



CS4961 Software Design Laboratory I

Collaboration using Git and GitHub

Chengyu Sun
California State University, Los Angeles



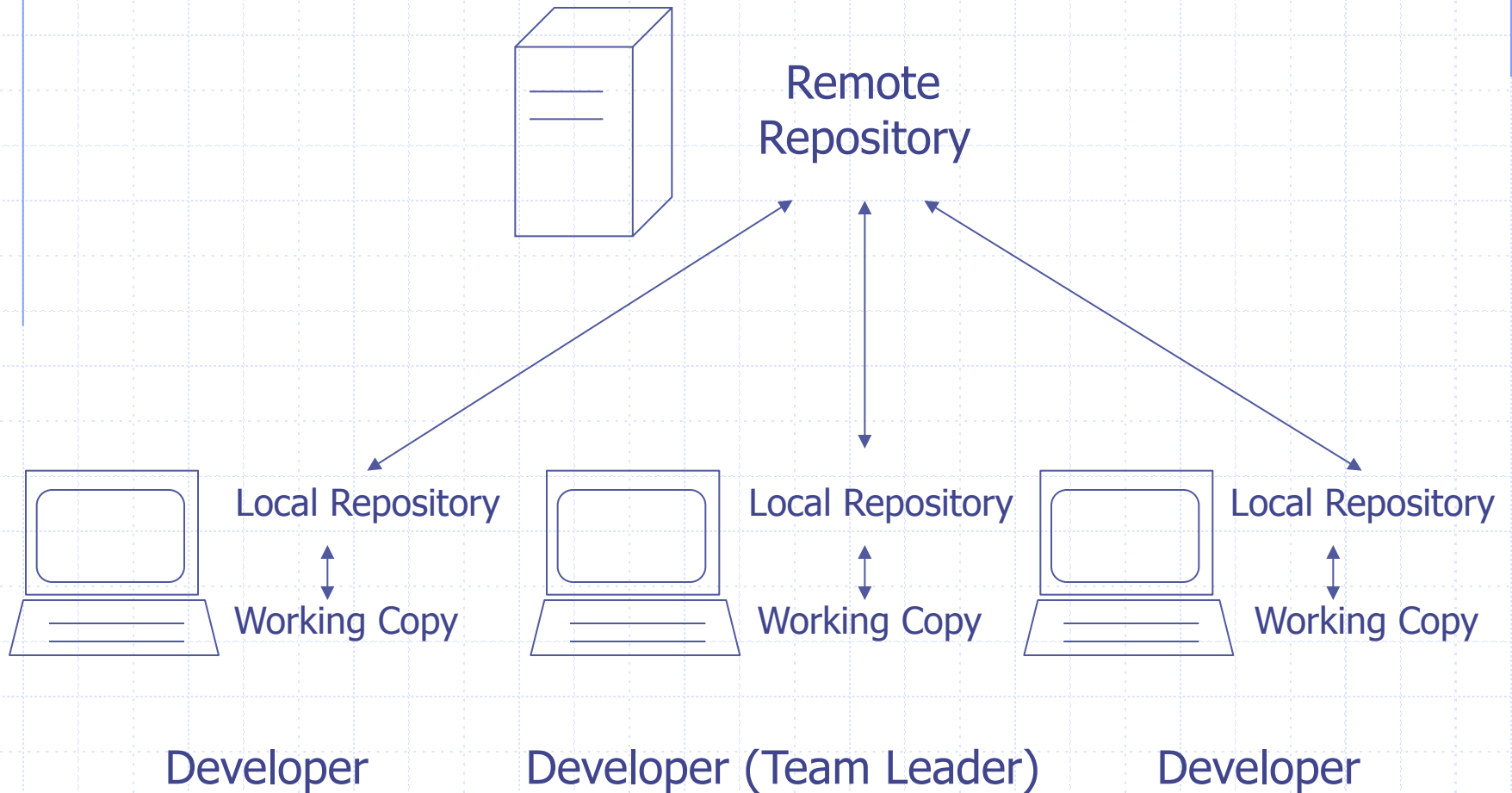
Prerequisites

- ◆ Basic version control concepts
- ◆ Common Git operations: `commit`,
`branch`, `merge`
- ◆ Common GitHub (i.e. remote Git repository) operations: `clone`, `push`,
`pull`
- ◆ Pro Git, Chapter 1-3 -
<https://git-scm.com/book/en/v2>

Collaboration Models

- ◆ Different ways a team of developers collaborate on a project
 - Centralized
 - De-centralized

Centralized Model



About Centralized Model

- ◆ Each developer work on *their own branches*
- ◆ A team leader is in charge of the important branches (e.g. `master`)
- ◆ Changes can be copied between branches using merge/rebase
- ◆ Similar to using a centralized VCS

Problems of Centralized Model

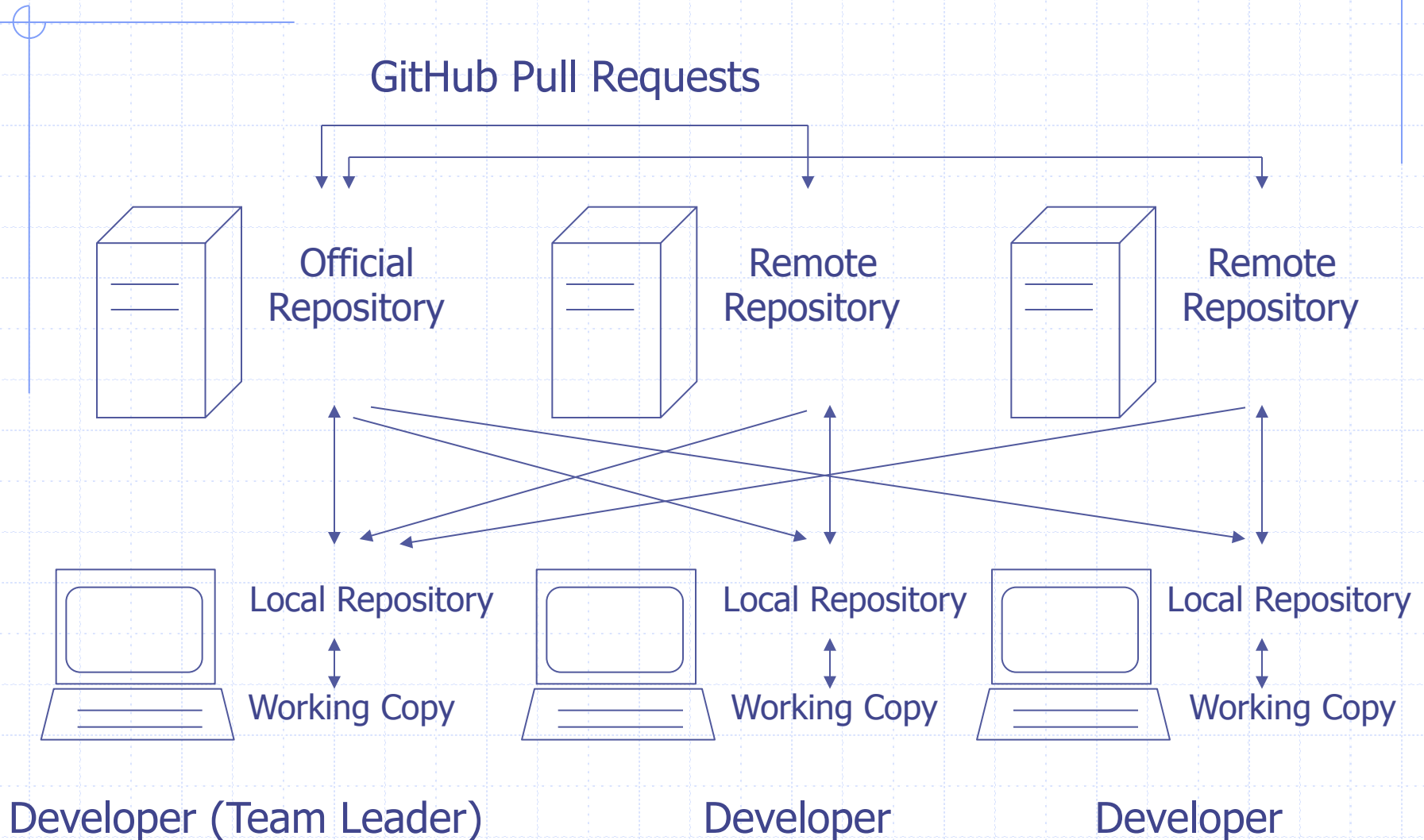
- ◆ All developers need *write* access to the same repository

- Repository “collaborators” on GitHub



- ◆ Infeasible for project with lots of contributors
- ◆ One inexperienced/malicious developer could mess up the whole repository

De-Centralized Model



About De-Centralized Model

- ◆ Each developer work with their own remote repository
- ◆ A team leader maintains the “official” project repository
- ◆ Changes can be shared using *Git remote operations* or *GitHub Pull Requests*
- ◆ The more “git way” of collaboration

Collaboration Example

- ◆ De-centralized model
- ◆ Exchange changes using GitHub Pull Requests

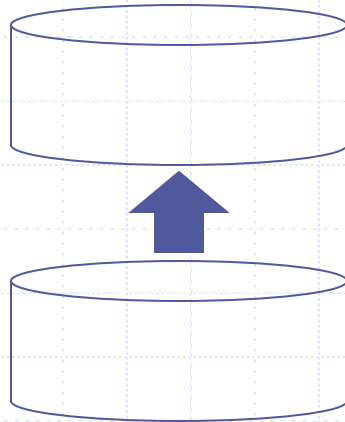
Developer	Role	Environment
senturain	Team Leader	Command line
cysun	Team Member	Eclipse

1. Put Project on GitHub

- ◆ Team leader creates the project and a local repository, then push the local repository to GitHub

GitHub

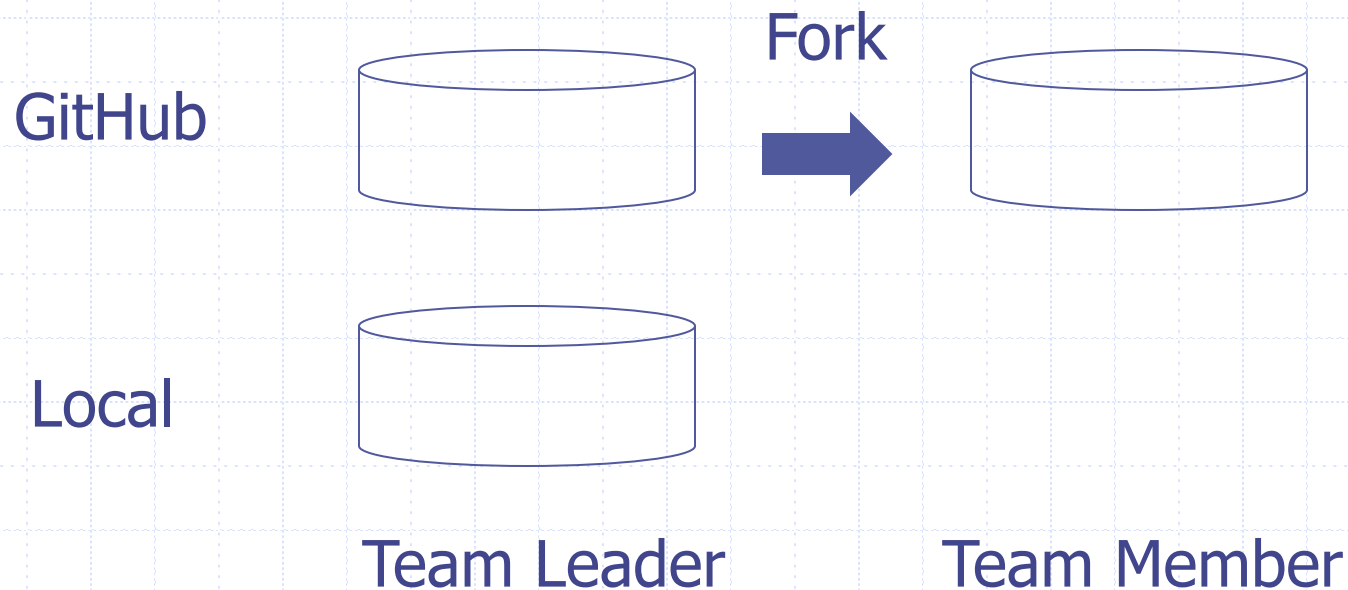
Local



Team Leader

2. Fork Project on GitHub

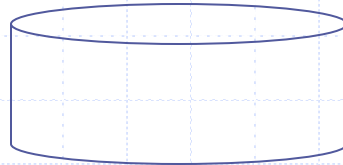
- ◆ Team member forks the project on GitHub



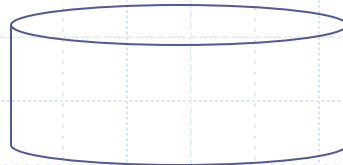
3. Clone the Project to Local Repository

- ◆ Team member clones the project, then imports it into Eclipse

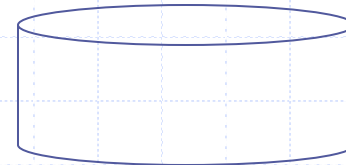
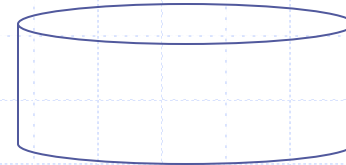
GitHub



Local



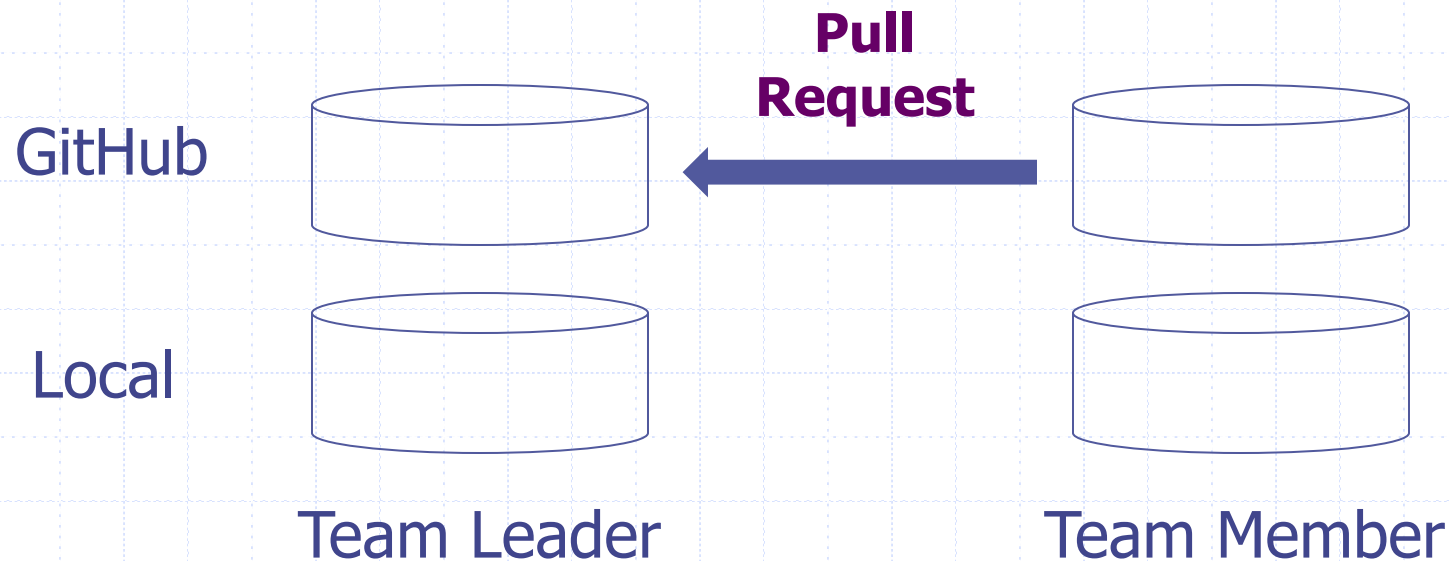
Team Leader



Team Member

4. Pull Request from Member to Leader

- ◆ Team member commits & pushes changes to his/her GitHub repo, then creates a Pull Request to notify the team leader about the changes
- ◆ Team leader accept the pull request and merges the changes into the team repository



5. Pull Request from Member to Him/Herself

- ◆ It's usually the responsibility of each project member to keep their repository in sync with the official repository
- ◆ Team member can create a Pull Request to him/herself

