

Software forges for git

Sylvain Bouveret, Grégory Mounié, Matthieu Moy

2021

[first].[last]@imag.fr

<https://git.pages.ensimag.fr/formation-git/slides/git-forges-handout.pdf>

pdf



Grenoble
INP
UCA
Ensimag

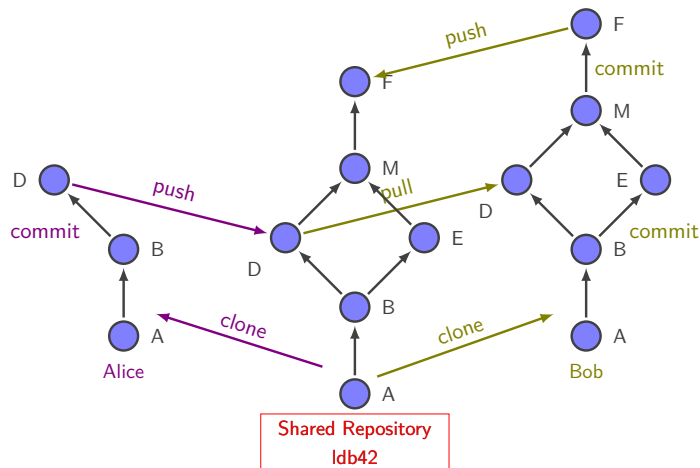
Why forges?

Working together with git...

Being alone is nice...

Working together is better.

We need a way to **share projects**.



Grenoble
ENSIMAG
INP
UCA

Basic git hosting

Grenoble
ENSIMAG
INP
UCA

Basically, to work together, you only need basic git hosting features:

1. file hosting
2. file exchange (using ssh and/or https)
3. user management and authentication

For instance: Gitolite (<https://gitolite.com/>)

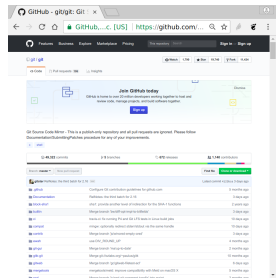
- lightweight git hosting system
- communication / authentication based on ssh
- user managements (with ssh keys) based on git itself!

Basic git hosting is enough for small projects. But for bigger ones, better use a software forge:

- Bug tracking (issues)
- Branches management
- Fine-grained roles
- External contributions with pull requests
- Code reviewing tools
- Continuous integration / deployment
- Online edition
- Web pages and documentation hosting
- ...

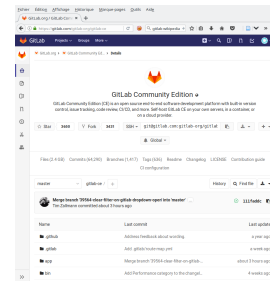
Github and Gitlab

Github: historically THE git platform



GitHub is a git server and a dedicated web site. 75 million repo. 25 million users. **Developer centric**. Each repository gets README and documentation publication, issue tracking, pull request, wiki, small websites, on-line edition, etc.

Gitlab: now the MUST-HAVE git platform



Gitlab provides similar functionalities. Free private repositories at <http://gitlab.com>. It is **Open Source**. Easy to deploy at enterprise level (incl. LDAP) (eg. <http://gricad-gitlab.u-ga.fr>, <http://gitlab.ensimag.fr>).

Starting with Gitlab

First, configure your profile

After you login to Gitlab, a good idea is to start configuring your profile, especially your SSH key.

1. If you don't have one, create your own SSH key using `ssh-keygen`
2. Upload your public key to your gitlab account (User profile ▷ SSH Keys)
3. Now you can clone / push / pull over ssh (without having to type your password)

Then, create a project

Once your profile is configured, you can create a project:

1. Click on the '+' button at the top of the page
2. Choose 'new project'
3. Complete the parameters and create the project

Once your project is created:

- Either clone it from your computer (`git clone...`)
- Or go to an existing repo on your computer, configure the remote to point to your Gitlab project (see later), and push your local changes.