

RE-COOP #8:

A round-up around GIT ...
... informal of course !!!

19th March 2015

Guillaume Lemaître
Joan Massich

Universitat de Girona, Université de Bourgogne



① History

② What is GIT?

Definitions

1st study case

2nd study case

3rd study case

4th study case

③ GIT more in details

The big picture

Branch principle

GitHub

Usual commands

④ GIT in even more details

git init

git add

git status

git commit

git log

git status

git diff

gitg

git branch

git checkout

Push forward

git merge

GitHub

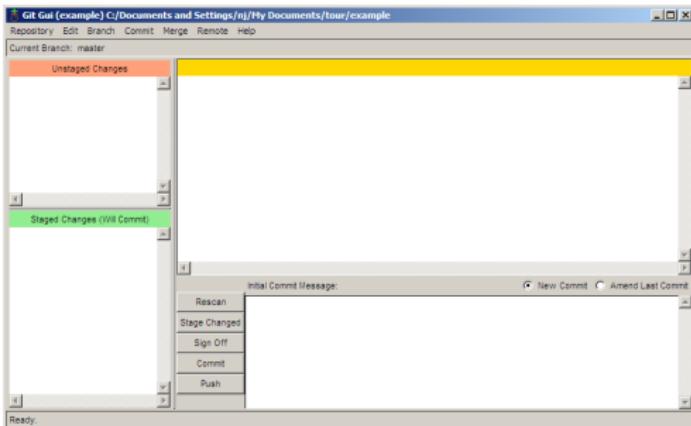
git remote add

git push



A bit of History ...

For Microsoft Windows users





A bit of History ...

For Linux users

```

[2] klosterhoff@klosterhoff:~$ git clone https://github.com/klosterhoff/test-repo.git
Cloning into 'test-repo'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100%, 10 objects
remote: Compressing objects: 100%, 10 objects
remote: Total 10 (delta 0), reused 0 objects (0 bytes), pack-reused 0 objects
Resolving deltas: 100% (10/10)
Checking connectivity... done.

[3] klosterhoff@klosterhoff:~/test-repo$ ls
README.md  test-repo

[4] klosterhoff@klosterhoff:~/test-repo$ cd test-repo/
[5] klosterhoff@klosterhoff:~/test-repo$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

[6] klosterhoff@klosterhoff:~/test-repo$ git add .
[7] klosterhoff@klosterhoff:~/test-repo$ git commit -m "Initial Commit"
[master 26d4a45] Initial Commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

[8] klosterhoff@klosterhoff:~/test-repo$ git log
commit 26d4a45 (HEAD → master)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sat Aug 22 22:18:33 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

[9] klosterhoff@klosterhoff:~/test-repo$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

[10] klosterhoff@klosterhoff:~/test-repo$ git push
Pushing to https://github.com/klosterhoff/test-repo.git
To https://github.com/klosterhoff/test-repo.git
 ! [rejected]        master <origin/master>
error: failed to push some refs to 'https://github.com/klosterhoff/test-repo.git'
hint: Updates were rejected because they would move history.
hint: This is likely due to repository settings on the remote end.
hint: You can avoid this error by explicitly forcing the update.
hint: See the 'push --force' documentation for details.

[11] klosterhoff@klosterhoff:~/test-repo$ git pull
From https://github.com/klosterhoff/test-repo
 * branch            master     -> FETCH_HEAD
Already up-to-date.

[12] klosterhoff@klosterhoff:~/test-repo$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

[13] klosterhoff@klosterhoff:~/test-repo$ git log
commit 26d4a45 (HEAD → master)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sat Aug 22 22:18:33 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

[14] klosterhoff@klosterhoff:~/test-repo$ git merge develop
Auto-merging README.md
CONFLICT (add/add): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the merged file.

[15] klosterhoff@klosterhoff:~/test-repo$ git status
On branch master
Your branch is behind 'origin/develop' by 1 commit.
  (use "git pull" to update your local branch)

Changes to be committed:
  (use "git reset HEAD <file>" to unstage)

    new file:   README.md

Changes not staged for commit:
  (use "git add <file>" to stage these changes)

    modified:  README.md

[16] klosterhoff@klosterhoff:~/test-repo$ git pull
From https://github.com/klosterhoff/test-repo
 * branch            master     -> FETCH_HEAD
Merge made by 'git pull'.
  CONFLICT (add/add): Merge conflict in README.md
  Merge conflict resolution included in previous commit.
  Resolve the conflicts and then commit the merged file.

[17] klosterhoff@klosterhoff:~/test-repo$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

[18] klosterhoff@klosterhoff:~/test-repo$ git log
commit 26d4a45 (HEAD → master)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sat Aug 22 22:18:33 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

commit 9e9a30c (origin/develop)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sun Aug 23 14:47:30 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

[19] klosterhoff@klosterhoff:~/test-repo$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

[20] klosterhoff@klosterhoff:~/test-repo$ git merge develop
Auto-merging README.md
CONFLICT (add/add): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the merged file.

[21] klosterhoff@klosterhoff:~/test-repo$ git status
On branch master
Your branch is behind 'origin/develop' by 1 commit.
  (use "git pull" to update your local branch)

Changes to be committed:
  (use "git reset HEAD <file>" to unstage)

    new file:   README.md

Changes not staged for commit:
  (use "git add <file>" to stage these changes)

    modified:  README.md

[22] klosterhoff@klosterhoff:~/test-repo$ git log
commit 26d4a45 (HEAD → master)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sat Aug 22 22:18:33 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

commit 9e9a30c (origin/develop)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sun Aug 23 14:47:30 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

commit e20a2f4 (origin/develop)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sun Aug 23 14:47:30 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

[23] klosterhoff@klosterhoff:~/test-repo$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

[24] klosterhoff@klosterhoff:~/test-repo$ git log
commit 26d4a45 (HEAD → master)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sat Aug 22 22:18:33 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

commit 9e9a30c (origin/develop)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sun Aug 23 14:47:30 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

commit e20a2f4 (origin/develop)
Author: Klosterhoff [mailto:klosterhoff@klosterhoff.de]
Date:   Sun Aug 23 14:47:30 2015 +0200

 1 file changed, 1 insertion(+)
 create mode 100644 README.md

[25] klosterhoff@klosterhoff:~/test-repo$ git log --graph
* 26d4a45 (HEAD → master) Initial Commit
* 9e9a30c (origin/develop) Add a new line to README.md
* e20a2f4 (origin/develop) Add another line to README.md

```

History

What is GIT?
ooooo

GIT more in details
ooooo

GIT in even more details
oooooooooooooooooooo



A bit of History ...
Original author

History

What is GIT?
ooooo

GIT more in details
ooooo

GIT in even more details
oooooooooooooooooooo



A bit of History ...
Original author

Linus Torvalds





What is GIT? Definitions

Formal definition

- ▶ Git is a distributed revision control system with an emphasis on speed, data integrity, and support for distributed, non-linear workflows.
- ▶ Since 2005, it has become the most widely adopted version control system for software development.

In an everyday language

- ▶ THIS IS COOL AND FANCY!!!!



What is GIT? Definitions

Formal definition

- ▶ Git is a distributed revision control system with an emphasis on speed, data integrity, and support for distributed, non-linear workflows.
- ▶ Since 2005, it has become the most widely adopted version control system for software development.

In an everyday language

- ▶ THIS IS COOL AND FANCY!!!!
- ✓ Keep track of all changes performed on a repository,



What is GIT? Definitions

Formal definition

- ▶ Git is a distributed revision control system with an emphasis on speed, data integrity, and support for distributed, non-linear workflows.
- ▶ Since 2005, it has become the most widely adopted version control system for software development.

In an everyday language

- ▶ THIS IS COOL AND FANCY!!!!
- ✓ Keep track of all changes performed on a repository,
- ✓ Allow to have multiple versions,
- ✓ Possibility to swap versions,



What is GIT? Definitions

Formal definition

- ▶ Git is a distributed revision control system with an emphasis on speed, data integrity, and support for distributed, non-linear workflows.
- ▶ Since 2005, it has become the most widely adopted version control system for software development.

In an everyday language

- ▶ THIS IS COOL AND FANCY!!!!
- ✓ Keep track of all changes performed on a repository,
- ✓ Allow to have multiple versions,
- ✓ Possibility to swap versions,
- ✓ Perform collaborative work,



What is GIT? Definitions

Formal definition

- ▶ Git is a distributed revision control system with an emphasis on speed, data integrity, and support for distributed, non-linear workflows.
- ▶ Since 2005, it has become the most widely adopted version control system for software development.

In an everyday language

- ▶ THIS IS COOL AND FANCY!!!!
- ✓ Keep track of all changes performed on a repository,
- ✓ Allow to have multiple versions,
- ✓ Possibility to swap versions,
- ✓ Perform collaborative work,
- ✓ Easy way to merge work,
- ✗ Need to learn the tool.

History

What is GIT?



GIT more in details



GIT in even more details



What is GIT? Study cases

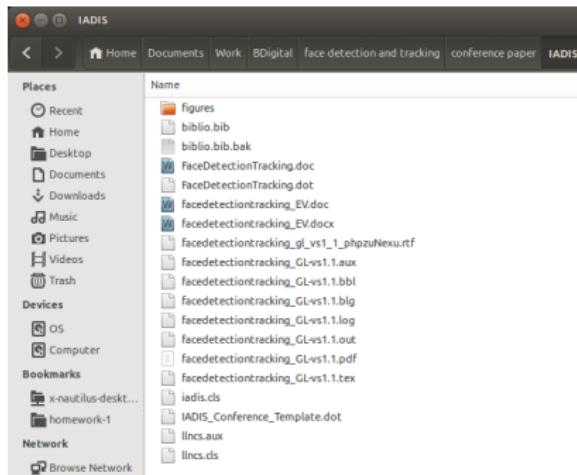
Software versioning





What is GIT? Study cases

Writing case



History

What is GIT?



GIT more in details



GIT in even more details



What is GIT? Study cases

Education case

Guillaume Lemaître / ViBOT Course - Medical Image Ana...

Search in this project

Project Files Commits Network Graphs Issues 0 Merge Requests

master

vibot-course-medical-image-analysis / +

Branches	Date	Last Commit
course_2012_2013_dev_handout	s ago	> b5abfec194a – Add the grate and report for the segmentation
course_2013_2014_dev	s ago	Guillaume Lemaître update the master branch
course_2013_2014_dev_handout	s ago	Guillaume Lemaître update the master branch
master	s ago	Guillaume Lemaître Add anonymized version
segmentation	8 months ago	Guillaume Lemaître Add the grate and report for the segmentation
README.md	12 months ago	Guillaume Lemaître Update the README file

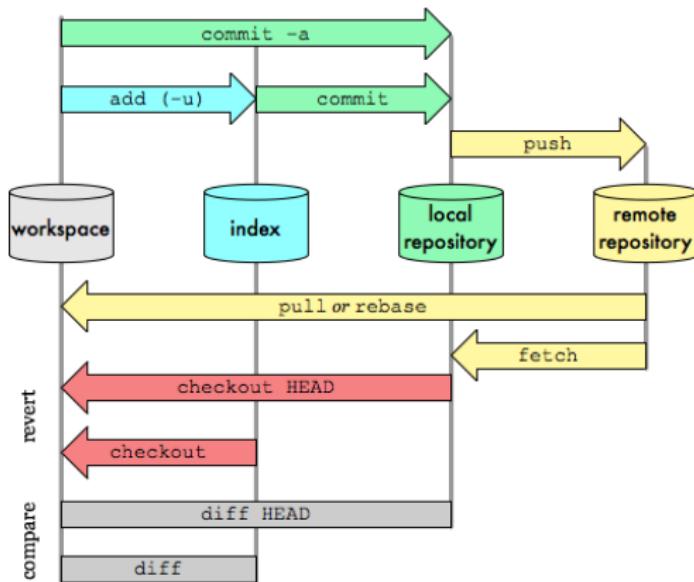


GIT more in details

The big picture

Git Data Transport Commands

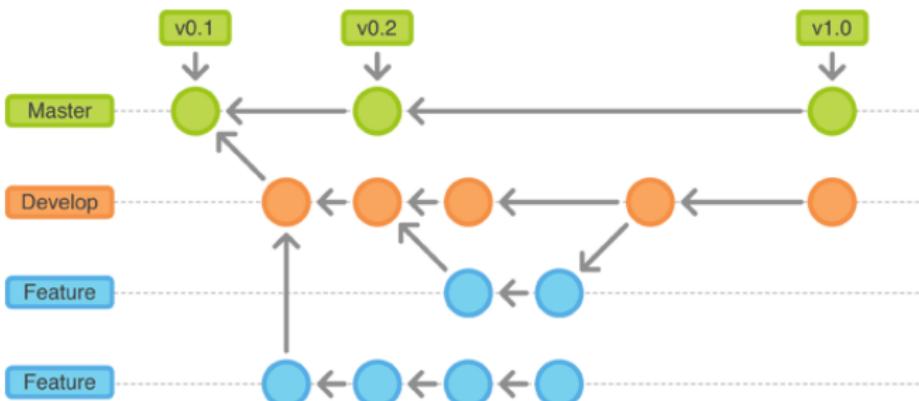
<http://csteelle.com>





GIT more in details

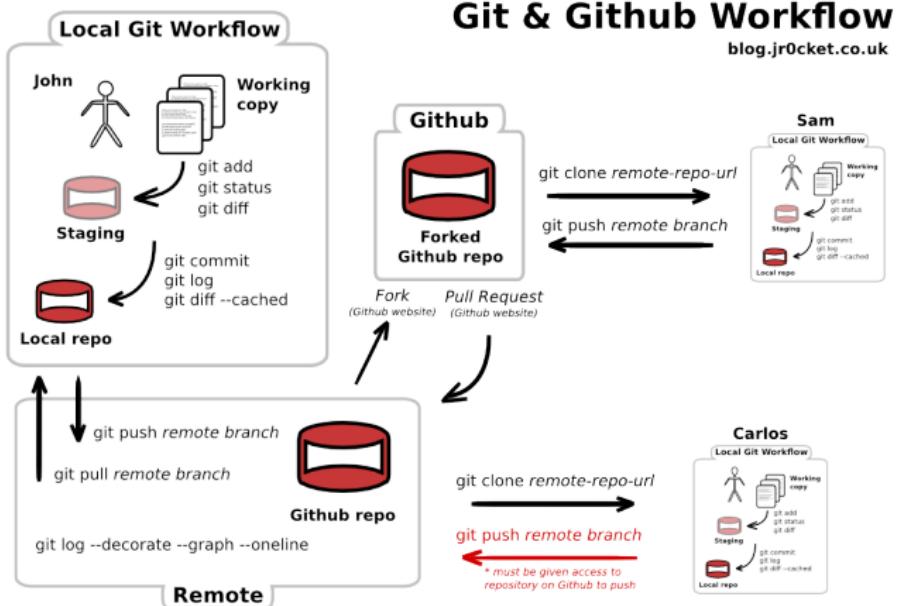
Branch principle





GIT more in details

And GitHub in all that ...





GIT more in details Usual commands

Local GIT - usual

- ▶ `git init`: initialize a git repository,
- ▶ `git add`: command to track change for a file,
- ▶ `git status`: get the status of the repository,
- ▶ `git diff`: see the difference inside the file,
- ▶ `git commit`: save the changes. It will be possible to come back on this node,
- ▶ `git checkout`: come back to a specific commit.

Local GIT - branching

- ▶ `git branch`: to create a new branch,
- ▶ `git checkout`: navigate to other branch,
- ▶ `git merge`: merge branches,
- ▶ `git cherry-pick`: select a specific commit for merging.



GIT more in details Usual commands

Remote GIT - synchronization

- ▶ `git remote add`: add the url of the server,
- ▶ `git push`: upload changes from local to remote,
- ▶ `git pull`: download changes from remote to local and apply changes,
- ▶ `git fetch`: download changes from remote,
- ▶ `git rebase`: apply changes downloaded from remote.

History

What is GIT?



GIT more in details



GIT in even more details



GIT in even more details

git init

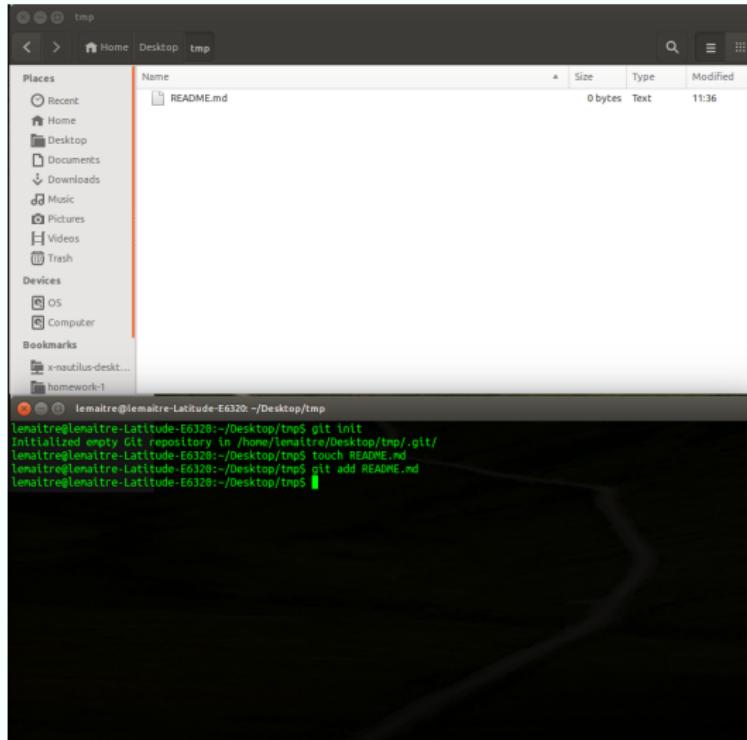
The screenshot shows a desktop environment with a file manager window open in the foreground, displaying a folder structure under 'Places' with 'tmp' selected. The terminal window in the background shows the command 'git init' being run in the directory '/Desktop/tmp'.

```
tmp
< > Home Desktop tmp
Places
Recent
Home
Desktop
Documents
Downloads
Music
Pictures
Videos
Trash
Devices
OS
Computer
Bookmarks
x-nautilus-deskt...
homework1
lennaire@lenaire-Latitude-E6320:~/Desktop/tmp
lennaire@lenaire-Latitude-E6320:~/Desktop/tmp$ git init
Initialized empty Git repository in /home/lennaire/Desktop/tmp/.git/
lennaire@lenaire-Latitude-E6320:~/Desktop/tmp$
```



GIT in even more details

git add



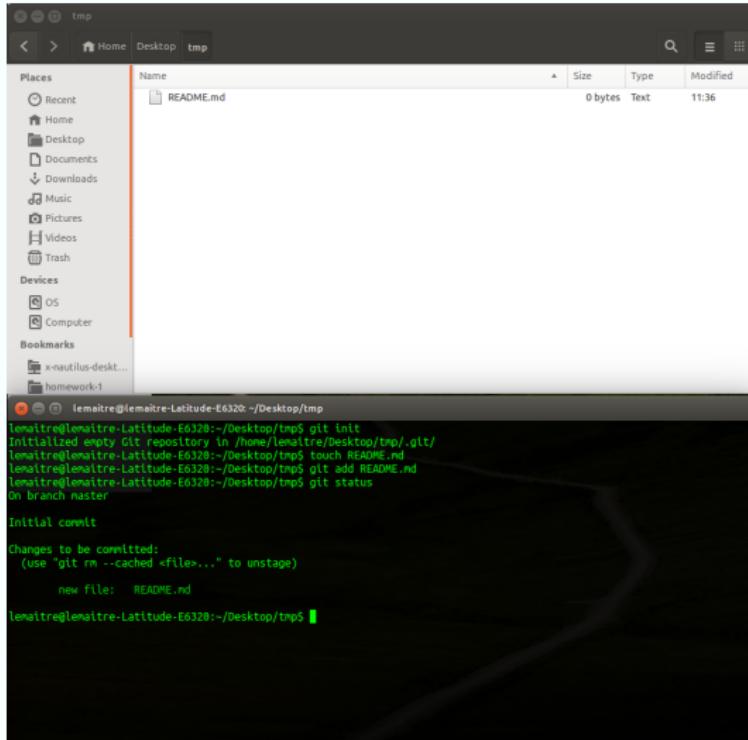
The screenshot shows a Linux desktop environment with a dark theme. A file manager window is open, displaying a single file named "README.md" in the "tmp" directory. The file is listed as a "Text" file with 0 bytes and was modified at 11:36. On the left side of the file manager, there is a sidebar with "Places" (Recent, Home, Desktop, Documents, Downloads, Music, Pictures, Videos, Trash), "Devices" (OS, Computer), and "Bookmarks" (x-nutilus-desktop..., homework-1). Below the file manager is a terminal window with the following text:

```
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ git init
Initialized empty Git repository in /home/lemaire/Desktop/tmp/.git/
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ touch README.md
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ git add README.md
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$
```



GIT in even more details

git status



The screenshot shows a desktop environment with a file manager window open, displaying a single file named "README.md" in the "/tmp" directory. The file is 0 bytes long and is a text file. Below the file manager is a terminal window with the following content:

```
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git init
Initialized empty Git repository in /home/lennaire/Desktop/tmp/.git/
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ touch README.md
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git add README.md
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

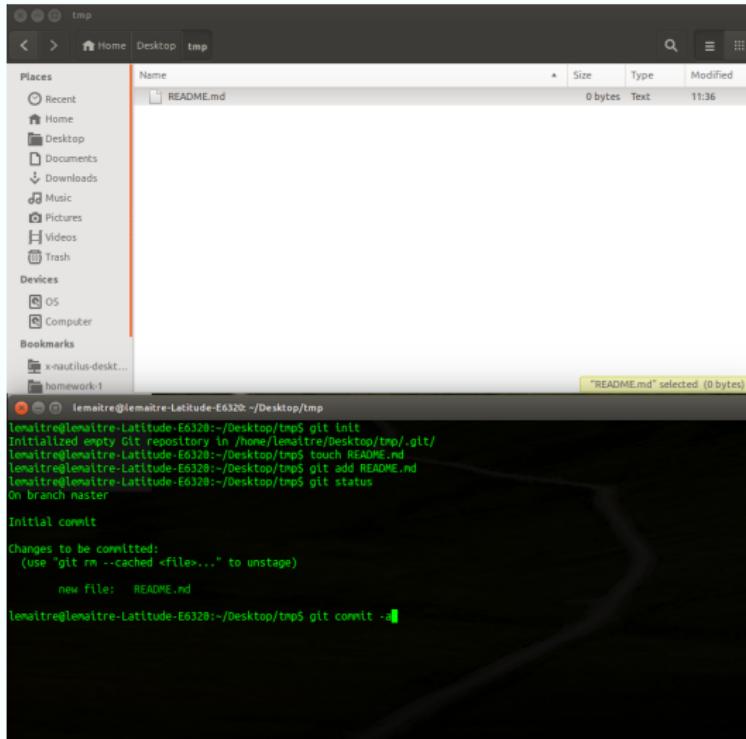
    new file:   README.md

lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$
```



GIT in even more details

git commit



The screenshot shows a desktop environment with a file manager window open in the foreground and a terminal window below it.

File Manager Window:

- Path: tmp
- Places:
 - Recent
 - Home
 - Desktop
 - Documents
 - Downloads
 - Music
 - Pictures
 - Videos
 - Trash
- Devices:
 - OS
 - Computer
- Bookmarks:
 - x-nautilus-deskt...
 - homework-1

Name	Size	Type	Modified
README.md	0 bytes	Text	11:36

"README.md" selected (0 bytes)

Terminal Window:

```
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git init
Initialized empty Git repository in /home/lennaire/Desktop/tmp/.git/
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ touch README.md
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git add README.md
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

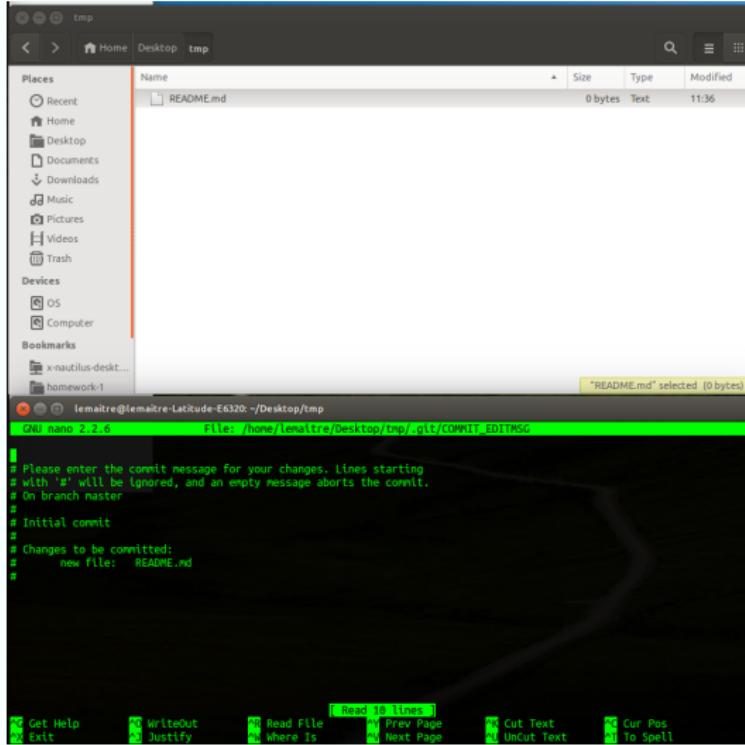
    new file:   README.md

lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git commit -m "Initial commit"
[master (root-commit) 8a2f3d0] Initial commit
  1 file changed, 0 insertions(+), 0 deletions(-)
  create mode 100644 README.md
```



GIT in even more details

git commit



The screenshot shows a Linux desktop environment with a dark theme. A file manager window is open, showing a single file named "README.md" in the "/tmp" directory. The terminal window below it is titled "File: /home/lenalitre/Desktop/tmp/.git/COMMIT_EDITMSG" and displays the nano 2.2.6 editor. The editor screen contains the standard git commit message template:

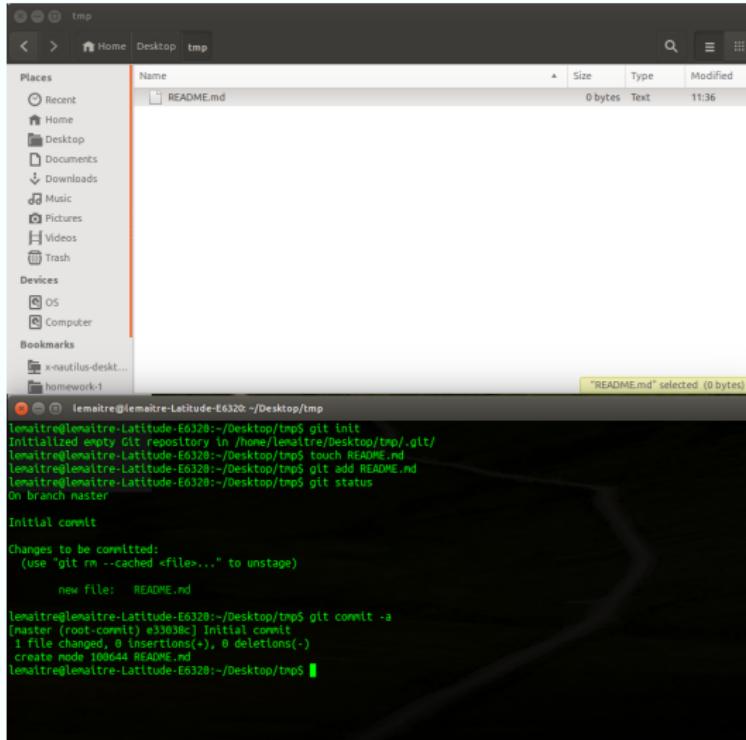
```
Please enter the commit message for your changes. Lines starting
with '#' will be ignored, and an empty message aborts the commit.
On branch master
Initial commit
Changes to be committed:
  new file: README.md
```

At the bottom of the terminal window, there are several keyboard shortcut keys displayed: Get Help, WriteOut, Read File, Where Is, Read 10 lines, Cut Text, Cur Pos, Exit, Justify, Prev Page, Next Page, Uncut Text, To Spell.



GIT in even more details

git commit



The screenshot shows a desktop environment with a file manager window open in the foreground, displaying a single file named "README.md" in the "/tmp" directory. The file is 0 bytes long and was modified at 11:36. The file manager's sidebar shows standard locations like Home, Desktop, Documents, Downloads, Music, Pictures, Videos, and Trash. Below the file manager is a terminal window with the following content:

```
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git init
Initialized empty Git repository in /home/lemaître/Desktop/tmp/.git/
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ touch README.md
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git add README.md
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

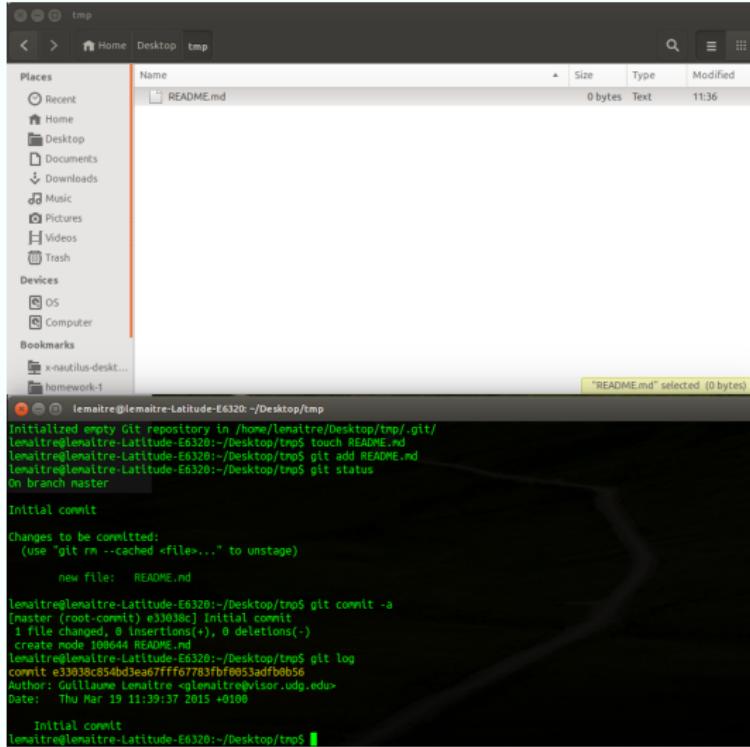
    new file:   README.md

lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git commit -a
[master (root-commit) e33838c] Initial commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$
```



GIT in even more details

git log



The screenshot shows a desktop environment with a file manager window open in the foreground and a terminal window in the background.

File Manager: The window title is "tmp". The sidebar shows "Places" with options like Recent, Home, Desktop, Documents, Downloads, Music, Pictures, Videos, and Trash. The main area shows a single file named "README.md" with the following details:

Name	Size	Type	Modified
README.md	0 bytes	Text	11:36

Terminal: The terminal window has the following content:

```
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ Initialized empty Git repository in /home/lemaire/Desktop/tmp/.git/
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ touch README.md
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ git add README.md
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

    new file: README.md

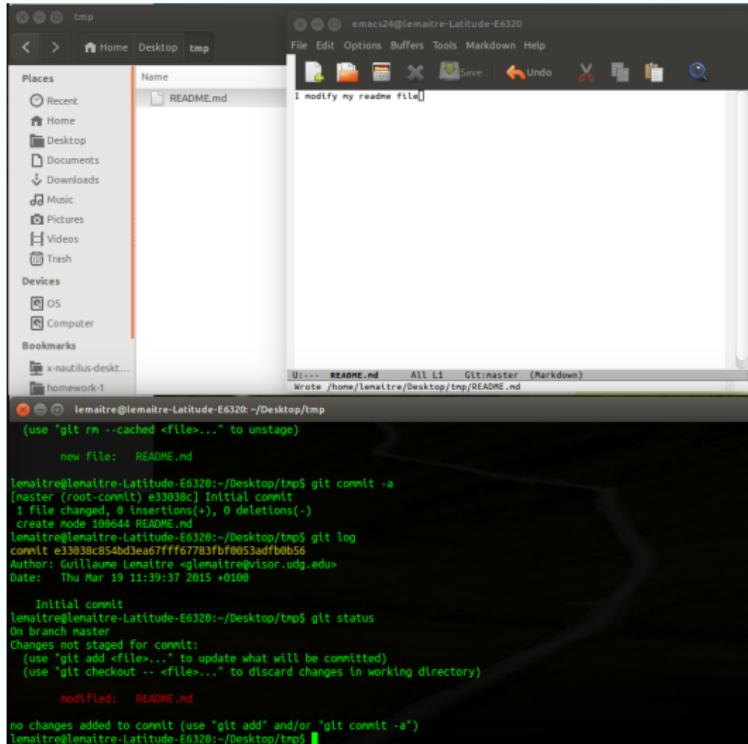
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ git commit -a
[master (root-commit) e33038c] Initial commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ git log
commit e33038c854bd3e067ff6f67783fb0053adfb0b56
Author: Guillaume Lemaire <glemaire@visor.udg.edu>
Date:   Thu Mar 19 11:39:37 2015 +0100

  Initial commit
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$
```



GIT in even more details

git status



The screenshot shows a Linux desktop environment with a dark theme. On the left, a file manager window titled 'tmp' is open, showing a single file named 'README.md'. The right side of the screen is occupied by a terminal window with the following content:

```
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git commit -a
[master (root-commit) e3303bc] Initial commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git log
commit e3303bc8c54bd3ea67ff67783fb0053adfb0bb56
Author: Guillaume Lemaître <gemaître@visor.udg.edu>
Date:   Thu Mar 19 11:39:37 2015 +0100

Initial commit
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git checkout -- <file>..." to discard changes in working directory)

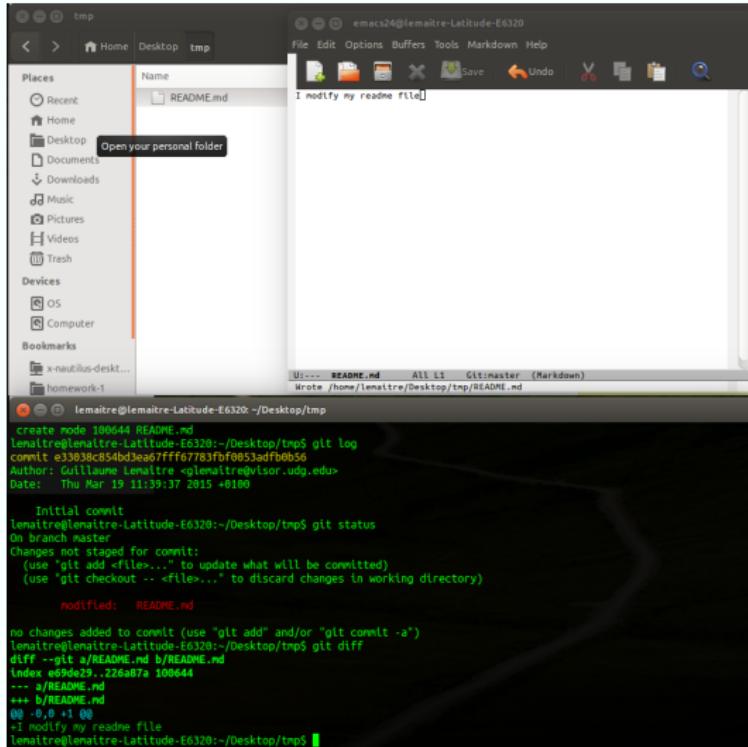
        modified: README.md

no changes added to commit (use "git add" and/or "git commit -a")
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$
```



GIT in even more details

git diff



```
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git add README.md
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git log
commit e38b38c854bd3e067fffe7783fb0853adfb0b056
Author: Guillaume Lennaire <glennaire@visor.udg.edu>
Date:   Thu Mar 19 11:35:37 2015 +0100

Initial commit
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git checkout -- <file>..." to discard changes in working directory)

        modified: README.md

no changes added to commit (use "git add" and/or "git commit -a")
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git diff
diff --git a/README.md b/README.md
index e69de2b..226a87a 100644
--- a/README.md
+++ b/README.md
@@ -0,0 +1 @@
+I modify my readme file
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$
```

History

What is GIT?



GIT more in details



GIT in even more details



GIT in even more details gitg

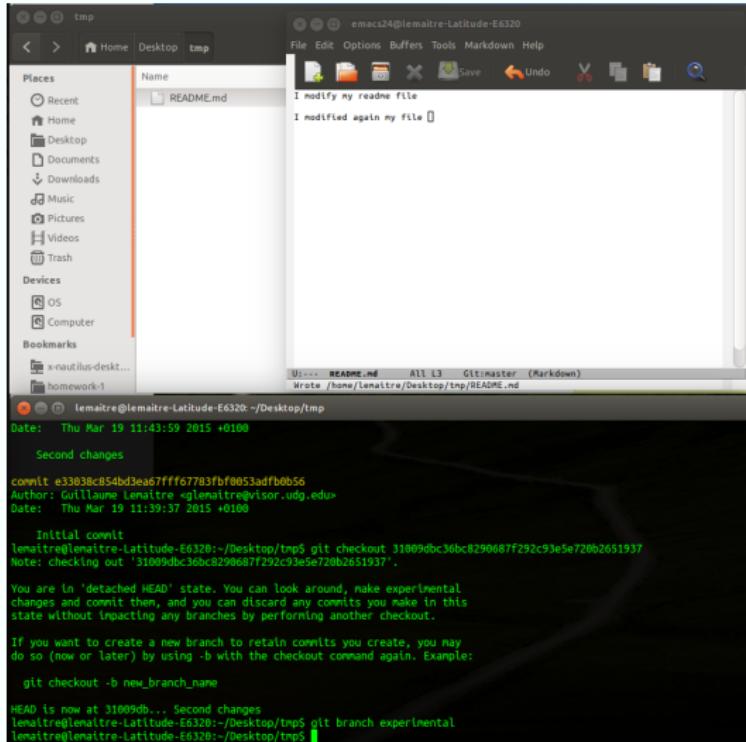
The screenshot shows a window titled "gitg - tmp (master)". It has tabs for "History" and "Commit". A dropdown menu "Branch: master" is open. A search bar with a magnifying glass icon is at the top right. The main area displays a table of commits:

Subject	Author	Date
master Second changes	Guillaume Lemaitre	jue 19 mar 2015 11:43:59 CET
Initial commit	Guillaume Lemaitre	jue 19 mar 2015 11:39:37 CET



GIT in even more details

git branch



The screenshot shows a Linux desktop environment with a file manager (Nautilus) and a terminal window.

In the file manager, a file named `README.md` is open, containing the text:

```
I modify my readme file  
I modified again my file
```

In the terminal window, the user is in a directory `/tmp`. They run `git status` and see:

```
Untracked files: README.md
```

Then they run `git branch` and see:

```
Second changes
```

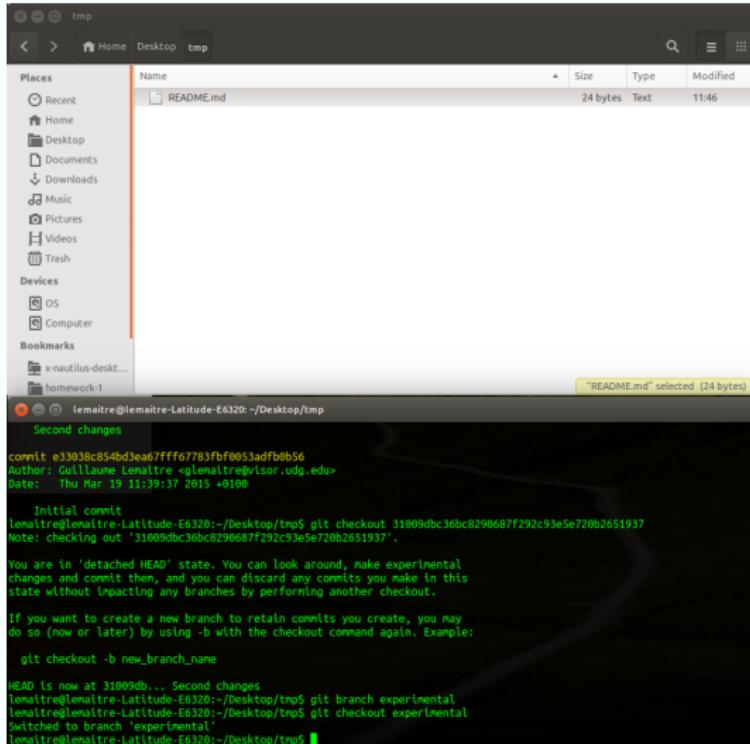
They then run `git checkout -b experimental` and see:

```
Initial commit  
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git checkout 31009dbc36bc8290687f292c93e5e720b2651937  
Note: checking out '31009dbc36bc8290687f292c93e5e720b2651937'.  
You are in 'detached HEAD' state. You can look around, make experimental  
changes and commit them, and you can discard any commits you make in this  
state without impacting any branches by performing another checkout.  
If you want to create a new branch to retain commits you create, you may  
do so (now or later) by using -b with the checkout command again. Example:  
  
git checkout -b new_branch_name  
  
HEAD is now at 31009db... Second changes  
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$ git branch experimental  
lennaire@lennaire-Latitude-E6320:~/Desktop/tmp$
```



GIT in even more details

git checkout



The screenshot shows a terminal window with the following content:

```
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ git checkout 31009dbc36bc8290687f292c93e5e720b2651937
Note: checking out '31009dbc36bc8290687f292c93e5e720b2651937'.
      Initial commit
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ git branch experimental
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$ git checkout experimental
Switched to branch 'experimental'
lemaire@lemaire-Latitude-E6320:~/Desktop/tmp$
```

The terminal shows the user performing a git checkout operation on a commit hash, which creates a new branch named 'experimental'. It also lists the current branch as 'experimental'.

History

What is GIT?
ooooo

GIT more in details
ooooo

GIT in even more details
oooooooooooo●oooooooo



GIT in even more details

A screenshot of a Git commit history interface. The window title is "gitg - tmp (experimental)". The tabs at the top are "History" and "Commit", with "History" selected. A dropdown menu "Branch: All branches" is open. The main area shows three commits:

Subject	Author	Date
[master] Third changes	Guillaume Lemaitre	jue 19 mar 2015 11:45:59 CET
[experimental] Second changes	Guillaume Lemaitre	jue 19 mar 2015 11:43:59 CET
Initial commit	Guillaume Lemaitre	jue 19 mar 2015 11:39:37 CET

History

What is GIT?



GIT more in details



GIT in even more details



GIT in even more details

The screenshot shows a Linux desktop environment with several windows open:

- A file manager window titled "tmp" showing a file named "README.md". The status bar indicates the path is "tmp".
- A terminal window titled "gitg - tmp (experimental)" showing a git commit history. The commits are:
 - experimental: Make changes in experimental (Author: Guillaume Lemaitre, Date: jue 19 mar 2015 11:49:37 CET)
 - master: Third changes (Author: Guillaume Lemaitre, Date: jue 19 mar 2015 11:45:59 CET)
 - Second changes (Author: Guillaume Lemaitre, Date: jue 19 mar 2015 11:43:59 CET)
 - Initial commit (Author: Guillaume Lemaitre, Date: jue 19 mar 2015 11:39:37 CET)
- gitk window showing the same commit history with detailed commit messages.

At the bottom of the gitk window, the status bar says "Loaded 4 revisions in 0.05s".



GIT in even more details

git merge

```
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

  git checkout -b new_branch_name

HEAD is now at 31009db... Second changes
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git branch experimental
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git checkout experimental
Switched to branch 'experimental'
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ gitg
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git commit -a
[experimental F54935d] Make changes in experimental
 1 file changed, 2 insertions(+)
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ gitg
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git merge --help
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git checkout master
Switched to branch 'master'
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$ git merge experimental
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
lemaître@lemaître-Latitude-E6320:~/Desktop/tmp$
```

History

What is GIT?



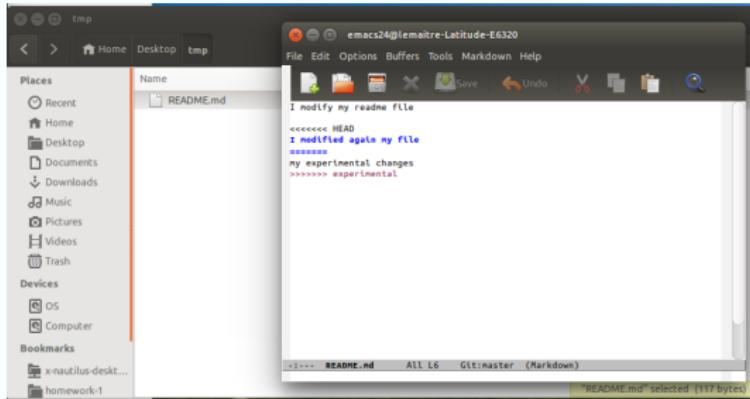
GIT more in details



GIT in even more details



GIT in even more details git merge





GIT in even more details

GitHub

Owner  glemaitre / Repository name 

Great repository names are short and memorable. Need inspiration? How about [animated-dubstep](#).

Description (optional)

 **Public**
Anyone can see this repository. You choose who can commit.

 **Private**
You choose who can see and commit to this repository.

Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None** | 

Create repository



GIT in even more details

git remote add

```
lémaitre@lemaitre-Latitude-E6320: ~/Desktop/tmp
git checkout -b new_branch_name

HEAD is now at 31009db... Second changes
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ git branch experimental
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ git checkout experimental
Switched to branch 'experimental'
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ git log
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ git commit -a
[experimental f54935d] Make changes in experimental
1 file changed, 2 insertions(+)
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ gitg
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ git merge --help
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ git checkout master
Switched to branch 'master'
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ git merge experimental
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ git commit -a
[master 5d1fb40] Merge branch 'experimental'
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ 
? Command not found
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$ git remote add origin https://github.com/glemaître/example-git.git
lémaitre@lemaitre-Latitude-E6320:~/Desktop/tmp$
```

History

What is GIT?



GIT more in details



GIT in even more details



GIT in even more details git push

```
lemaître@lemaître-Latitude-E6320: ~/Desktop/tmp$ git push origin master
When push.default is set to 'matching', git will push local branches
to the remote branches that already exist with the same name.

In Git 2.0, Git will default to the more conservative 'simple'
behavior, which only pushes the current branch to the corresponding
remote branch that 'git pull' uses to update the current branch.

See 'git help config' and search for 'push.default' for further information.
(The 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

Username for 'https://github.com': "C
lemaître@lemaître-Latitude-E6320: ~/Desktop/tmp$ git push origin master
Username for 'https://github.com': glenmaître
Password for 'https://glenmaître@github.com':
Counting objects: 13, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (13/13), 1.00 KIB | 0 bytes/s, done.
Total 13 (delta 2), reused 0 (delta 0)
To https://github.com/glenmaître/example-git
 ! [new branch]    master -> master
lemaître@lemaître-Latitude-E6320: ~/Desktop/tmp$
```

History

What is GIT?



GIT more in details



GIT in even more details



GIT in even more details git push

glemaître / example-git

Unwatch 1

Description Website

Short description of this repository Website for this repository (optional) Save or Cancel

5 commits 1 branch 0 releases 0 contributors

branch: master example-git / +

Merge branch 'experimental' ...

Guillaume Lemaître authored 4 minutes ago latest commit 5d1fb40d5c

README.md Make changes in experimental 9 minutes ago

README.md

I modify my readme file

my experimental changes