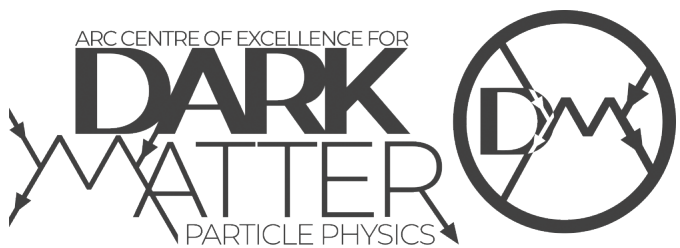
Practical 2: Collaborative Git

CDM Computing Subgroup Workshop

Albert Kong

THE UNIVERSITY of ADELAIDE



February, 2024

Forking Workflow

Very often used in public open-source projects, but can be used in private projects too.

- Start by creating a fork of a repository on GitHub, eg this workshop's repository
- Clone your fork and make/commit some changes on a new branch
- Push your changes to your fork and open a pull request on GitHub

Merge Conflicts

Git can usually figure out how to merge two sets of changes to the same file as long as they modify different areas of the file.
If both commits modify the same area then the conflict will need to be resolved manually:

- Begin the merging process with `git merge <branch>`
- Check which files have conflicts with `git status`
- Open a conflicting file in your editor and search for sections marked by `<<<<<<<`, `=======` and `>>>>>>>`
- Stage the file after resolving the conflicts
- When all conflicting files have been resolved, commit to finalise the merge

Submodules

Sometimes you want to include another Git repository within your own, eg as a dependency.

- To add an existing repository as a submodule, use `git submodule add <url>`
- After cloning a repository with submodules, use `git submodule init` followed by `git submodule update` to fetch and checkout the appropriate commits for all submodules
- Alternatively, add the `--recurse-submodules` option when using `git clone` to achieve the same result