COMET LiCS

CEMAC

cemac-support@leeds.ac.uk

CONTENTS

Notes regarding the COMET LiCS system.

CHAPTER

ONE

SOFTWARE VERSION CONTROL

The GitLab service is currently used for hosting Git repositories related to the operation of the COMET LiCS system.

1.1 Current GitLab Usage

The COMET LiCS related material is stored within a GitLab *group* at https://gitlab.com/comet_licsar.

There are currently 11 repositories, 5 of which are publicly visible.

Several of the repositories have not been updated in some time (one year or more), but this may be due to the content not requiring regular updates.

The licsar_proc, licsar_framebatch and licsar_documentation repositories appear to be the most active and frequently updated.

The *licsar_documentation* repository makes use of the GitLab wiki functionality. This allows the documentation to be created in markdown format, and rendered by the GitLab web interface.

1.2 Git Hosting Options

There are various options available for hosting Git repositories, with GitHub and GitLab being the most popular services available, and the two which we will consider here.

Both services have been around for several years, and can be considered stable and reliable. GitHub is the most well known and popular of the two services, and is owned by Microsoft.

1.3 Feature Comparison

The features offered at the non charged level of hosting by GitHub and GitLab are both very similar.

- GitLab pricing information
- Github pricing information

In the past GitLab allowed private repositories at the non charged level, where as GitHub did not, and this could have been one of the main reasons why GitLab was originally chosen for the COMET LiCS repositories.

Where GitLab allows repositories to be organised in groups, GitHub uses organisations, for example, the Met Office GitHub organisation.

GitLab and Github both have educational programs:

- https://about.gitlab.com/solutions/education/
- https://docs.github.com/en/education/

These services allow an individual to apply to join the educational program (GitLab Education, GitHub Education), to access features which usually incur a charge.

We do not have any experience with the GitLab offering, but there are several members of the School of Earth and Environment in Leeds who have applied successfully for the GitHub education program. The main benefit is that this allows the individual to upgrade an organisation to the *Team* level, which adds some additional features, mainly to the functionality of private repositories.

1.4 Hosting Considerations

As there is not a huge number of repositories currently located within the COMET LiCS GitLab group, and these would be fairly simple to migrate to GitHub, I believe it would be worth seriously consider migrating to GitHub.

1.4.1 Familiarity

As GitHub is the most popular Git hosting service, this the service which is most likely to be familiar to new members of the group, so there is a lower barrier to entry.

Git training (for example using the Software Carpentry Git Lessons) may also include specific sections of working with GitHub.

GitLab works in a reasonably similar way to GitHub, and it is even possible to log in to GitLab with a GitHub account, but the requirement to use an 'unfamiliar' system can often prove a significant barrier to engagement.

1.4.2 Working With Existing Projects

Some projects of significant interest to the COMET LiCS group, such as LiCSBAS and ISCE are hosted on GitHub.

Working on GitHub would make it easier to work with the code hosted in these repositories.

For example, there is a fork of the LiCSBAS repository hosted in the COMET LiCS GitLab group. If this was hosted on GitLab, it would be possible to track the differences between this fork and the original repository via the GitHub web interface, as well as easily contribute changes back to the upstream source.

1.4.3 Integration With Third Party Tools

Many third party tools which integrate with Git repositories will integrate with GitLab repositories just as easily as as GitHub repositories, such as Binder and Read The Docs.

One tool which could be of significant interest to the group, and integrates easily with GitHub is Zenodo, which allows creation a copy of the content of a repository at a particular release, with a DOI, making the code easily citable.

There is some more information here, and a simple example of a repository which has been archived with Zenodo can be found at https://doi.org/10.5281/zenodo.5997119.

Each time a new release is created in a repository which is linked to Zenodo, a new DOI is created, where there is also an overarching DOI which always links to the most recent release.

1.4.4 Restricted GitLab Functionality Without Verification

Possibly the largest issue with GitLab is the inability to use any of the CI / CD (continuous integration / continuous deployment) features without first verifying your identity by providing a credit card.

This means it is not possible to use features such as the Pages (GitLab Pages, GitHub Pages) feature to easily host web content, which can be extremely useful.

It also means it is not possible to enable functionality such as automated testing of code, when changes are pushed to a repository.

The Pages feature on GitHub is much more simple to use and can be enabled with a couple of clicks in the repository settings page. Enabling the Pages feature is more complicated on GitLab, which may also be off putting to individuals who want a simple way to host web content.

1.5 Future Use Of Hosting Services

There are opportunities for increased use of features offered by services such as GitLab and GitHub which could benefit the COMET LiCS code and its users.

This may include automated documentation building and deployment, or automated testing of code when changes are pushed to a Git repository.

Some of these things may have an initial time cost, for example creating tests for the software, setting up the tasks to run the tests, and so on, but he benefits may be significant in the long run.

It may also be worth considering how Git tags, software versioning and releases might be used effectively by the COMET LiCS project. For example, should there be a version of the software which directly corresponds to a version of the data set?

Any additional software used as part of the processing, and scripts used to build these tools could also be stored in Git repositories, and this may allow easier duplication of the processing environment of different systems.