

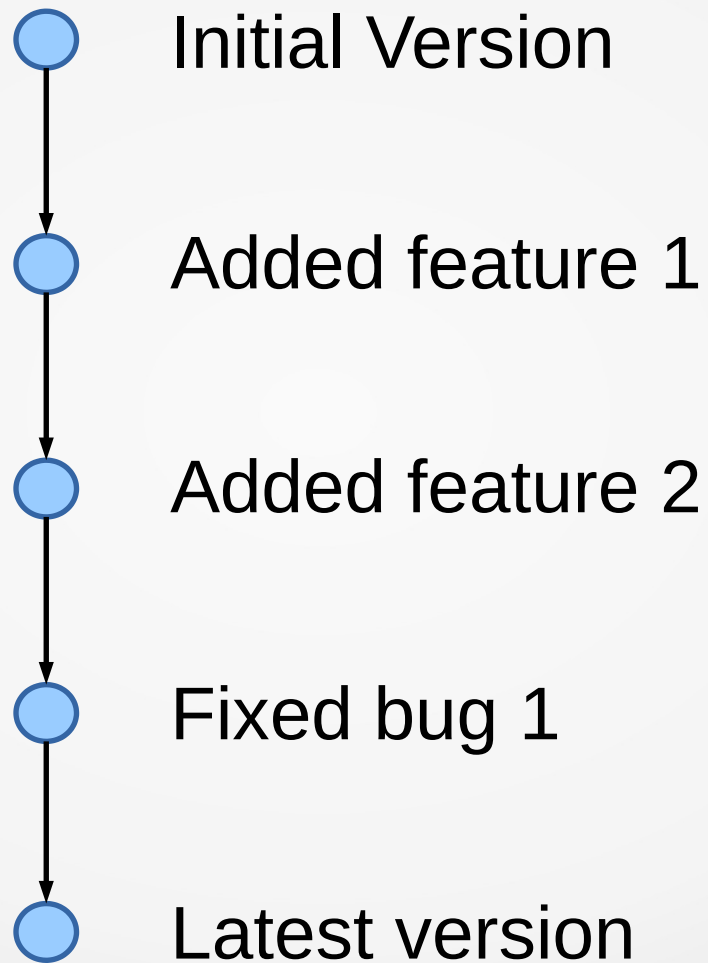
Git-Version Control



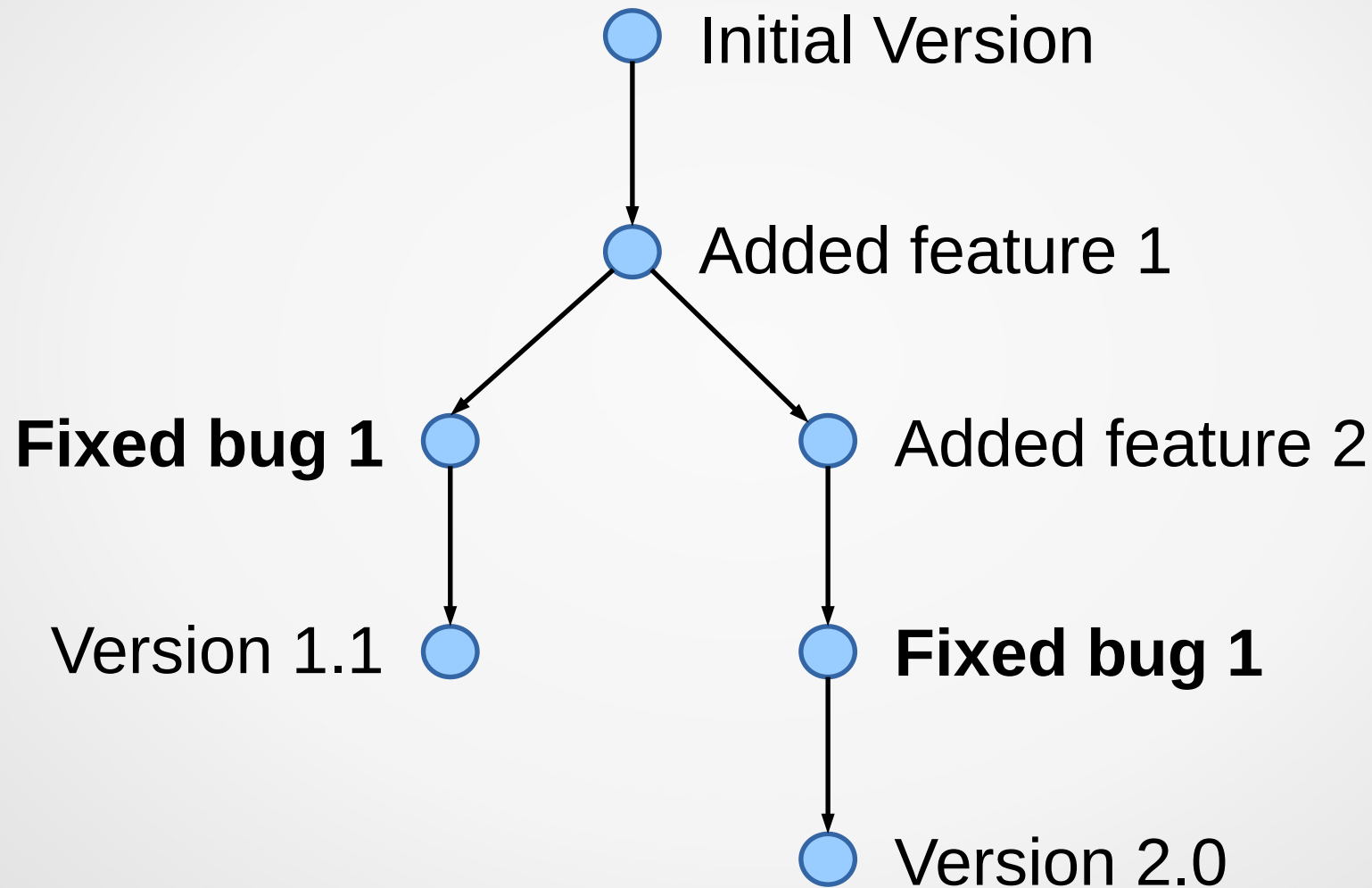
Why?

- Keep track of changes
- Work as a group
- Release Managements
- Identify regressions easily
- Maintain personal changes to code elsewhere

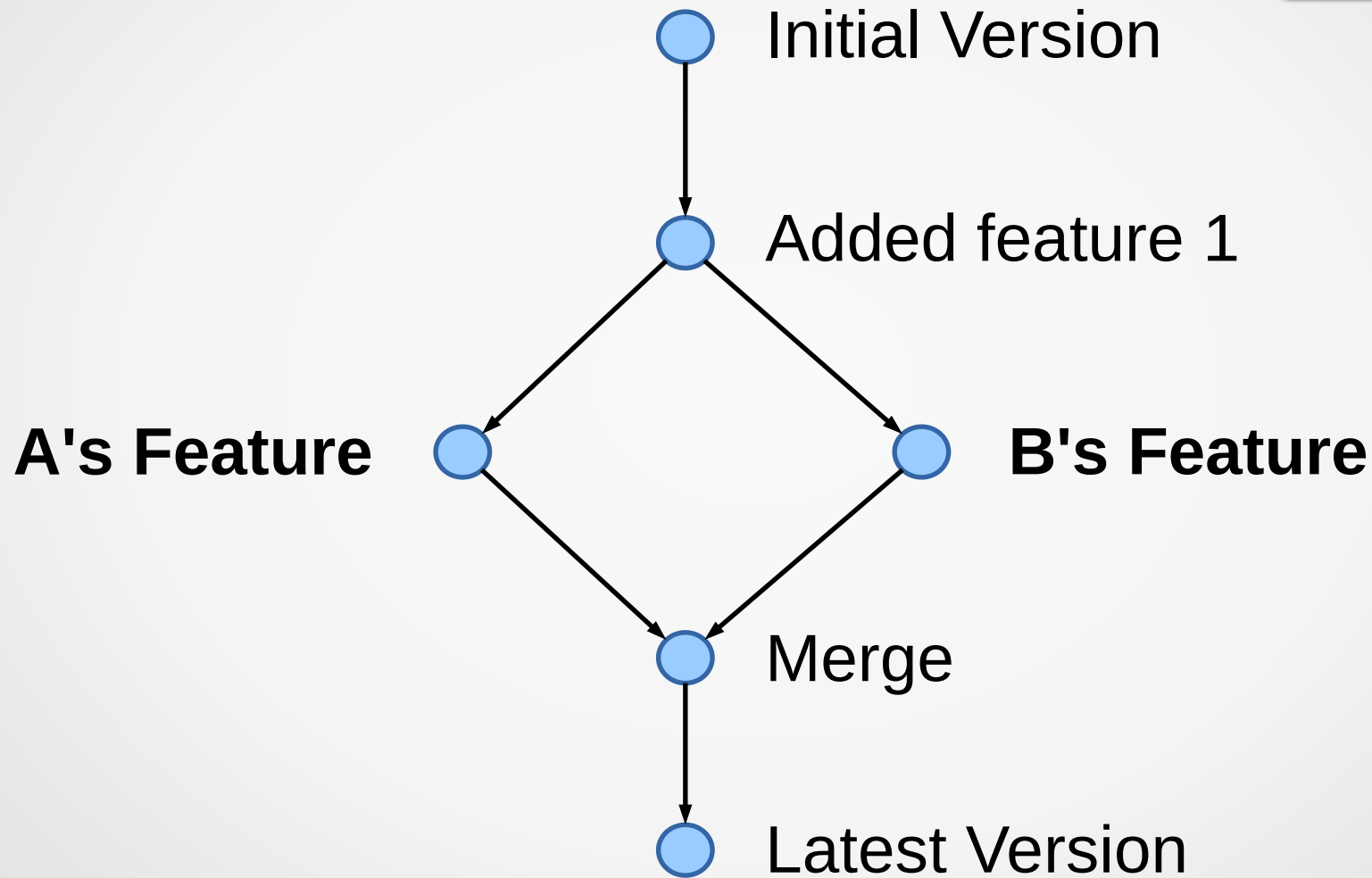
Revisions



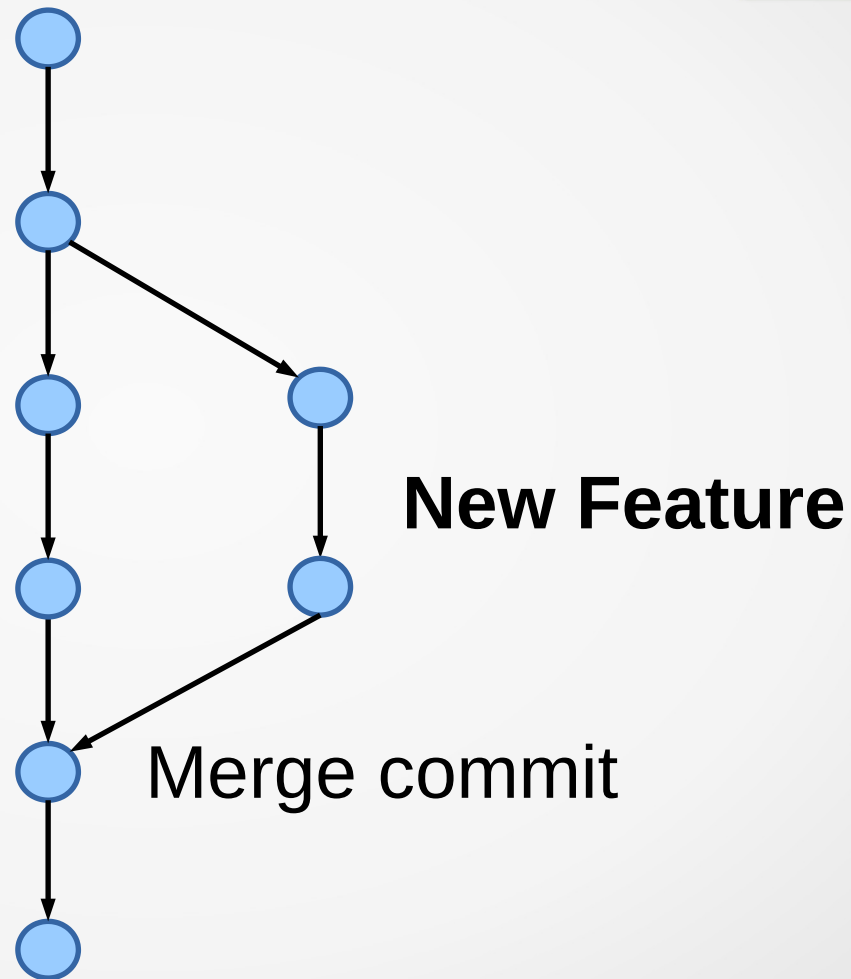
Release Management



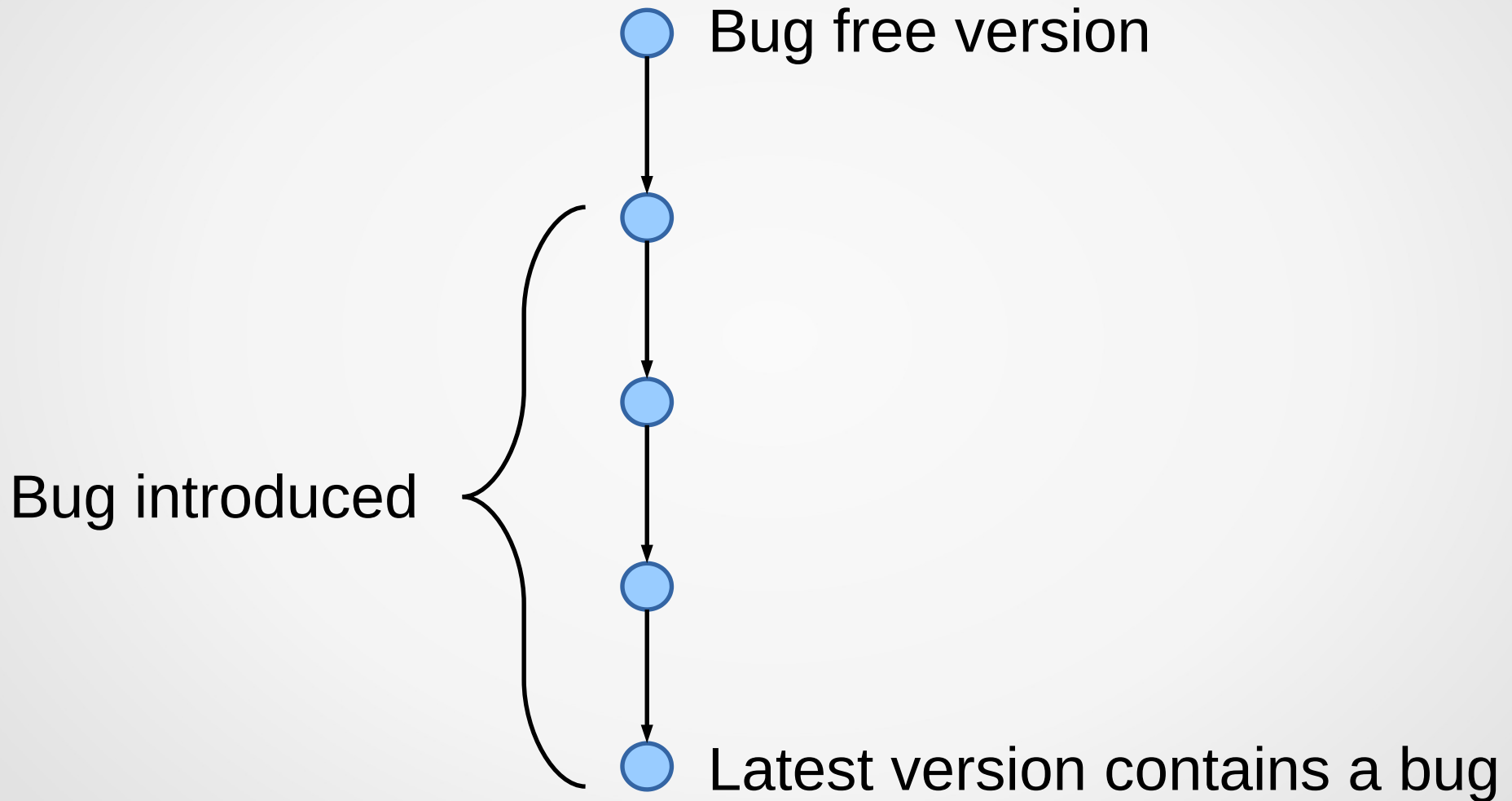
Work as a Group



Working in Branches

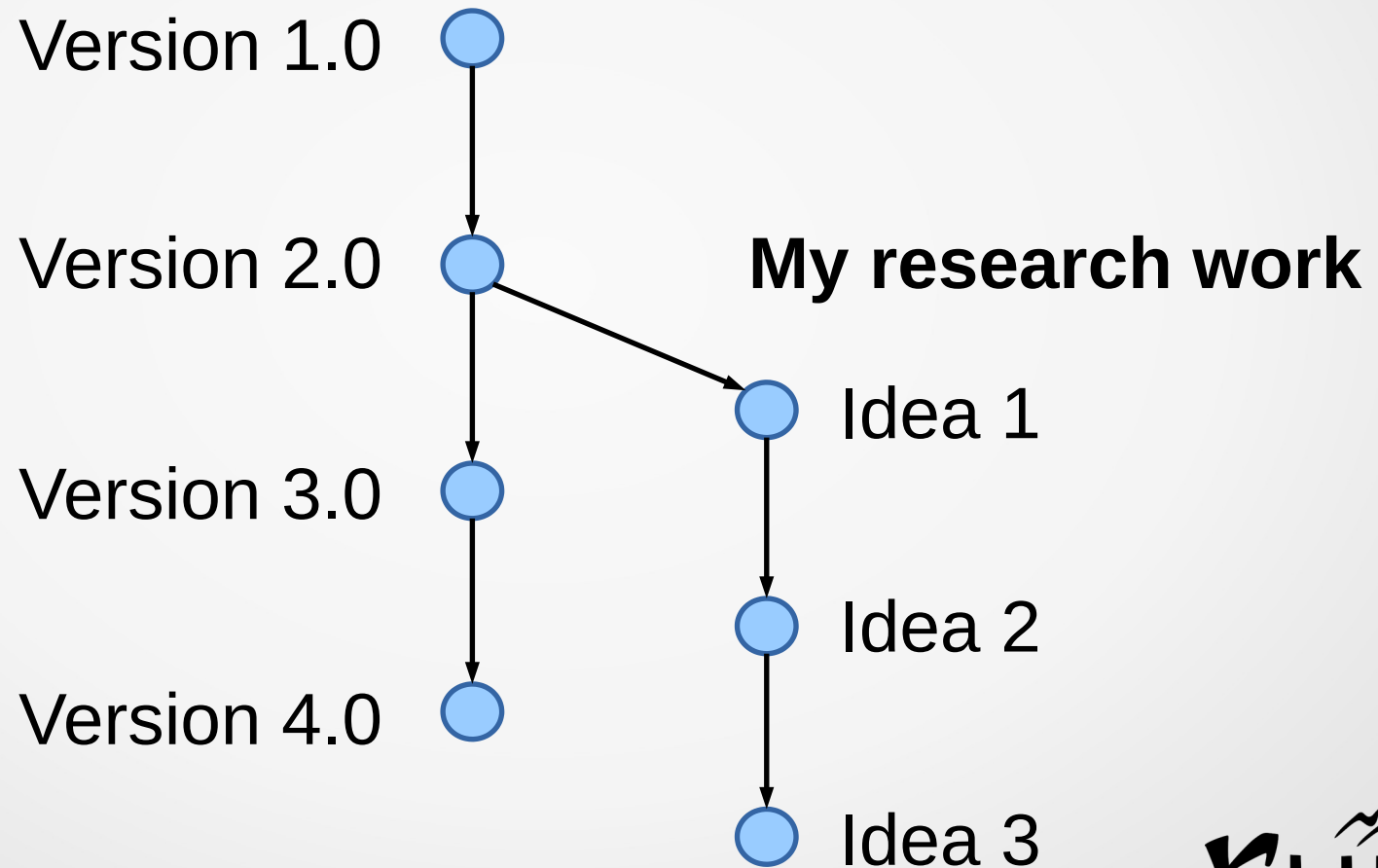


Identify Regressions



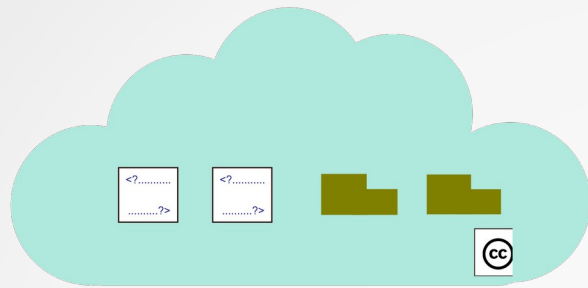
Personal Changes

Free Software Project



gitlab

Fork



Repository-1



Repository-1

Fork-R>R

Remote



Local



git

Clone-R>L



Repository-1

How to install git

- Windows user
→ git Bash

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

- Linux User

→ `$ sudo apt install git`

- How to check whether git is installed or not?

`$ git` (In terminal)

Getting Started with Git -config part

```
$ git config --global user.name "abcd"
```

```
$ git config --global user.email  
"abcdef@gmail.com"
```

Create a Repository

To clone a Repo

```
$ git clone "Project Repo URL"
```

```
$ mkdir myproject; cd myproject
```

```
$ git init
```

Initialized empty Git repository in
~/myproject/.git/

Adding Files to Repository

```
$ nano swecha
```

```
$ git add swecha
```

```
$ git status
```

```
...
```

Changes to be committed:

new file: swecha

Committing Files

```
$ git commit -m 'First commit'
```

```
$ git log
```

Editing Files

- Edit your file

```
$ nano swecha
```

- To display the content in that file

```
$ cat swecha
```

Reviewing Changes

```
$ git diff
```


Reviewing Changes & Committing

```
$ git status
```

Changes not staged for commit:

```
    modified:   swecha
```

```
$ git add swecha
```

```
$ git commit -m 'Second change'
```

Viewing Historic Changes

```
$ git log
```

```
4a66009 Second change
```

```
c9cea53 First commit
```

Creating a Branch

- Creating a branch

```
$ git branch myexperiment
```

- Listing branches

```
$ git branch
```

```
* master  
myexperiment
```

Switching a Branch

```
$ git branch
* master
  myexperiment

$ git checkout myexperiment

$ git branch
  master
* myexperiment
```

Merging a Branch

```
$ git checkout master
```

```
$ git merge myexperiment
```

Pushing the changes

```
$ git push origin master
```

→It will push the changes from local repo to Remote repository

Pull Changes from Remote Server

```
$ git pull
```

→It will update the changes in the remote repository to your local repository.

More Topics of Interest

- Bisecting
- stashing
- Rebasing
- Changing history
- Interactive operations
- Signing commits
- Cherry-picking
- Grep in history
- Annotate

Graphical Tools: Gitg

The screenshot displays the Gitg application window titled "awareness-about-govt-schemes (~/Desktop)". The interface is divided into several sections:

- Left Sidebar:** Contains navigation options: "All commits", "Branches" (with "master" selected and noted as "1 ahead"), "Remotes" (showing "origin" with "master"), and "Tags".
- Commit History:** A list of commits on the "master" branch. The most recent commit is highlighted in blue: "Project" by "Ch.Naga Malleswari" on "Aug 10, 18:00". Other recent commits include "Update README.md" and "Govt schemes and policies" by the same user.
- Commit Details:** Below the history, the selected commit is shown with the author "Ch.Naga Malleswari" and email "nagamalli.chnr@gmail.com", dated "Monday 10 August 2020 06:00:38 +05:30 +0530". The commit message "Project" is visible.
- Diff View:** At the bottom, a diff for "Government Policies and Schemes/README.md" is shown. It highlights changes in red (removals) and green (additions). The changes include adding a project name and a description of the information system.

```
@@ -1,4 +1,4 @@
1  - # Project Name: Government policies and Schemes
1  + # Government policies and Schemes Information System
2
3
4  - A Information System which helps the user to know about the schemes and policies
4  + An Information System which helps the user to know about the schemes and policies for which they are accessible.
```

References

- Git: <http://git-scm.com>
- Official Git Tutorial:
<http://www.kernel.org/pub/software/scm/git/docs/gittutorial.html>
- ProGit Book: <http://progit.org>
- Git Manual Pages: *man git*

Reach us...

- <https://t.me/klglug>
- <https://fb.me/SwechaAP>
- <https://t.me/SwechaAP>
- <https://twitter.com/SwechaAP>
- <https://www.instagram.com/swechaap/>