



THE UNIVERSITY *of* EDINBURGH

---

# Using Git and GitHub for Team Collaboration

Fei Yao  
28/02/2022

---

<https://feiyao-edinburgh.github.io/>

# Git and GitHub

---

- Git and GitHub are not the same thing.
  - Git is an open-source, version control tool created in 2005 by developers working on the Linux operating system;
  - GitHub is a company founded in 2008 that makes tools which integrated git.
- You do not need GitHub to use git, but you cannot use GitHub without using git.
- Alternatives to GitHub include GitLab, BitBucket, and etc. All of these are referred, in git-speak, as “remotes” that will make sharing code with others easier.

# An example workflow

- The project leader creates a repository (repo).
  - Add a README file to set main branch as the default branch.
  - Can modify the README contents later on.
  - Add .gitignore and the licence later on.

The screenshot shows a GitHub repository page. At the top, there are buttons for 'main' (selected), '1 branch', and '0 tags'. Below this, the repository name 'geofyao Initial commit' is displayed, along with the commit hash '753d6c4' and 'now'. A table lists the files: 'README.md' with 'Initial commit' and 'now'. The main content area shows the 'README.md' file with the title 'methane-emission-service' and the description 'Methane Emission Service Development'. A red box highlights the 'main' branch button and the '1 branch' button.

## Create a new repository

A repository contains all project files, including the revision history. Already have a project? [Import a repository.](#)

Owner \*



geofyao

Repository name \*

methane-emission-service



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

Description (optional)

Methane Emission Service Development



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

This will set `main` as the default branch. Change the default name in your [settings](#).

Create repository

# An example workflow

- The project leader adds collaborators to the repo.
  - The collaborators have push access to the repo.

The screenshot shows the GitHub repository settings page. The 'Settings' tab is selected in the top navigation bar. In the left sidebar, the 'Collaborators' link under the 'Access' section is highlighted with a red box. A red arrow points from this box to the 'Who has access' section. Another red arrow points from the 'Manage access' section to the 'Add people' button. The 'Who has access' section shows 'PRIVATE REPOSITORY' and 'DIRECT ACCESS' tabs, both indicating no collaborators. The 'Manage access' section displays the message 'You haven't invited any collaborators yet' and a green 'Add people' button.

## Who has access

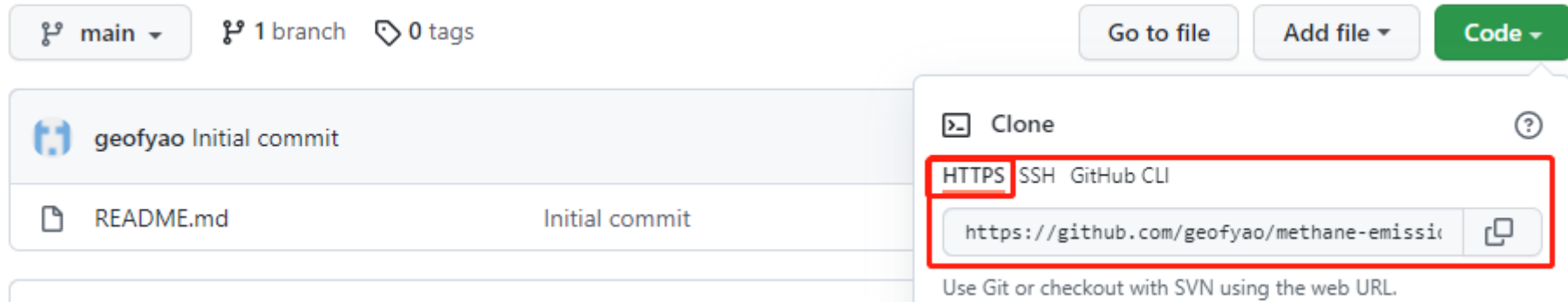
This section details the repository access settings. The 'PRIVATE REPOSITORY' tab is active, showing that only those with access can view it, with a 'Manage' link. The 'DIRECT ACCESS' tab shows that 1 collaborator has access to the repository.

## Manage access

This section shows the interface for managing repository access. It includes a 'Select all' checkbox, a search bar labeled 'Find a collaborator...', and a list of collaborators. One collaborator, 'Fei Yao' (FeiYao-Edinburgh), is listed with a role of 'Collaborator'.

# An example workflow

- The collaborators (including the repo owner) clone the repo to their local machines.
  - As collaborators, they can type their own username and password when prompted to do so.
  - HTTPS requires username and personal access token (PAT).
  - SSH may require SSH keys (similar to JASMIN although not tested).
  - `cd`
  - `git clone https://github.com/geofyao/methane-emission-service.git`



# An example workflow

---

- The collaborators always create a new branch where they will develop codes **without affecting the main branch.**
  - `cd methane-emission-service`
  - `git remote -v # show remote URL.`
  - `git branch # See that you are in the main branch.`
  - `git branch feature/radiative # Better to use a branch name indicating what the collaborators are working.`
  - `git checkout feature/radiative # Switch to the new branch.`
  - `# Can combine the above two lines of codes to "git checkout -b feature/radiative".`
  - `mkdir -p radiative # Now make some changes under the new branch.`
  - `cd radiative`
  - `touch README.md # Each subdir can have its own README.md.`

# The collaborators push their branch to the remote

---

- Track changes: add to the staging environment => make commits.
  - `git status` # Git has noted the changes but won't track it until we explicitly tells it to.
  - `git add --all` # Or can add one by one.
  - `git status` # Git has added the changes to the staging environment but not a commit although it is about to.
  - `git commit -m 'First commit.'` # Package all files that have been added to the staging environment to a commit.
  - `git log` # To see the commit.
  - `git log origin/main..HEAD` # See un-pushed commits if necessary.
  - # `origin` is an alias of the remote repo. `HEAD` can be thought as the current branch.
  - `git pull origin feature/radiative` # In case other commits are made to this branch during the period.
  - `git push origin feature/radiative` # Package all commits that you'd like to push to the remote.

# An example workflow

- The collaborators make a pull request (PR) from their GitHub page.

The screenshot shows the GitHub interface for the repository `geofyao / methane-emission-service`. The repository is marked as `Private`. The navigation bar includes links for `Code`, `Issues`, `Pull requests`, `Actions`, `Projects`, `Security`, and `Insights`. The repository statistics show `1` Watch, `0` Fork, and `0` Star.

The main content area is titled `Open a pull request` and includes the instruction: "Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#)."

The branch selection area shows `base: main` and `compare: feature/radiative`, with a green checkmark indicating `✓ Able to merge. These branches can be automatically merged.`

The `Compare & pull request` button is highlighted in a yellow box. Below it, the `First commit.` section is highlighted in a red box, showing the `Write` and `Preview` tabs, a text area for `Leave a comment`, and a `Create pull request` button.

The right sidebar contains the `About` section for `Methane Emission Service Development`, showing `0` stars, `1` watching, and `0` forks. It also includes `Releases` and `Packages` sections, both indicating no published items.



# An example workflow

- Other collaborators will be notified about the PR.

## [geofyao/methane-emission-service] First commit. (PR #1)

From: "Fei Yao" <notifications@github.com> (Forward by noreply@github.com)  
To: "geofyao/methane-emission-service" <methane-emission-service@noreply.github.com>  
Cc: "Subscribed" <subscribed@noreply.github.com>

You can view, comment on, or merge this pull request online at:

<https://github.com/geofyao/methane-emission-service/pull/1>

### Commit Summary

- [7284bf9](#) First commit.


### File Changes (1 file)

- A [radiative/README.md](#) (0)

### Patch Links:

- <https://github.com/geofyao/methane-emission-service/pull/1.patch>
- <https://github.com/geofyao/methane-emission-service/pull/1.diff>

Reply to this email directly, [view it on GitHub](#), or [unsubscribe](#).  
Triage notifications on the go with GitHub Mobile for [iOS](#) or [Android](#).  
You are receiving this because you are subscribed to this thread.


 Pull requests 1

Actions Projects Security Insights Settings


## First commit. #1

Open FeiYao-Edinburgh wants to merge 1 commit into `main` from `feature/radiative`


Conversation 0 Commits 1 Checks 0 Files changed 1


 FeiYao-Edinburgh commented 4 minutes ago

No description provided.

 First commit.

Add more commits by pushing to the `feature/radiative` branch on `geofyao/methane-emission-service`.

 Continuous integration has not been set up  
GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.

 This branch has no conflicts with the base branch  
Merging can be performed automatically.

Merge pull request You can also open this in GitHub Desktop or view command line instructions.

Write Preview H B I ≡ <> ↻ ≡

# An example workflow

- Other collaborators review the PR and may make additional changes.

First commit. #1

[Open](#) FeiYao-Edinburgh wants to merge 1 commit into `main` from `feature/radiative`

Conversation 0 Commits 1 Checks 0 **Files changed 1** +0 -0

Changes from all commits File filter Conversations Jump to

0 radiative/README.md

Empty file.

0 / 1 files viewed [Review changes](#)

Finish your review

Write Preview H B I ≡ <> 🔗 ≡ ≡ ☑ @ 📎 ↶

Add some words to README.md.

Attach files by dragging & dropping, selecting or pasting them.

☐ Comment  
Submit general feedback without explicit approval.

☐ Approve  
Submit feedback and approve merging these changes.

☒ **Request changes**  
Submit feedback that must be addressed before merging.

[Submit review](#)

# An example workflow

- The collaborators make additional changes per request and re-push.

- `vi README.md`
- `git add --all`
- `git commit -m 'Second commit.'`
- `# git log origin.main..HEAD`
- `git pull origin feature/radiative`
- `git push origin feature/radiative`

[geofyao/methane-emission-service] First commit. (PR #1)



Some content in this message has been blocked because the sender isn't in your Safe senders list.



geofyao <notifications@github.com>

Sat 26/02/2022 12:20

To: geofyao/methane-emission-service <methane-emission-service@noreply.github.com>

Cc: YAO Fei; Author <author@noreply.github.com>

**This email was sent to you by someone outside the University.**

You should only click on links or attachments if you are certain that the email is genuine.

@geofyao requested changes on this pull request.

Add some words to README.md.

—

Reply to this email directly, [view it on GitHub](#), or [unsubscribe](#).

Triage notifications on the go with GitHub Mobile for [iOS](#) or [Android](#).

You are receiving this because you authored the thread.

[Reply](#)

[Reply all](#)

[Forward](#)

# An example workflow

- Other collaborators will be notified again to remind them to review and may approve the additional changes made to the previous PR.

Re: [geofyao/methane-emission-service] First commit

From: "Fei Yao" <notifications@github.com> (Forward by noreply@github.com)

To: "geofyao/methane-emission-service" <methane-emission-service@norep

Cc: "姚飞" <yaofei@pku.edu.cn> "Push" <push@noreply.github.com>

@FeiYao-Rdinhursh pushed 1 commit.

from 1 commit File filter Conversations Jump to

radiative/README.md

@@ -0,0 +1 @@

+ This is the radiative module of the methane project.

2 GitHub, Inc. Terms Privacy Security Status Docs



geofyao requested changes 18 minutes ago

[View changes](#)

geofyao left a comment



Add some words to README.md.



New changes since you last viewed

[View changes](#)



Second commit.

3b67e91

Finish your review

Write

Preview

H B I

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

☐ Comment

Submit general feedback without explicit approval.

☒ Approve

Submit feedback and approve merging these changes.

[geofyao/methane-emission-service] First commit. (PR #1)



Some content in this message has been blocked because the sender isn't in your Safe senders list. I



geofyao <notifications@github.com>

Sat 26/02/2022 12:30

To: geofyao/methane-emission-service <methane-emission-service@noreply.github.com>

Cc: YAO Fei; Author <author@noreply.github.com>

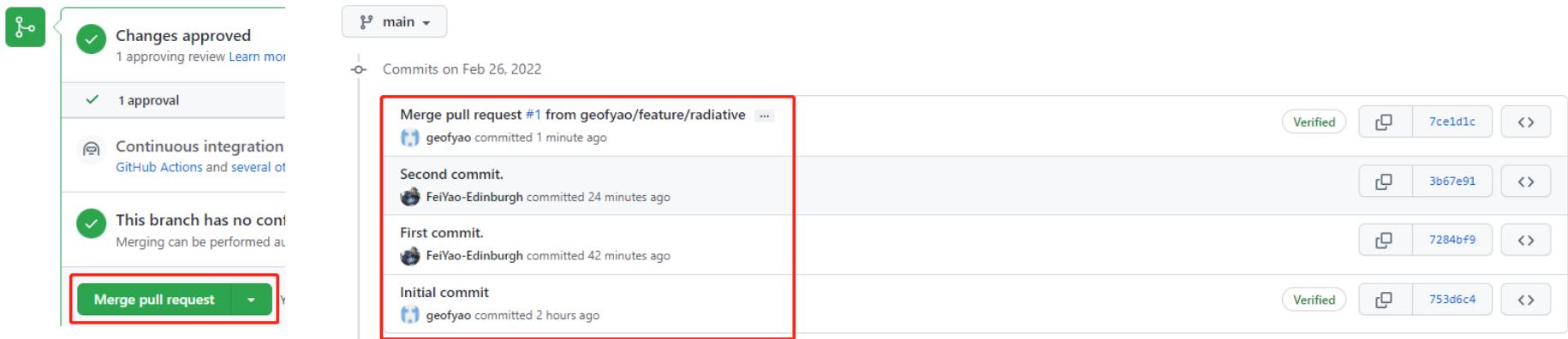
**This email was sent to you by someone outside the University.**

You should only click on links or attachments if you are certain that the email is genuine a

@geofyao approved this pull request.

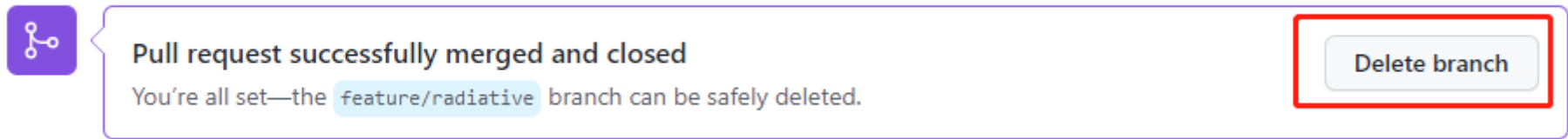
# An example workflow

- All collaborators agree with the working branch to be merged to main branch and subsequently delete the working branch.



The screenshot displays the GitHub pull request interface. On the left sidebar, the 'Merge pull request' button is highlighted with a red box. The main area shows the commit history for the 'main' branch, with a red box highlighting the merge action. The commit history includes:

- Merge pull request #1 from geofyao/feature/radiative (Verified, 7ce1d1c)
- Second commit. (FeiYao-Edinburgh committed 24 minutes ago, 3b67e91)
- First commit. (FeiYao-Edinburgh committed 42 minutes ago, 7284bf9)
- Initial commit (Verified, 753d6c4)



The screenshot shows a notification box with the text: 'Pull request successfully merged and closed. You're all set—the feature/radiative branch can be safely deleted.' The 'Delete branch' button is highlighted with a red box.

# An example workflow

---

- The collaborators sync the remote changes back to their local machines and start next run. All collaborators behave similarly.
  - `cd ..` # After switch to main radiative no longer exists.
  - `git checkout main` # Always know where you are before running git-related codes.
  - `git pull origin main` # Keep update with remote. Or may `git merge feature/radiative` in main branch.
  - `git log` # To see completely identical as remote.
  - `git branch -d feature/radiative` # Remove already done branches just like the remote operation.
  - `git checkout -b new-branch` # New branch to work on.

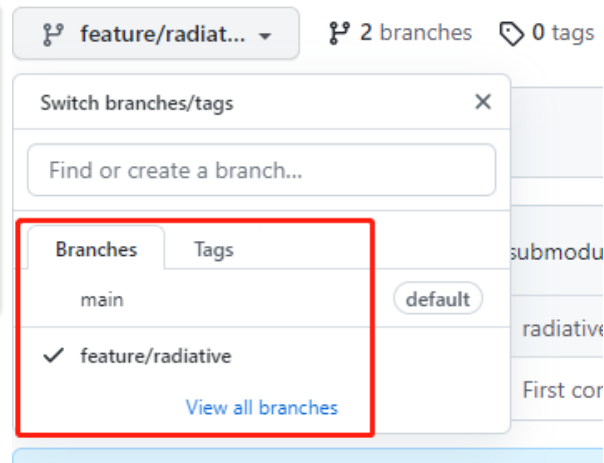
# Recap

- A common git and GitHub collaboration workflow:

- Fetch and merge changes from the remote.
- Create a branch to work on a new project feature.
- Develop the feature on the branch and commit the work.
- Fetch and merge from remote again (in case new commits were made).
- Push branch up to the remote for review.

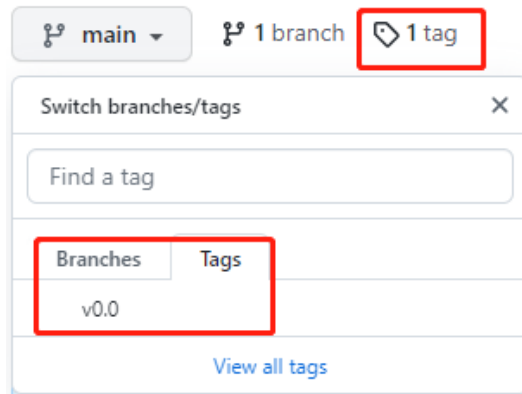
- A safeguard against merge conflicts:

- Collaborators work on separate subdirs.
- They negotiate when have to work on common files.



# Tags: keeping track of project milestones

- A tag is similar to a branch; it's simply a lightweight moveable pointer to one of the commits.
  - `git tag --list`
  - `git tag -a v0.0 -m "The 0.0 version."`
  - `git show v0.0`
  - `git push origin v0.0` # Always push code first.
  - # Create and checkout a new branch "feature/xxx" at tag "v0.0".
  - `git checkout -b feature/xxx v0.0`
  - `git tag -d v0.0` # Can delete unwanted local tags.
  - `git push origin --delete v0.0` # Can delete unwanted remote tags.





# Release

- Releases, based on tags, are deployable software iterations you can package and make available for a wider audience to download and use.

Releases

Tags

v0.0

Existing tag

v0.0

Write

Preview

H

B

I

≡

<>

🔗

≡

≡

📄

@

📄

↶

✓ Auto-generate release notes

## What's Changed

\* Second commit. by @geofyao in <https://github.com/FeiYao-Edinburgh/mesd/pull/1>

\* radiative.sh to call all submodules of radiative transfer models. by @geofyao in <https://github.com/FeiYao-Edinburgh/mesd/pull/2>

## New Contributors

\* @geofyao made their first contribution in <https://github.com/FeiYao-Edinburgh/mesd/pull/1>

\*\*Full Changelog\*\*: <https://github.com/FeiYao-Edinburgh/mesd/commits/v0.0>

Attach files by dragging & dropping, selecting or pasting them.

↓ Attach binaries by dropping them here or selecting them.

☐ This is a pre-release

We'll point out that this release is identified as non-production ready.

Publish release

Save draft

Releases

Tags

now

geofyao

v0.0

de6855c

Compare

v0.0

Latest

What's Changed

• Second commit. by @geofyao in #1

• radiative.sh to call all submodules of radiative transfer models. by @ge

New Contributors

• @geofyao made their first contribution in #1

Full Changelog: <https://github.com/FeiYao-Edinburgh/mesd/commits/v0.0>

Contributors

geofyao

Assets 2

Source code (zip)

Source code (tar.gz)

# .gitignore: prevent unwanted file types

---

- Can reside in root and subdirs
- Prevent intermediate/backup files.
  - .ipynb\_checkpoints
  - \_\_pycache\_\_
- Prevent large data files and so forth.
  - \*.nc
  - \*.nc4
  - Etc.
- Etc.

# Issues

- Assign specific issues/tasks to specific collaborators:

- Bugs to be fixed
- Features to be added
- Etc.

## Add some codes to radiative.sh #3

Open FeiYao-Edinburgh opened this issue now · 0 comments



FeiYao-Edinburgh commented now

Add some codes to radiative.sh



FeiYao-Edinburgh added the enhancement label now



FeiYao-Edinburgh assigned geofyao now

Issues Pull requests Actions Projects Wiki Security Insights Settings

Add some codes to radiative.sh

Write Preview H B I E < > Link List Checkmark @ Share ↩

Add some codes to [radiative.sh](#)

Attach files by dragging & dropping, selecting or pasting them.

Styling with Markdown is supported

Submit new issue

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

Assignees

geofyao

Labels

Apply labels to this issue

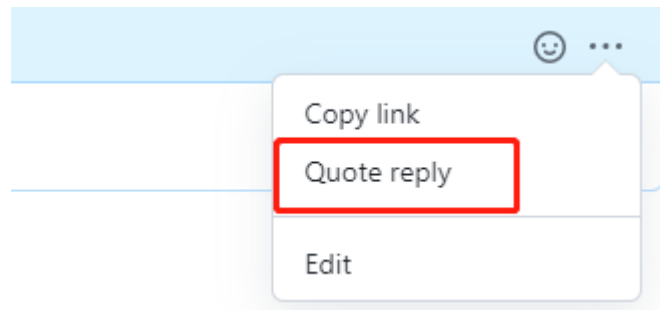
Filter labels

- bug  
Something isn't working
- documentation  
Improvements or additions to documentation
- duplicate  
This issue or pull request already exists
- enhancement  
New feature or request
- good first issue  
Good for newcomers
- help wanted  
Extra attention is needed
- invalid  
This doesn't seem right
- question
- Edit labels

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training

# Issues

- Make scientific discussions
  - Simple but powerful markdown
  - Quote reply



## Noting down some thoughts about machine learning

- Not sure if the current model can handle this
- ML in GEE seems to require outlining the area of interest for classification
- Might need to retrain a new model
  - o This will require a new training set
  - o We would need to outline the area of interest
- This would produce a more useful map (e.g., showing the proximity to infrastructure etc)
- Allows further development of the model
- Current training data set will look different to the training data that isn't seen in TROPOMI

## \*Training a new model:

- Labour intensive but possible short cuts
- If we use the old dataset of images (which I still have), upload them to GEE then outline the areas we don't have to search for new plumes
- We would need to manipulate the images to be uploaded (e.g., provide fake coordinates & times) and can then draw round the plumes using GEE
- Don't need 'no plume' examples as it'll just use the surrounding areas
- Possibly more errors involved because we need to decide the extent of a plume

## 3.2 Multi-band-single-pass (MBSP) retrieval

Our second method is a multi-band retrieval that estimates methane enhancements from differences between the band-11 and band-12 reflectances measured on a single satellite pass. For this method, the fractional change in reflectance is given by

$$\Delta R_{\text{MBSP}} = \frac{c R_{12} - R_{11}}{R_{11}}, \quad (3)$$

where  $c$  is now determined by least-squares fitting of  $R_{12}$  against  $R_{11}$  across the scene. The fractional absorption model is then

$$m_{\text{MBSP}}(\Delta\Omega) = \frac{T_{12}(\Omega + \Delta\Omega) - T_{12}(\Omega)}{T_{12}(\Omega)} - \frac{T_{11}(\Omega + \Delta\Omega) - T_{11}(\Omega)}{T_{11}(\Omega)}. \quad (4)$$

This approach relies on surface reflectance similarities between the two adjacent bands. The empirical scaling factor

# Licence

---

- The LICENSE, LICENSE.md, or LICENSE.txt file is often used in a repository to indicate how the contents of the repo may be used by others.
  - The license is best chosen from the get-go, even if for a repository that is not public.
  - Choosing a license that is in common use makes life easier for contributors and users, because they are more likely to already be familiar with the license and don't have to wade through a bunch of jargon to decide if they're ok with it.
  - People who incorporate General Public License (GPL'd) software into their own software must make their software also open under the GPL license; most other open licenses do not require this.
  - People who are not lawyers should not try to write licenses from scratch.

# GitHub Pages

---

- GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.
  - Public versus private

## GitHub Pages

---

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

### Source

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more.](#)

None ▾

Save

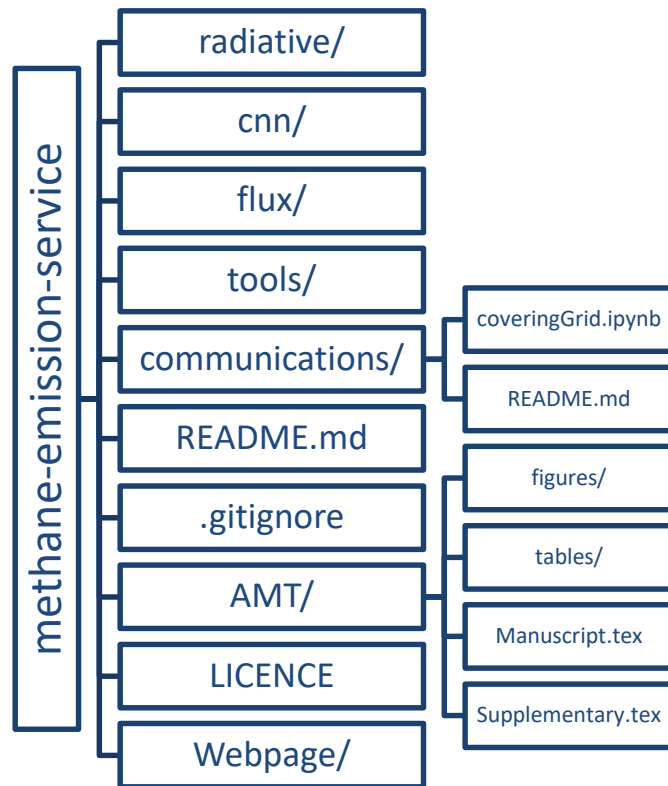


**Publish privately to people with read access to this repository**

Try risk-free for 30 days using a GitHub Enterprise organization, or [learn more about changing the visibility of your GitHub Pages site.](#) ✕

# The organization of the project

- A good top-design can make life easier.
- Can link with Overleaf:
  - Link the entire project to Overleaf.
  - Link the paper subdir (e.g. AMT/) to Overleaf. This will involve some further git commands.
  - Either way, we place all codes including Latex scripts within one place!



# Reference

---

- An Intro to Git and GitHub for Beginners (Tutorial)
  - <https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners>
- Version Control with Git
  - <https://swcarpentry.github.io/git-novice/>
- Using Git Tags To Version Coding Tutorials
  - <https://medium.com/@emmabostian/using-git-tags-to-version-coding-tutorials-cf9ff28fad4f>
- Git and GitHub cheat sheet
  - <https://education.github.com/git-cheat-sheet-education.pdf>



# Additional slides

---

- Initialize a repo from local machines and link it to remote.

# Initialize a repo from local machines and link it to remote

---

- Create a local git repository.
  - `cd`
  - `mkdir methane-emission-service # methane emission service development`
  - `cd methane-emission-service`
  - `git init`
  - `git checkout -b main # primary branch name`

# Initialize a repo from local machines and link it to remote

---

- Add a new file (README.md) to the repo.
  - `touch README.md`
  - `git status` # git has noted the file but won't track it unless we explicitly add it in a commit.
- Add the file to the staging environment.
  - `git add README.md`
  - `git status` # git has added the file to a staging environment but not a commit although it's about to.
- Package all files added to the staging environment to a commit.
  - `git commit -m 'Initial commit.'` # leaving a clear explanation of your changes will be extremely helpful for future programmers (perhaps future you!)
  - `git log` # see that the commit has been added to the log


# Initialize a repo from local machines and link it to remote

- Create a remote repo.

## Create a new repository

A repository contains all project files, including the revision history. Already have a repository? [Import a repository.](#)

Owner \*

 FeiYao-Edinburgh ▾

Repository name \*

methane-emission-service ✓

Great repository names are short and memorable. Need inspiration? How about...

Description (optional)

Methane Emission Service Development

☐  Public

Anyone on the internet can see this repository. You choose who can commit.

☒  Private

You choose who can see and commit to this repository.

### Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore


Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository

## Quick setup — if you've done this kind of thing before

 Set up in Desktop or ☐ HTTPS ☐ SSH

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

## ...or create a new repository on the command line

```
echo "# methane-emission-service" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/FeiYao-Edinburgh/methane-emission-service.git
git push -u origin main
```

## ...or push an existing repository from the command line

```
git remote add origin https://github.com/FeiYao-Edinburgh/methane-emission-service.git
git branch -M main
git push -u origin main
```

Only needed if not a "git checkout -b main" after "git init".

## ...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

# Initialize a repo from local machines and link it to remote

- Link local and remote repo.
  - `git remote add origin https://github.com/FeiYao-Edinburgh/methane-emission-service.git`
  - `git push origin main` # origin is an alias of remote repo!
  - `git remote -v`

The screenshot displays a GitHub repository interface. At the top, a navigation bar includes a branch selector showing 'main' with a dropdown arrow, '1 branch', and '0 tags'. To the right are buttons for 'Go to file', 'Add file', and 'Code'. Below this, a commit entry for 'FeiYao-Edinburgh' is shown with the message 'Initial commit.', commit hash '5021e30', and timestamp '43 seconds ago'. A red box highlights the '1 commit' indicator. Below the commit, a file named 'README.md' is listed with the same commit information. At the bottom, a light blue banner prompts the user to 'Add a README with an overview of your project.' with a corresponding 'Add a README' button.

main ▾ 1 branch 0 tags

Go to file Add file ▾ Code ▾

FeiYao-Edinburgh Initial commit. 5021e30 43 seconds ago 1 commit

README.md Initial commit. 43 seconds ago

Add a README with an overview of your project. Add a README