZ0ZjSS6t+DkoSoyWR9bUZgzLGibSILPv/wi0k09NZR/DNbX+RqW460n4TIZMd tom00@csg007
tom00@csg007:~$
```



This is my public key (yours will be different), copy it to the gitlab website

Clone the repository

tom00@csg007: ~

File Edit View Search Terminal Help

```
tom00@csg007:~$ git clone git@gitlab.com:tom00/python-lab.git
Cloning into 'python-lab'...
warning: You appear to have cloned an empty repository.
Checking connectivity... done.
tom00@csg007:~$
```

```
tom00@csg007:~$ cd python-lab/
tom00@csg007:~/python-lab$ ls
tom00@csg007:~/python-lab$ ls -la
total 12
drwxr-xr-x  3 tom00 tom00 4096 Nov 30 17:04 .
drwxr-xr-x 74 tom00 tom00 4096 Nov 30 17:04 ..
drwxr-xr-x  7 tom00 tom00 4096 Nov 30 17:04 .git
tom00@csg007:~/python-lab$
```

This is a clone of the repository
i.e. exactly the same as
what is on the server

The repository is empty, but
There is a .git file that stores
All the revision history

And how to connect to the
server

Start coding

In the repository make a file e.g. `touch script.py`

Although this file is in the same directory it is not included in the repository list (`.git`)

The file needs to be added

```
git add script.py
```

Once you are happy with your revisions you need to commit this to your local repository

```
git commit -a
```

(you must add text to describe your changes)

Finally you can push you repository back to the server

```
git push
```

(the repository remembers the server)

Viewing revisions

Each revision can be viewed

python-lab

add more



tom00 authored about a minute ago

Add to script.py



tom00 authored 2 minutes ago

changes



tom00 authored 5 minutes ago

Differences can be shown
between revisions

Authored by  **tom00** 6 minutes ago

1 parent 28ef8fa0

 master

DOWNLOAD AS ▾

BROWSE CODE ▸

add more

Showing **1 changed file** with **1 additions** and **1 deletions**

INLINE

SIDE-BY-SIDE

script.py



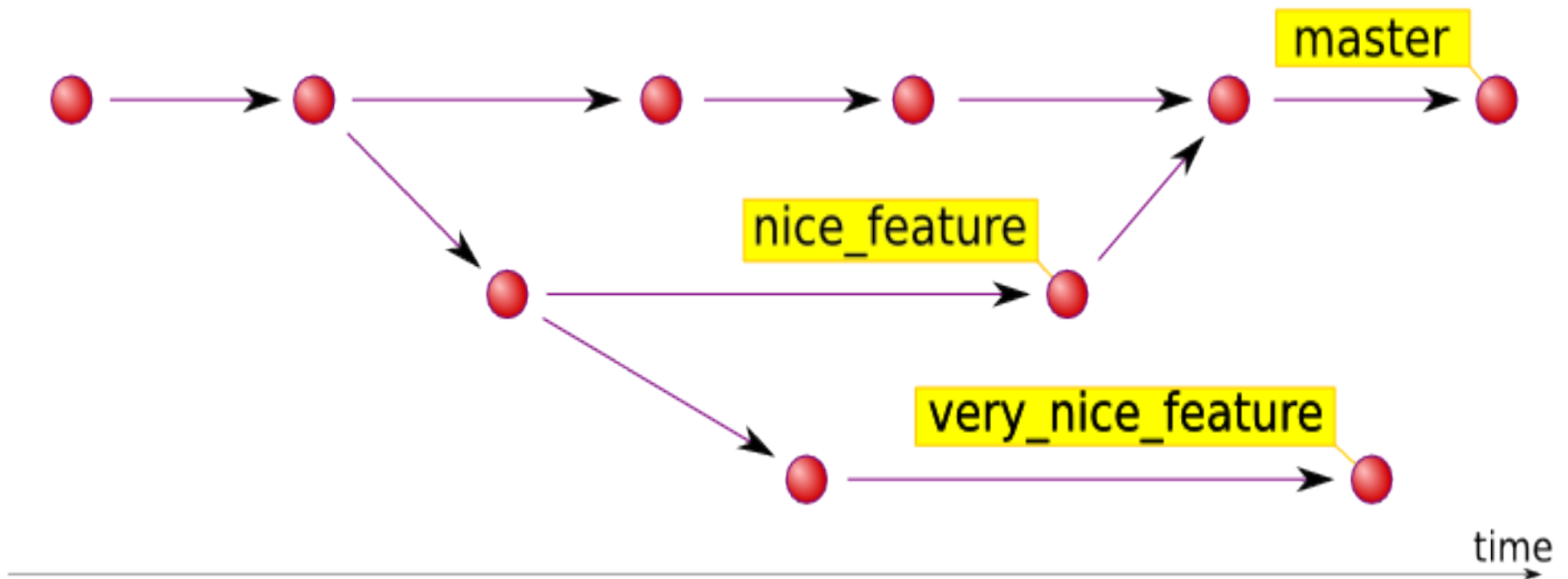
VIEW FILE @503711A

1 -I add something now

1 +I add something now, now I add some more

Branching

Branching allows you to try experimental changes which may or may not be merged back into the master



Branching

```
tom00@csg007:~/python-lab$ git branch
* master

tom00@csg007:~/python-lab$ git branch test
tom00@csg007:~/python-lab$ git branch
* master
  test

tom00@csg007:~/python-lab$ git checkout test
Switched to branch 'test'
tom00@csg007:~/python-lab$
```

List branches

Create a new branch

Switch to editing the "test" branch

Now that we have branches we have to use

`git push --all`

To send all branches to the server

Make changes

`git commit -a`

`git push --all`

Master branch and test branch are now different

The diagram illustrates the state of a repository with two branches: master and test. Two green arrows originate from a central point above the text 'Master branch and test branch are now different' and point towards the respective branch panels. The master branch panel on the left shows a commit history of 3 commits as of 30 Nov, 2015, with recent changes including 'add more' and 'Add to script.py'. The test branch panel on the right shows a commit history of 4 commits as of 30 Nov, 2015, with recent changes including 'changes to test branch', 'add more', 'Add to script.py', and 'changes'. Both panels are for the 'python-lab' repository.

master python-lab

30 Nov, 2015
3 commits

add more
tom00 authored 30 minutes ago

Add to script.py
tom00 authored 31 minutes ago

changes
tom00 authored 34 minutes ago

test python-lab

30 Nov, 2015
4 commits

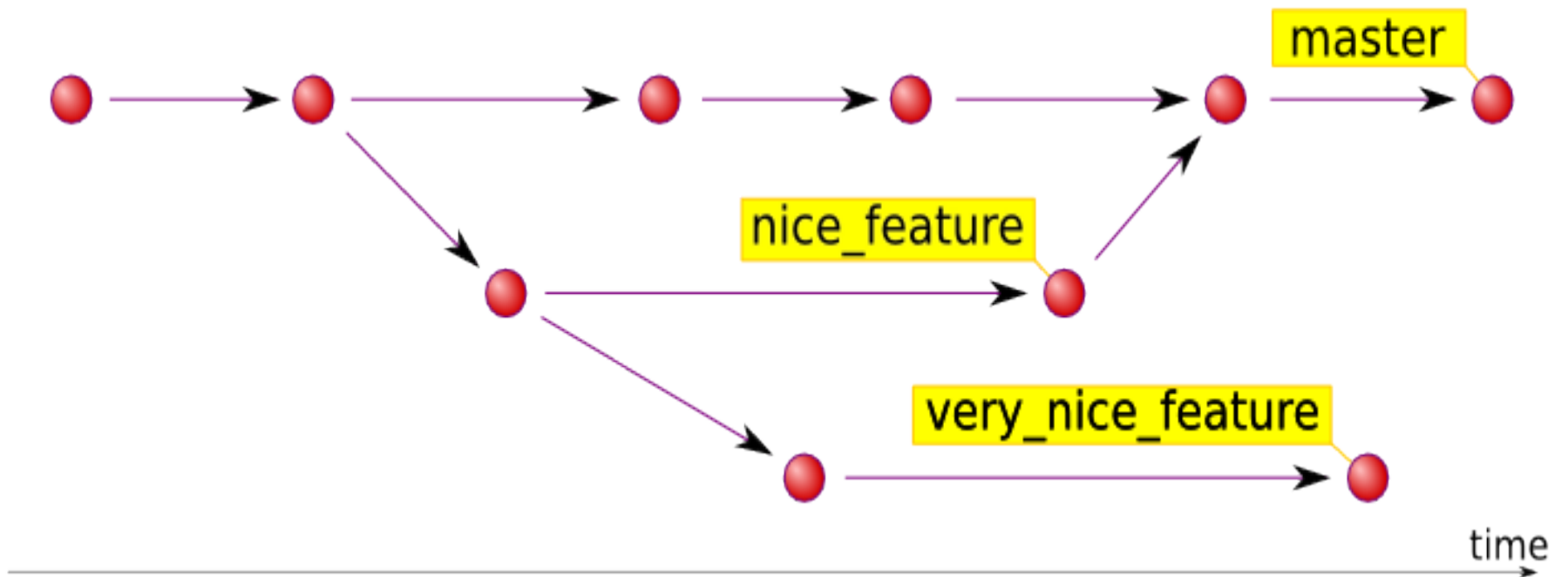
changes to test branch
tom00 authored 3 minutes ago

add more
tom00 authored 29 minutes ago

Add to script.py
tom00 authored 30 minutes ago

changes
tom00 authored 34 minutes ago

Merging



Merging


```
tom00@csg007:~/python-lab$ git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.
tom00@csg007:~/python-lab$ git merge test
Updating 503711a..48ae6c6
Fast-forward
 script.py | 4 +++-
 1 file changed, 3 insertions(+), 1 deletion(-)
tom00@csg007:~/python-lab$
```


Goto master branch


Merge changes in test
back with the master branch


master


python-lab

 30 Nov, 2015
4 commits

changes to test branch
 tom00 authored 12 minutes ago

add more
 tom00 authored 38 minutes ago

Add to script.py
 tom00 authored 39 minutes ago

changes
 tom00 authored 42 minutes ago

Conflicts

Git handles easy merges automatically
If it does not know how to merge it asks

```
tom00@csg007:~/python-lab$ git merge test2
Auto-merging script.py
CONFLICT (content): Merge conflict in script.py
Automatic merge failed; fix conflicts and then commit the result.
tom00@csg007:~/python-lab$ █
```

```
tom00@csg007:~/python-lab$ cat script.py
<<<<<<< HEAD
I don't want to add something now, now I add some more.
=====
I might add something now, now I add some more.
>>>>>>> test2

I want to test some changes now in a new branch
tom00@csg007:~/python-lab$ █
```

Fixing conflicts

```
tom00@csg007:~/python-lab$ cat script.py
<<<<<<< HEAD
I don't want to add something now, now I add some more.
=====
I might add something now, now I add some more.
>>>>>>> test2

I want to test some changes now in a new branch
tom00@csg007:~/python-lab$ █
```

Simple: edit the file and select the revision you want
Conflict resolution software is available e.g. “git mergetool”

```
tom00@csg007:~/python-lab$ cat script.py
I might add something now, now I add some more.

I want to test some changes now in a new branch
tom00@csg007:~/python-lab$ █
```

Reset

You can take the master back in time to any committed revision

```
tom00@csg007:~/python-lab$ git log --oneline
ee2fa7f Revert "test"
66382e2 retest
286ba5f test
c4d8f9a fix conflict Merge branch 'test2'
1bef609 add something
d42112a add a conflict
48ae6c6 changes to test branch
503711a add more
28ef8fa Add to script.py
65731d8 changes
tom00@csg007:~/python-lab$ git reset --hard 66382e2
HEAD is now at 66382e2 retest
```

Practical problem

Create a gitlab account

In groups of 2 or 3, use git to develop a script to combine all pdf files in a directory into a single file:

Hint, ghost script can be used as follows,

```
gs -dBATCH -dNOPAUSE -q -sDEVICE=pdfwrite -sOutputFile=out_file in1.pdf in2.pdf
```

This will join in1.pdf and in2.pdf and create a file “out_file”