Agenda

- What is Git ?
- Installation of Git
- Git basis
- Github
- First steps with Git

WHAT IS GIT ?

What is Git ?

- Started in 2005
- Created by Linus Torvald
    - To aid Linux kernel development
- Most modern version control system
    - others
        - CVS
        - Subversion
        - Perforce
        - Mercurial

"Git is a [free and open source](https://git-scm.com/) distributed version control system designed to handle everything from small to very large projects with speed and efficiency."

https://git-scm.com/

redhat.

What do you store in Git ?

- Any type of files
- Ideally, files that will be modified and where you need to keep track of the modifications
    - Config files
    - Puppet receipt
    - Ansible playbook
    - Code

Companies & Projects Using Git

Google facebook Microsoft twitter Linked in. NETFLIX PostgreSQL

ANDROID RAILS Qt GNOME eclipse K X

redhat.

Why Git is so popular ?

- Nearly every operation is local
- Has integrity
  - Everything in Git is check-summed before it is stored
- Git generally only adds data

INSTALLATION OF GIT

Git installation

RHEL/CENTOS/FEDORA

```
# yum install git
```

WINDOWS

https://git-scm.com/download/win

MAC

```
# brew install git
```

GIT BASIS

redhat.

GIT key concepts 1/2

- Snapshots
  - Takes a picture (reference) of what all you files look like at that moment
- Commit
  - Act of creating a snapshot
    - Contains 3 informations
      - How the files changed from previously
      - A reference to the commit that came before it
      - A hash code name (6aasdfdsfaasdf34..)
- Repositories (repo)
  - Collection of all the files and the history
    - Consist of all your commits
    - Can live on a local machine or remote (github)
    - The act of copying a repository is called **cloning**

redhat.

GIT key concepts 2/2

- Branch
  - All commit in git live in some branch
  - The main branch in a project is called the master branch (doesn't have to)
- Head
  - A reference to the most recent commit (almost always)
- Origin
  - Git has the concept of "remotes", which are simply URLs to other copies of your repository.
    - When you clone another repository, git automatically creates a remote named "origin" and points to it

The three states

- Modified
  - File changed but not committed
- Staged
  - Marked a modified file
- Committed
  - Data is safely stored
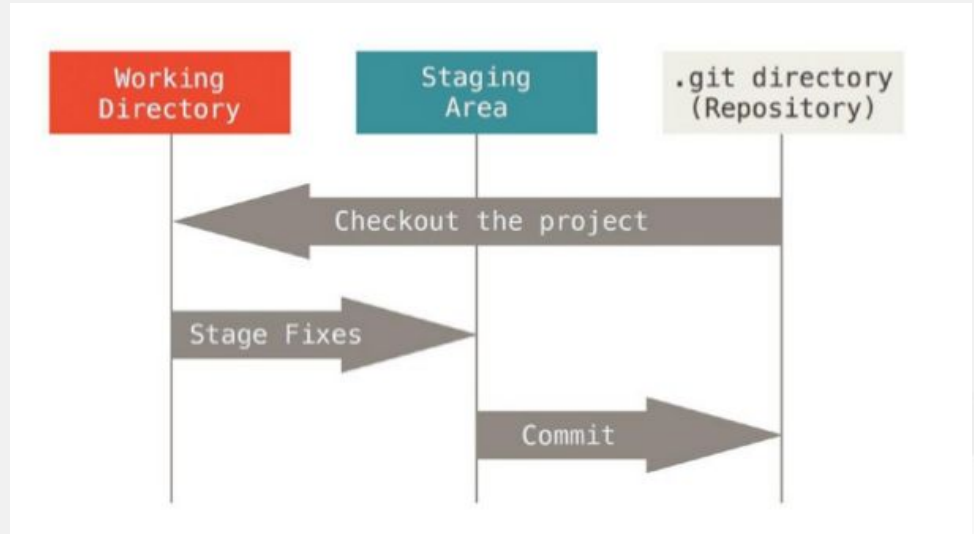
Basic Git workflow

1. You modify files in your working directory.

2. You stage the files, adding snapshots of them to your staging area.

3. You do a commit, which takes the files as they are in the staging area and stores that snapshot permanently to your Git directory.

GITHUB

What is Github ?

- www.github.com
- Largest web-based git repository
- Allow for code collaboration
- Add some extra functionality on top of git
- Funded in 2008
- Enterprise edition available
  - Alternatives : Gitlab, Bitbucket, ...
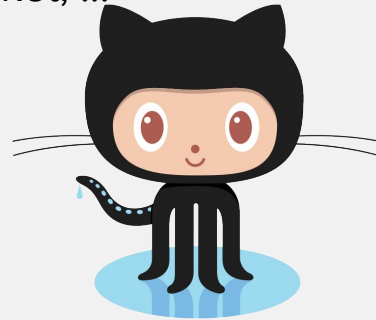
Most active projects on Github

2016 : https://octoverse.github.com/

Repositories with the most open source contributors

FortAwesome/Font-Awesome		10,654
docker/docker		8,253
npm/npm		7,041
jlord/patchwork		6,806
facebook/react-native		6,250
Microsoft/vscode		5,855
atom/atom		5,745
FreeCodeCamp/FreeCodeCamp		5,622
angular/material		4,355
angular/angular		4,217

Organizations with the most open source contributors

Microsoft		16,419
facebook		15,682
docker		14,059
angular		12,841
google		12,140
atom		9,698
FortAwesome		9,617
elastic		7,220
Apache		6,999
npm		6,815

redhat.

LAB - Create an account on Github

- Go to http://www.github.com and create an account by clicking the 'Sign up' button on the top right.
- Create a Repository (project)
- Click on readme than commit the code
- Click on Clone/Download URL, select use https
- Copy the link

LAB - FIRST STEPS WITH GIT

First steps with git 1/6

Initial configuration of git

```
[mlessard@laptop ~]$ git config --global user.name "michael.lessard"
[mlessard@laptop ~]$ git config --global user.email "mlessard@redhat.com"
[mlessard@laptop ~]$ git config –global push.default simple  (avoid to specify the branch)

[mlessard@laptop ~]$ git config --list
```

redhat.

First steps with git 2/6

Let's **clone** the project that we just created on github

```
[mlessard@laptop ~]$ mkdir projects ; cd projects
[mlessard@laptop projects]$ git clone https://github.com/michaellessard/testrhug.git
Cloning into 'testrhug'...
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.

[mlessard@laptop projects]$ ls
testrhug

[mlessard@laptop projects]cd testrhug
[mlessard@laptop testrhug]$ ls
README.md
```

redhat.

First steps with git 3/6

Check the status

```
[mlessard@laptop testrhug]$ git status
# On branch master
nothing to commit, working directory clean
```

Add a file (stage the file)

```
[mlessard@laptop testrhug]$ touch mycode.txt
[mlessard@laptop testrhug]$ git status
On branch master
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       mycode.txt
nothing added to commit but untracked files present (use "git add" to track)
[mlessard@laptop testrhug]$ git add mycode.txt
```

First steps with git 4/6

Commit your change (store the change locally)

```
[mlessard@laptop testrhug]$ git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#     new file:   mycode.txt
#
[mlessard@laptop testrhug]$ git commit -m "first commited file"
[master 7c2a292] first commited file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 mycode.txt
```

At this point, the new file is committed to the project in my laptop, but not
synced up.

First steps with git 5/6

Push your change to the remote project

```
[mlessard@laptop testrhug]$ git push

Counting objects: 4, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 281 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/michaellessard/testrhug.git
   6145d4e..7c2a292  master -> master
```

First steps with git 6/6

Everyone else who wants to have the latest revision of your project must :
  - execute a "git pull" if they already cloned the project or
  - the "git clone URL" command to download the whole project

Let's now see the result on github.com

Some tips and tricks

If you want to stage multiple files :   `$ git add .`

If you want to see the changelog of your project :   `$ git log`
                    -p, which shows the difference introduced in each commit

If you want to cache your github credential :

`$ git config --global credential.helper 'store --file ~/.my-credentials'`

redhat.