



© 2015 Hoàng Anh Tú 



Seminar Lab 01

GIT: Hệ thống quản lý phiên bản phân tán



Nội dung

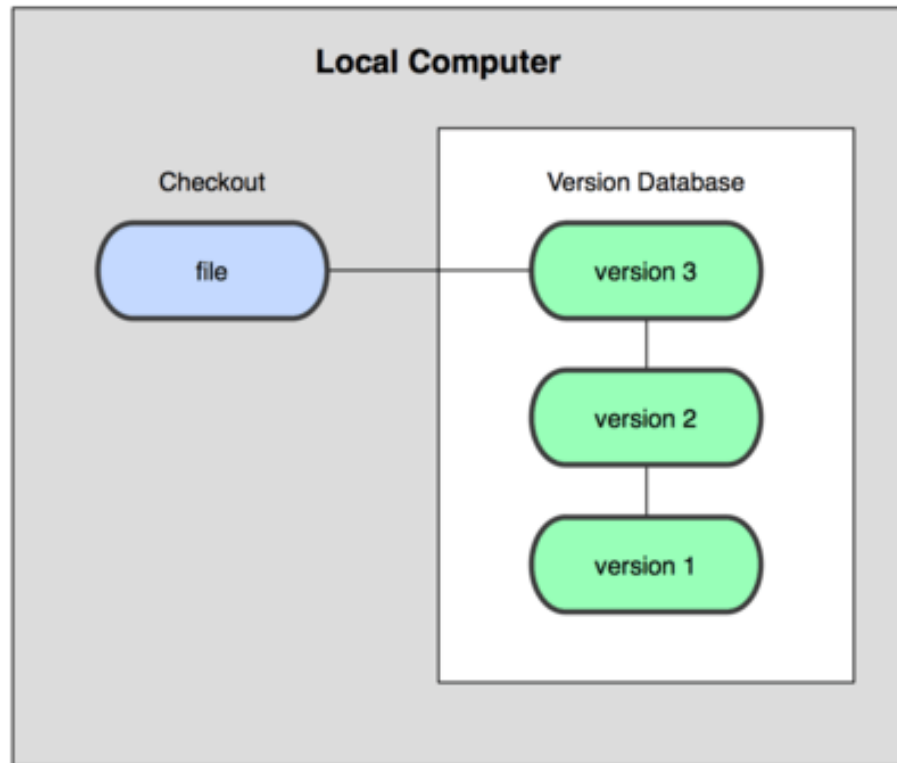




GIỚI THIỆU

Hệ thống quản lý phiên bản cục bộ

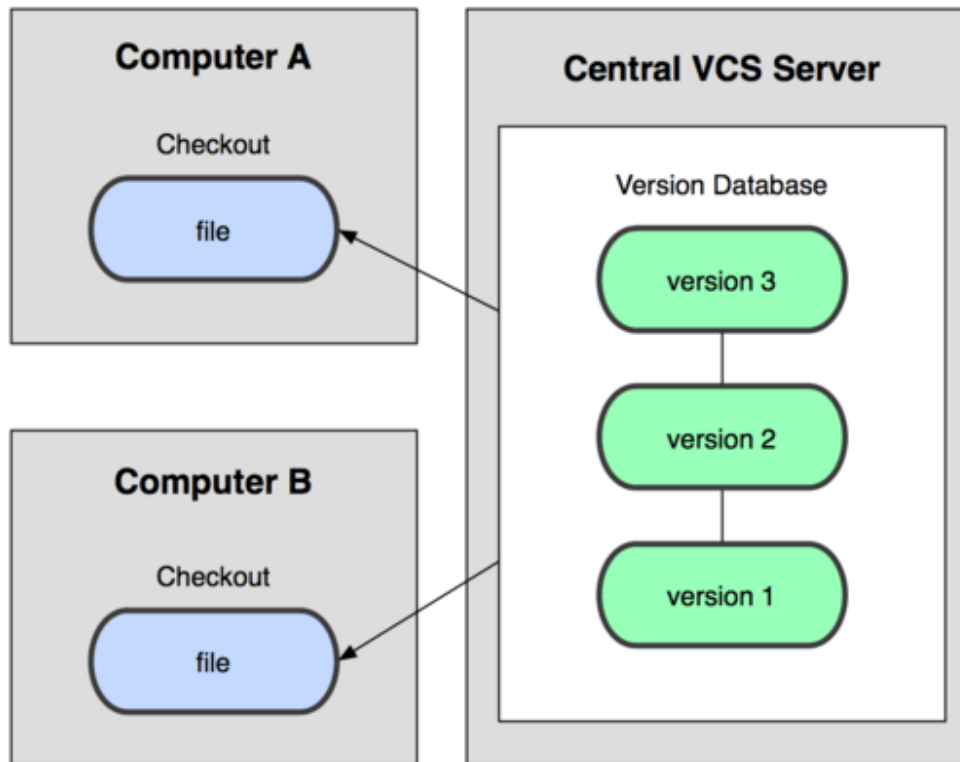
❖ *Mỗi tập tin sẽ được lưu trữ với nhiều phiên bản khác nhau*



<http://git-scm.com/book/en/Getting-Started-About-Version-Control>

Quản lý phiên bản tập trung

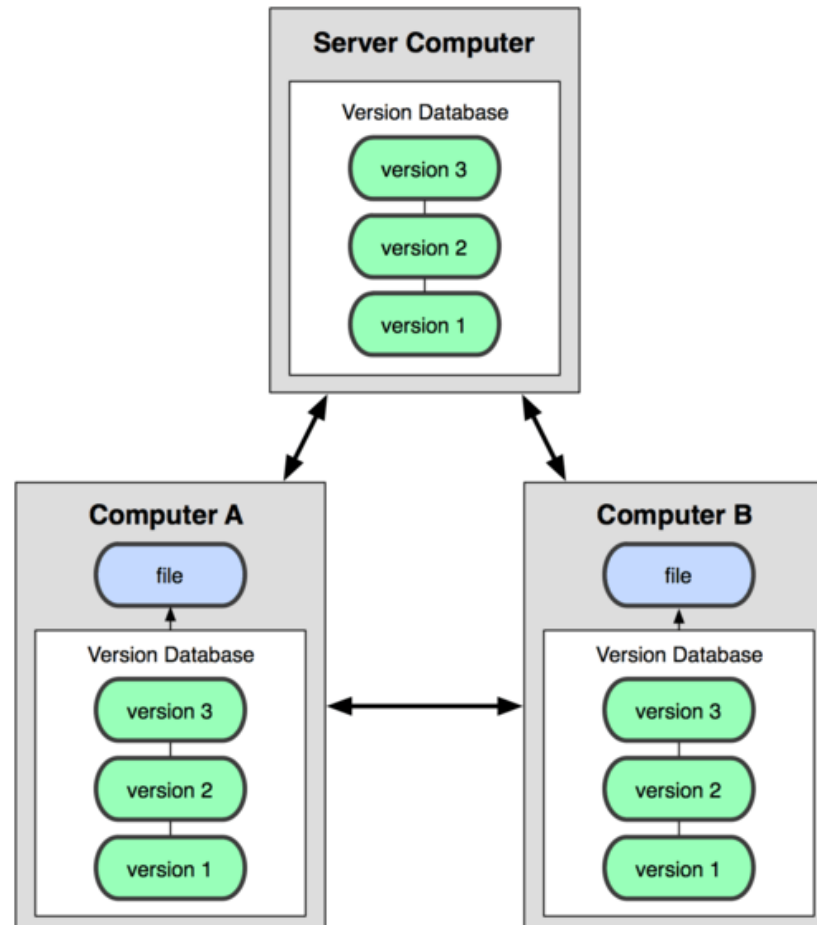
❖ *Nhiều người dùng có thể cùng lúc tương tác vào hệ thống chung*



<http://git-scm.com/book/en/Getting-Started-About-Version-Control>

GIT: Quản lý phiên bản phân tán

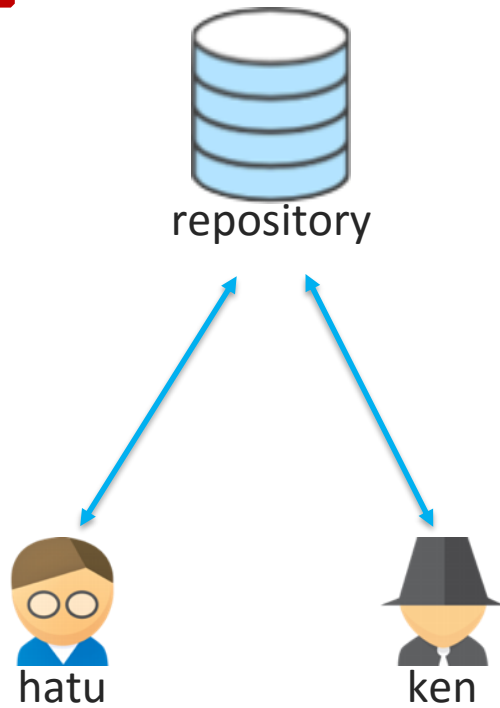
❖ *Mỗi máy có thể xem như một nơi lưu trữ tất cả phiên bản*



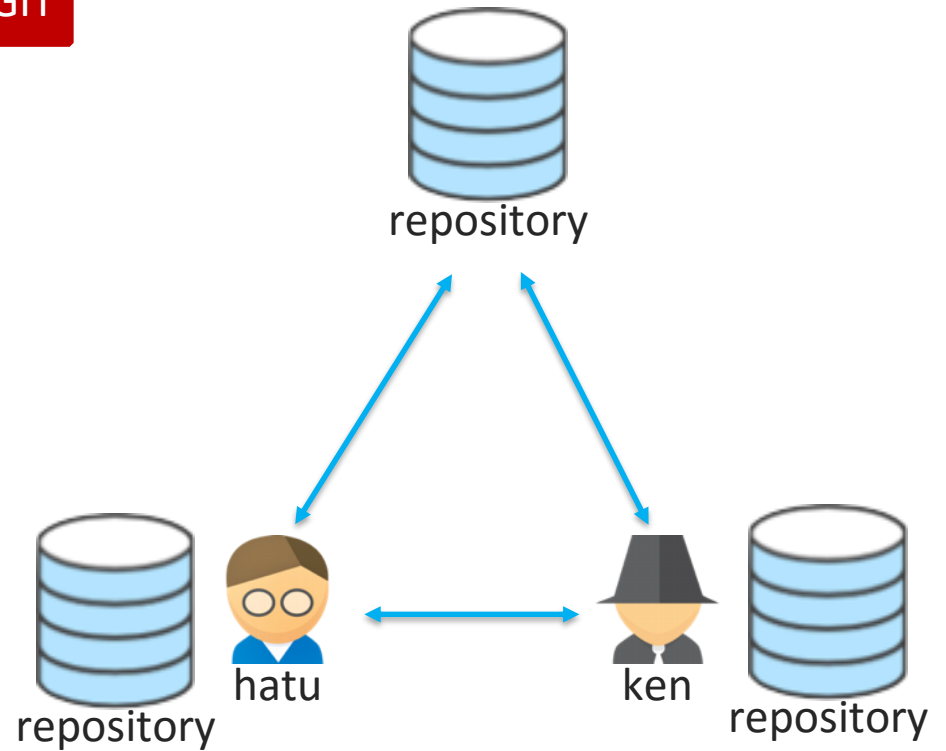
<http://git-scm.com/book/en/Getting-Started-About-Version-Control>

GIT vs SVN

SVN



GIT



Website: <http://git-scm.com>

Hỗ trợ:



Một số host có thể sử dụng: Bitbucket, GitHub,...

Một số công cụ hỗ trợ:

—Free: SourceTree, GitHub

Tài liệu tham khảo:

—<http://git-scm.com/doc>

—<https://www.atlassian.com/git/tutorials>

Chuẩn bị GIT

◆ *Đăng ký tài khoản tại một host GIT*

– Bitbucket: <https://bitbucket.org>

◆ *Download và cài đặt ứng dụng hỗ trợ*

– SourceTree: <http://www.sourcetreeapp.com>

◆ *Đăng ký SourceTree:*

– chạy Source và chọn File/Registration...

– vào tool/option, chọn tab General và nhập Fullname, Email Address



CÁC THAO TÁC CƠ BẢN

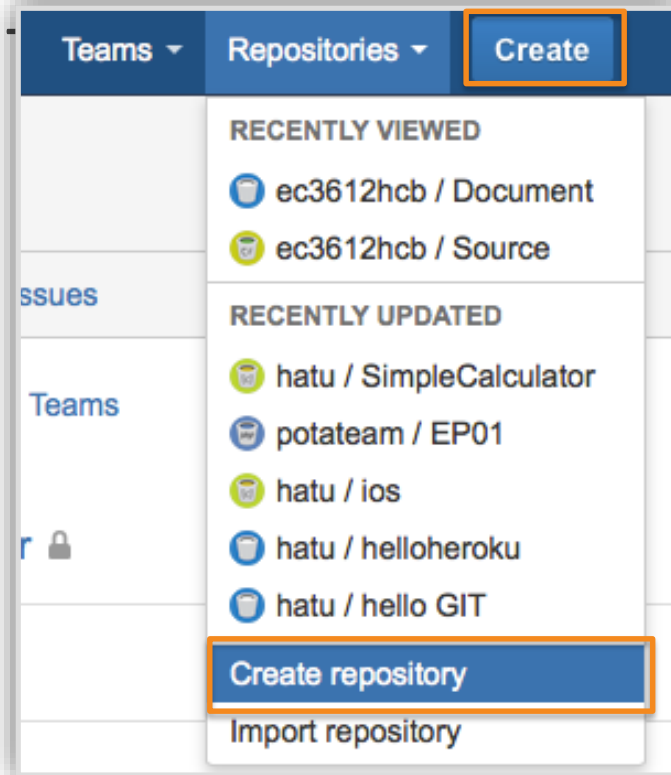
Tạo (Init) Repository (Sử dụng giao diện web của server)

❖ *Tạo repository tại server Bitbucket:*

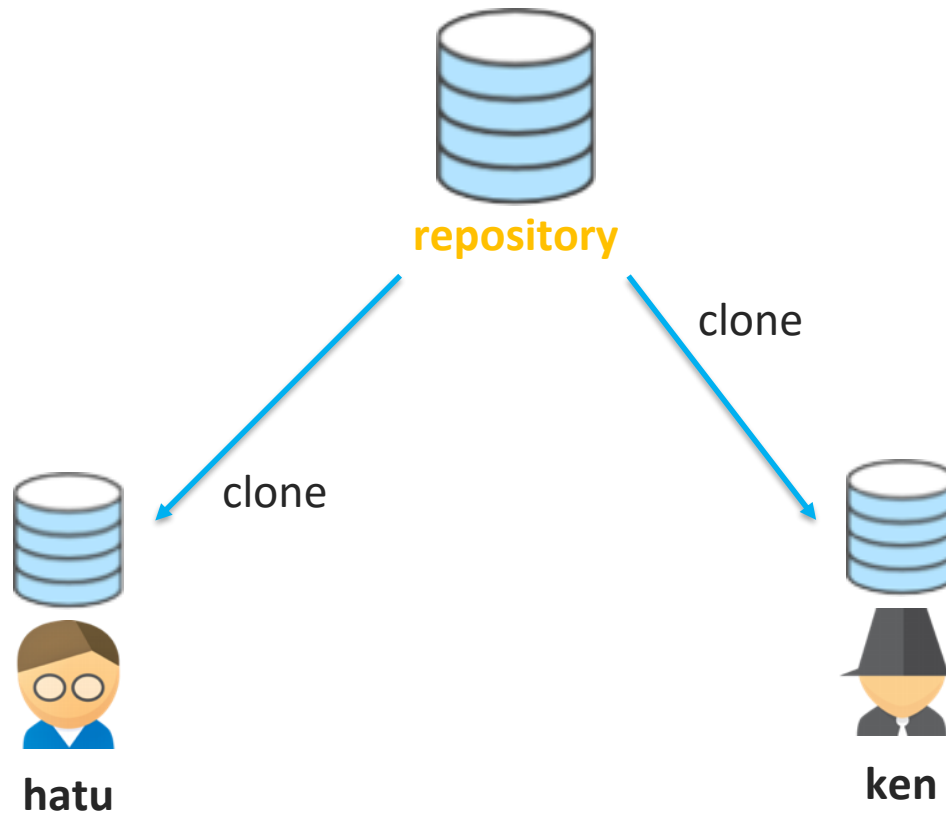
—Đăng ký tài khoản tại server Bitbucket:

www.bitbucket.org

— Chọn Repository Type là GI



Clone repository từ server về máy



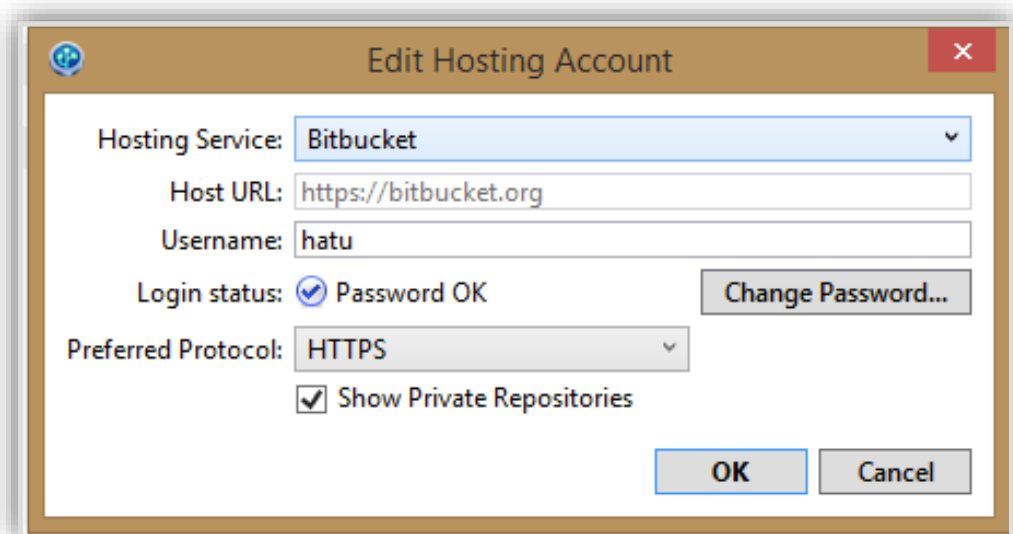
Clone repository từ server về máy

❖ *Xem các repository hiện có trong tài khoản*

—Vào View/Show Hosted Repositories...

—Chọn Edit Accounts...

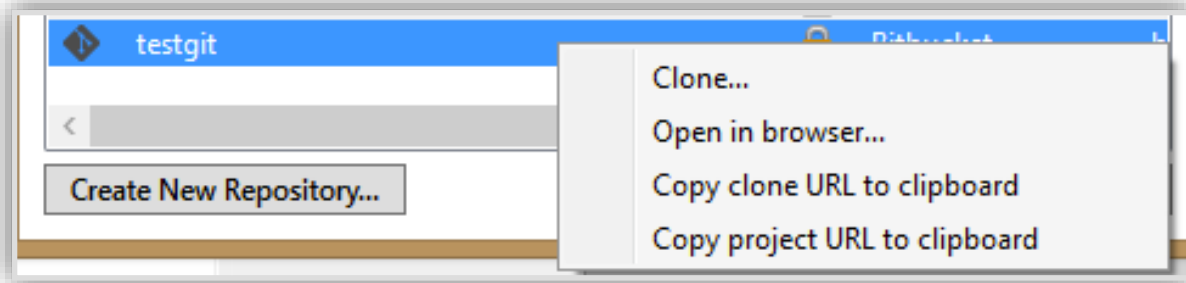
—Chọn Add và đăng nhập bằng tài khoản Bitbucket đã đăng ký



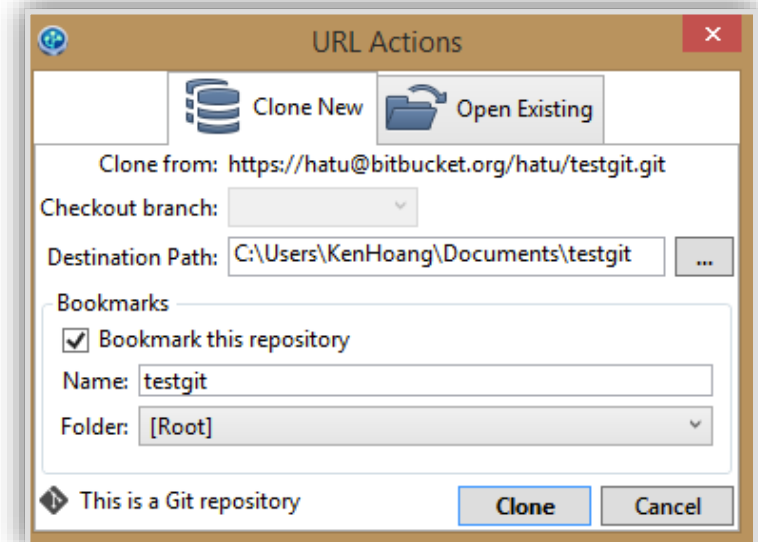
Clone repository từ server về máy

❖ Tải (clone) repository về máy

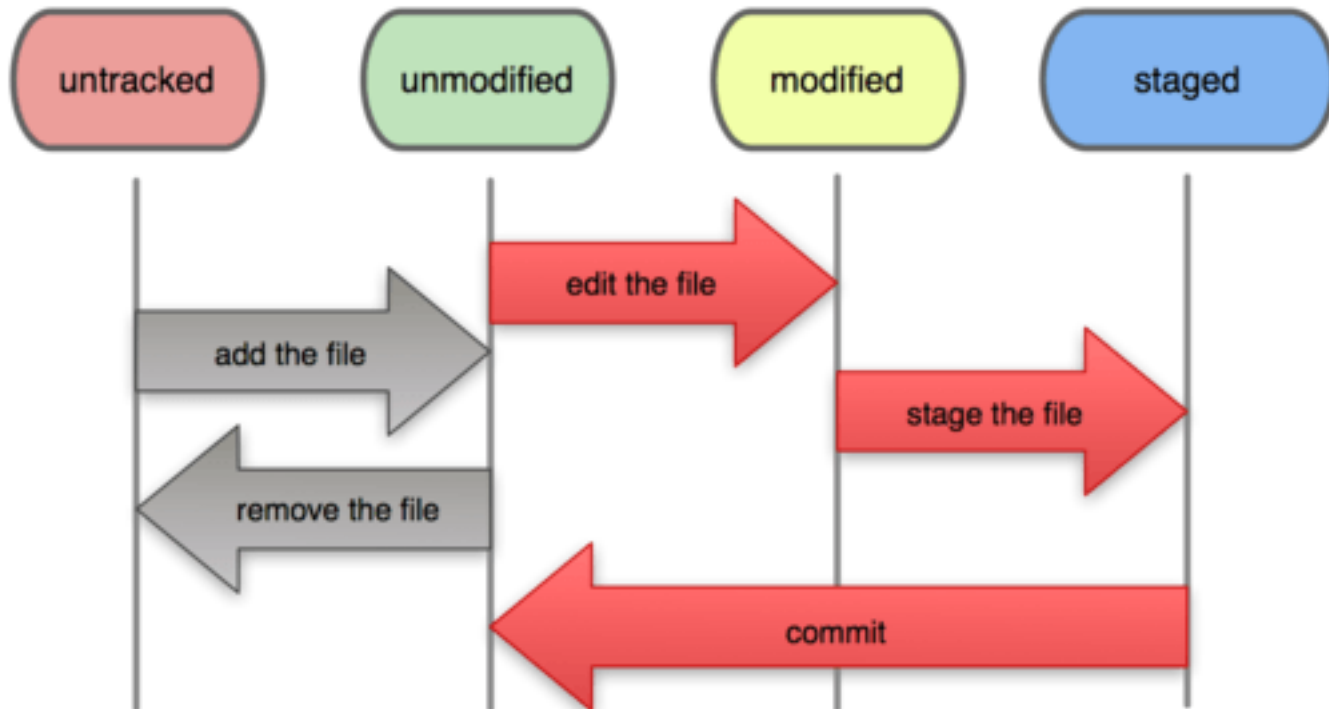
– Click phải repository cần tải và chọn Clone...



– Nhập thông tin cần thiết và đường dẫn tương ứng

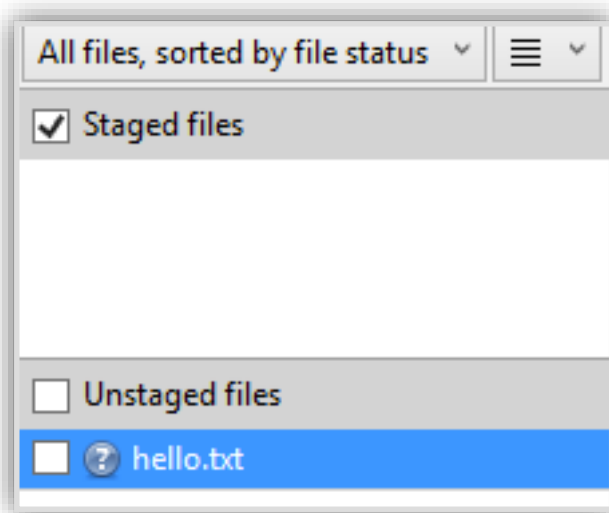


Vòng đời của các tập tin trong GIT

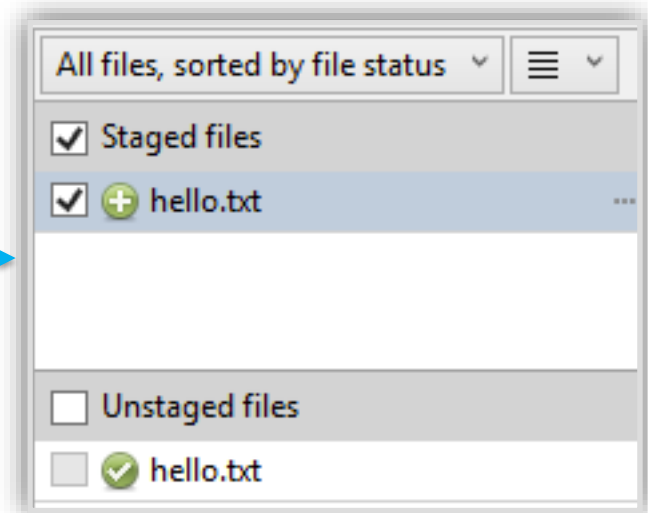


Add tập tin vào repository

- ❖ Tạo tập tin `hello.txt` vào thư mục chứa repository
- ❖ Tập tin mới tạo sẽ xuất hiện ở mục `unstaged files`
- ❖ Chọn các tập tin muốn thêm vào và chọn `Add`



trước khi add

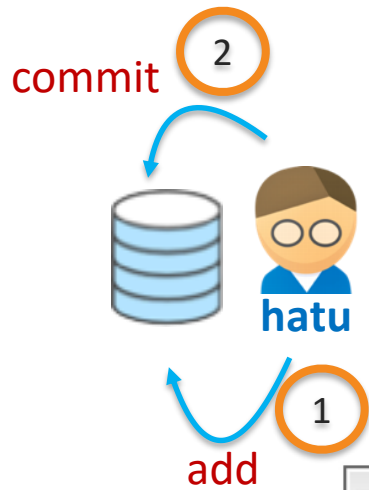


sau khi add

Commit tập tin lên hệ thống



central repository

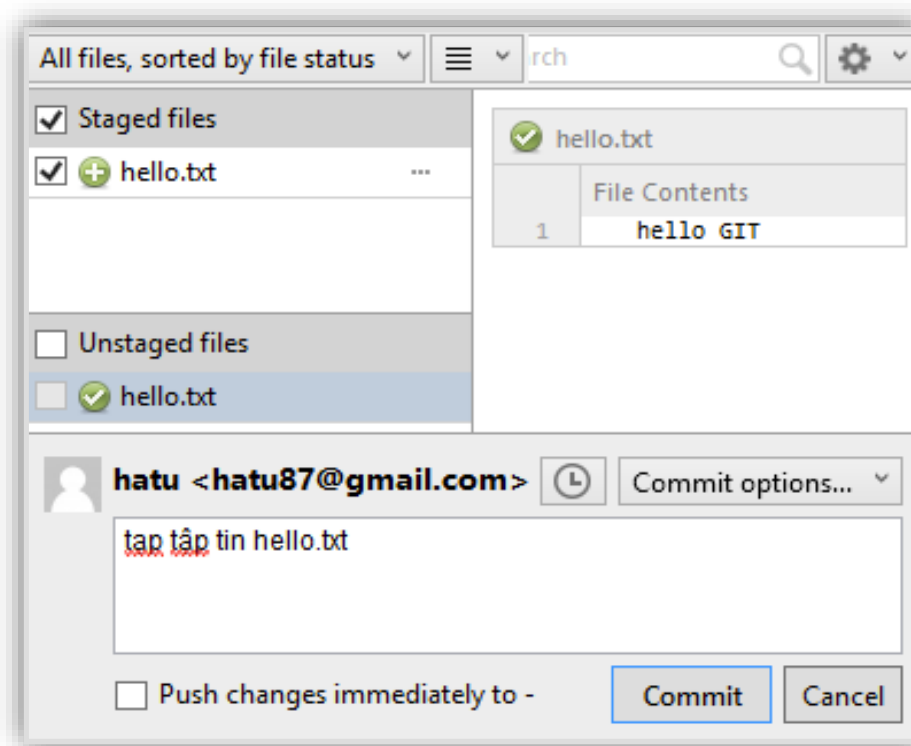


hello.txt



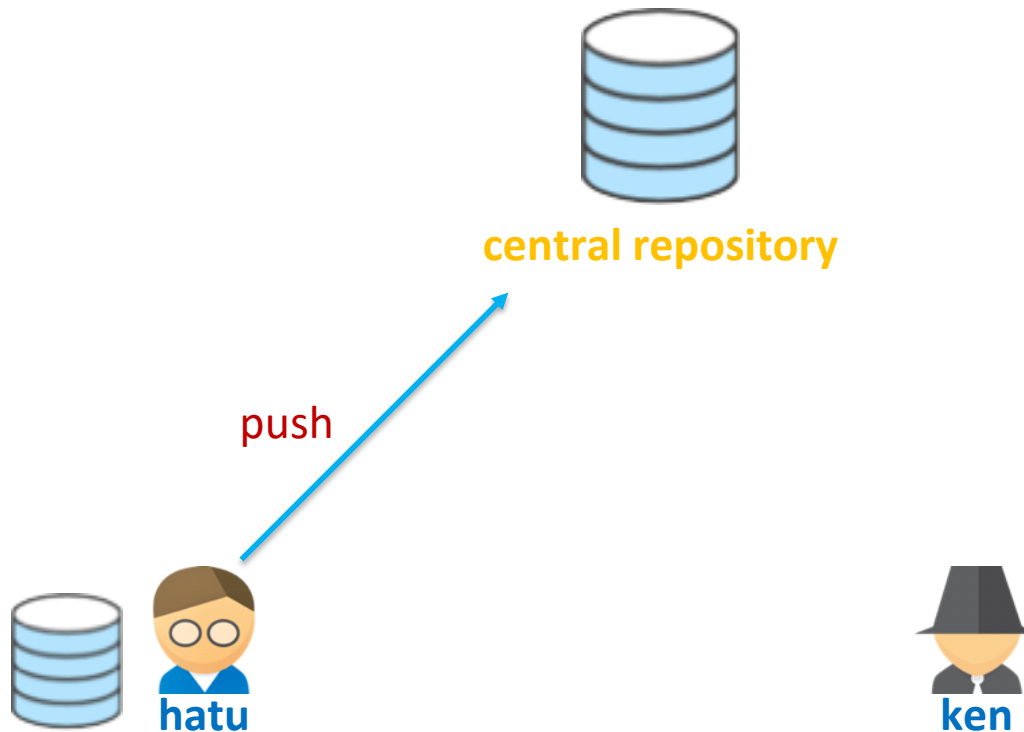
Commit tập tin lên hệ thống

- ❖ *Chọn Commit và nhập thông điệp mô tả commit*
- ❖ *Sau khi hoàn tất, nhấn commit để đưa các tập tin đã tạo lên server của BitBucket*



Push tập tin lên server Bitbucket

Push: Đưa dữ liệu từ local repository lên repository chính

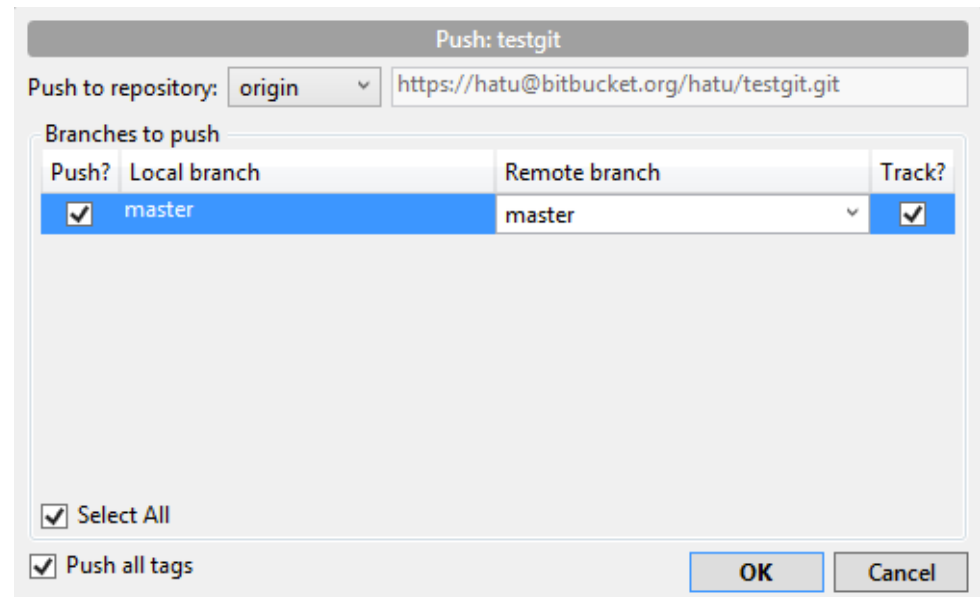


Push tập tin lên server Bitbucket

❖ *Chọn chức năng Push trên toolbar*

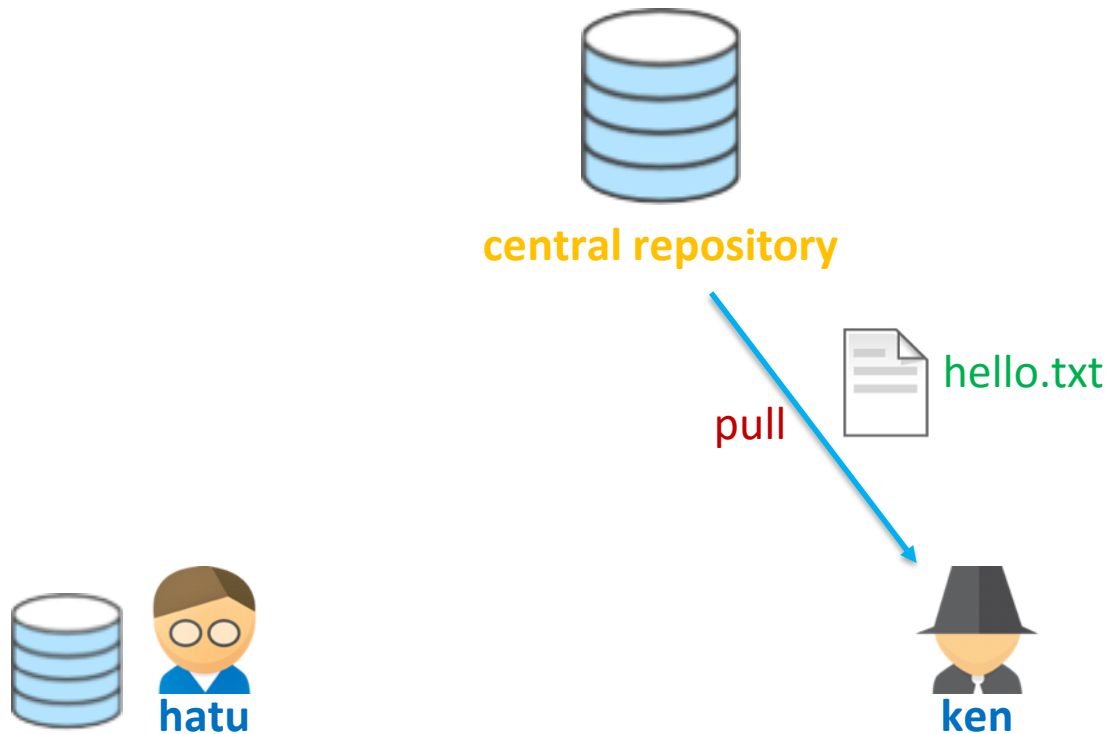
❖ *Chọn repository:*

—origin: tên remote repository của Bitbucket



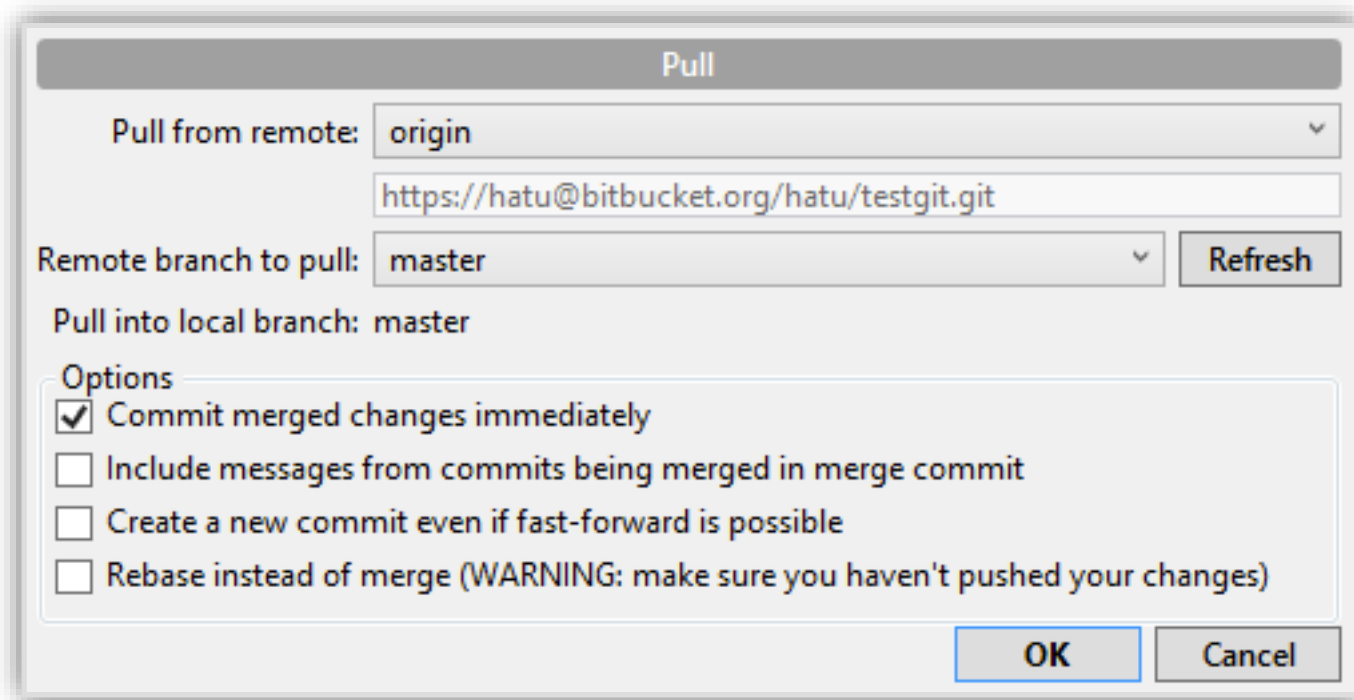
Pull dữ liệu mới từ server về

Push: Đưa dữ liệu từ local repository lên repository chính



Pull dữ liệu mới từ server về

❖ *Chọn chức năng Pull trên toolbar*

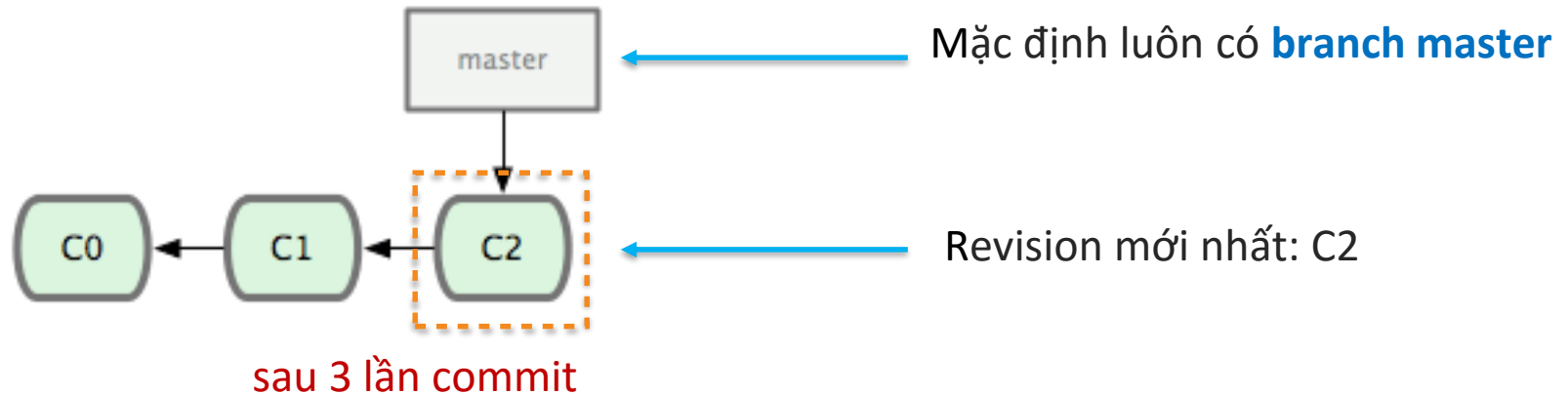




TƯƠNG TÁC VỚI NGƯỜI DÙNG KHÁC

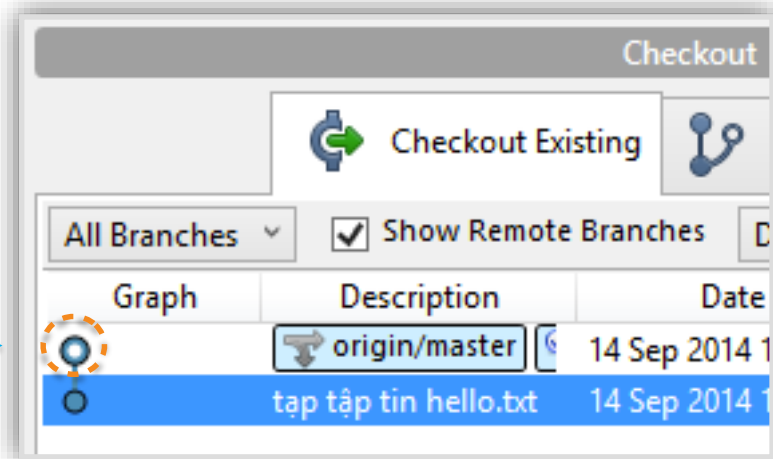
Branch & Merge

❖ Sau mỗi lần commit, master luôn chỉ vào revision cao nhất



Chọn tab log/history để xem đồ thị các revision

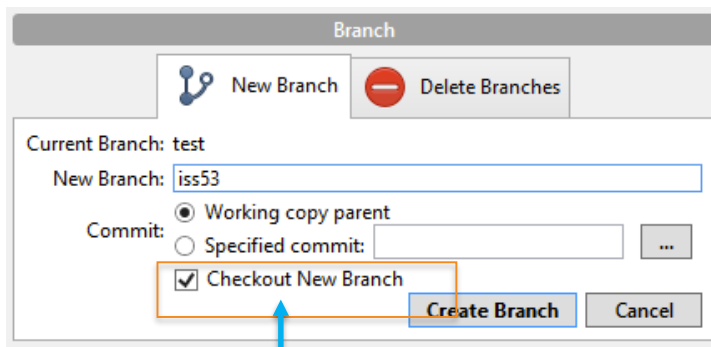
Chấm sáng chỉ vị trí hiện tại



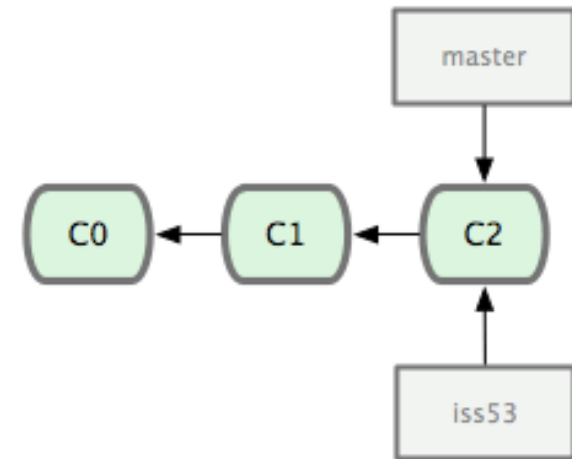
Tạo branch

❖ Sử dụng chức năng branch trong toolbar

tạo branch trong sourcetree

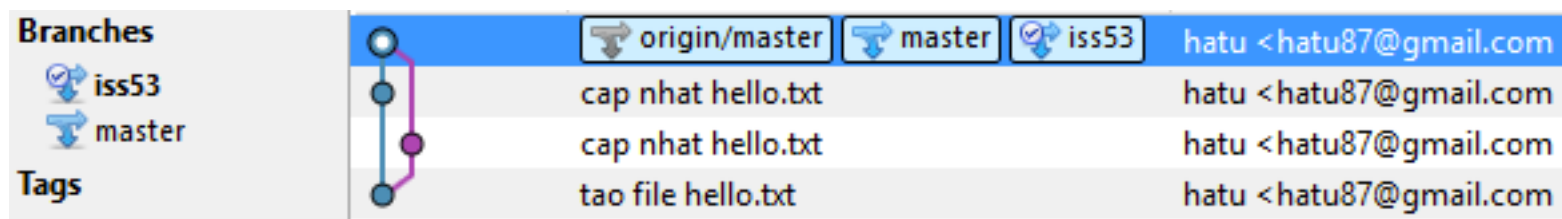


di chuyển đến branch iss53 sau khi tạo



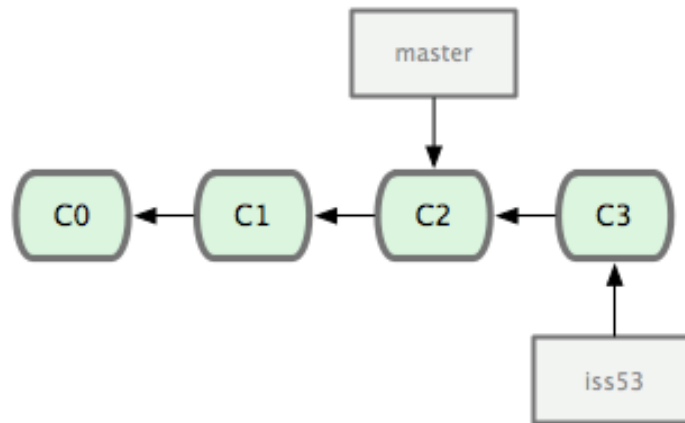
branch mới sẽ tạo con trỏ đến vị trí hiện tại

các branch trong sourcetree



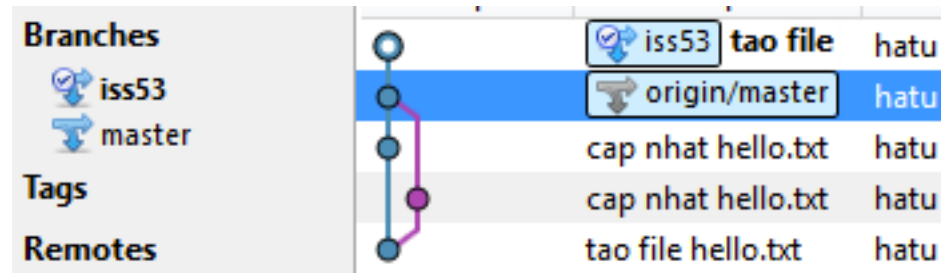
Tạo tập tin mới và thực hiện commit trên branch mới

❖ *Sử dụng chức năng Add và Commit như bình thường*



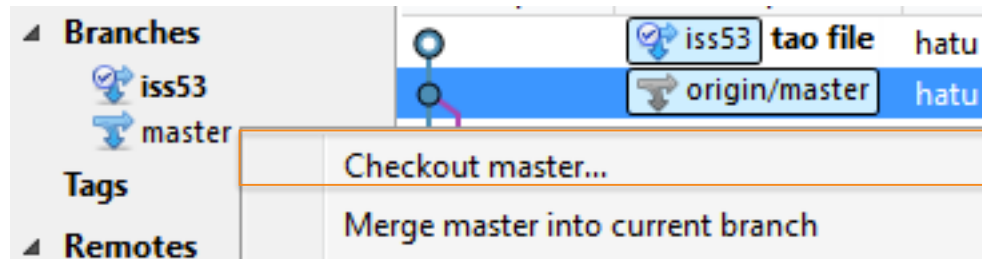
chỉ có con trỏ của branch iss53 di chuyển sau khi commit

các branch trong sourcetree sau khi commit trên branch iss53



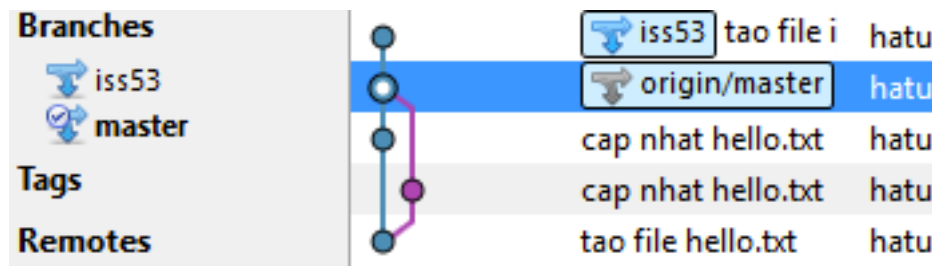
Sử dụng checkout để di chuyển sang branch khác

❖ *chọn branch muốn di chuyển đến và chọn checkout*

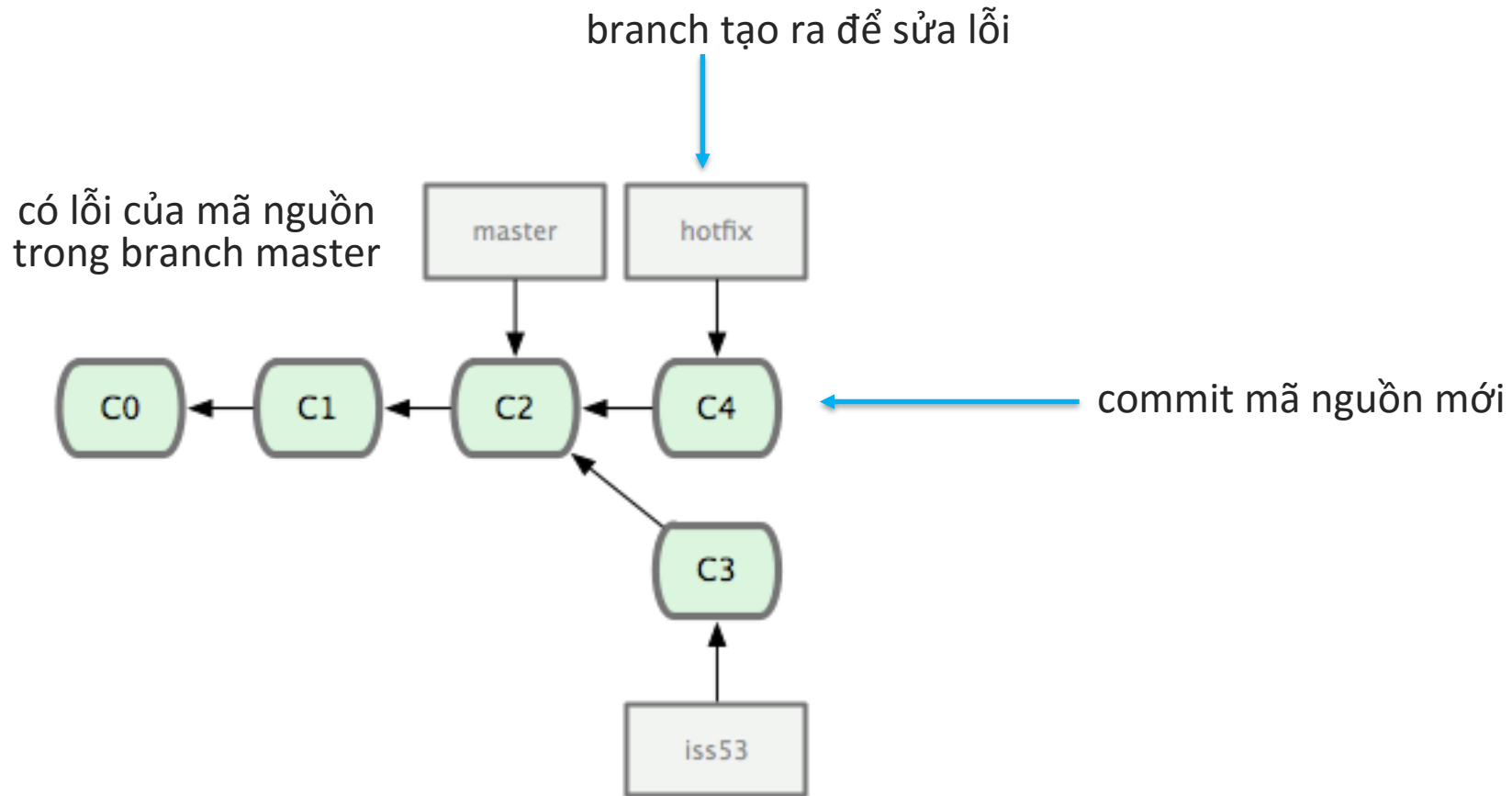


di chuyển đến
branch master

sau khi chuyển sang branch mới

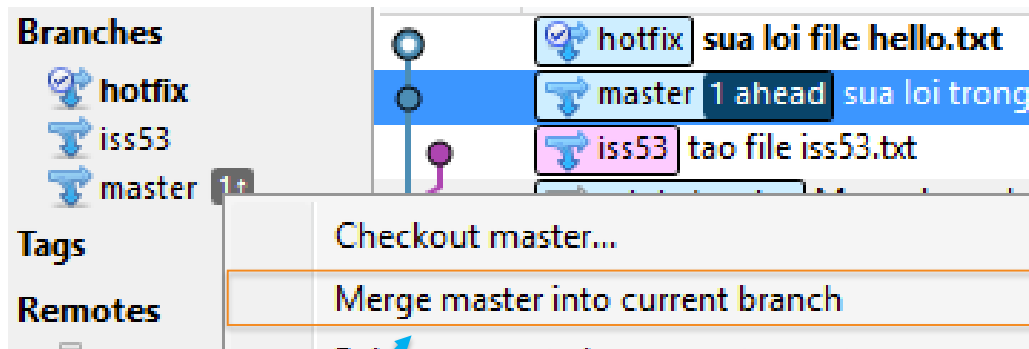


Branch có thể được sử dụng để sửa lỗi

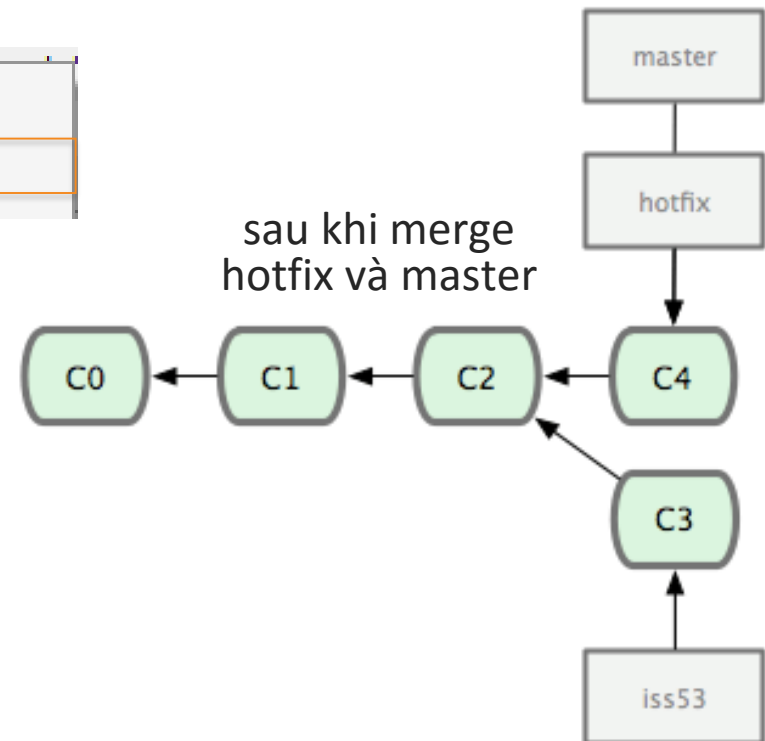


Sử dụng merge để nối branch sau khi sửa lỗi

❖ Sau khi sửa lỗi xong, sử dụng merge để nối branch master và hotfix

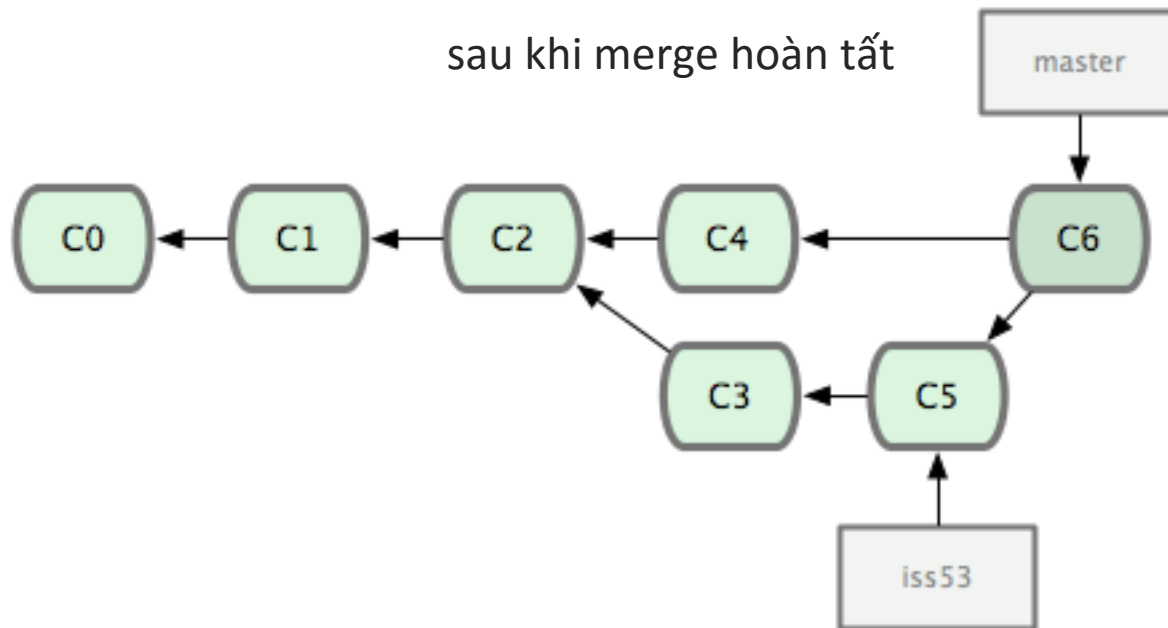


merge trong
sourcetree

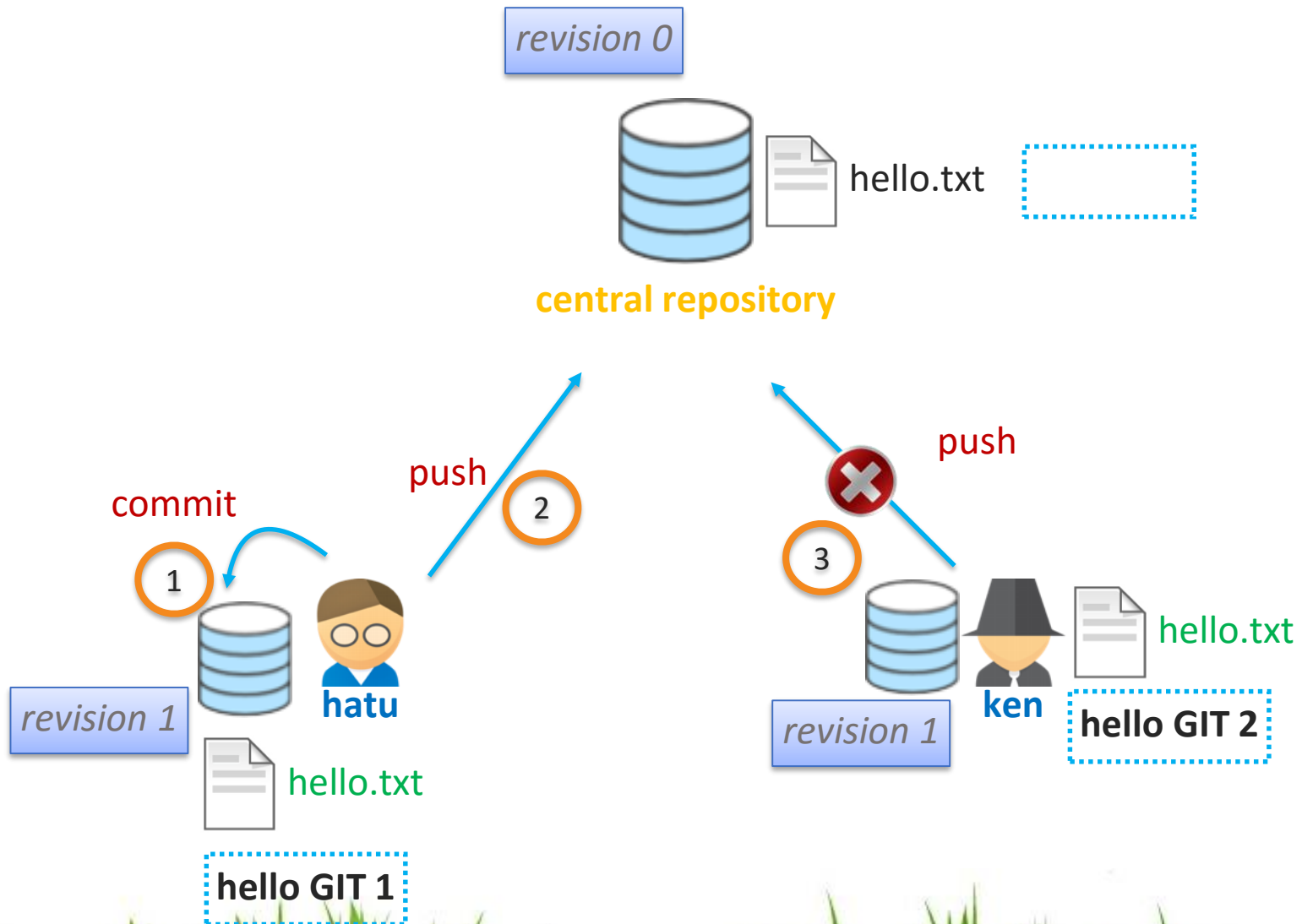


Sử dụng merge để nối branch chức năng và master

❖ Sau khi cài đặt xong chức năng iss53, có thể dùng merge để cập nhật chức năng trên vào master



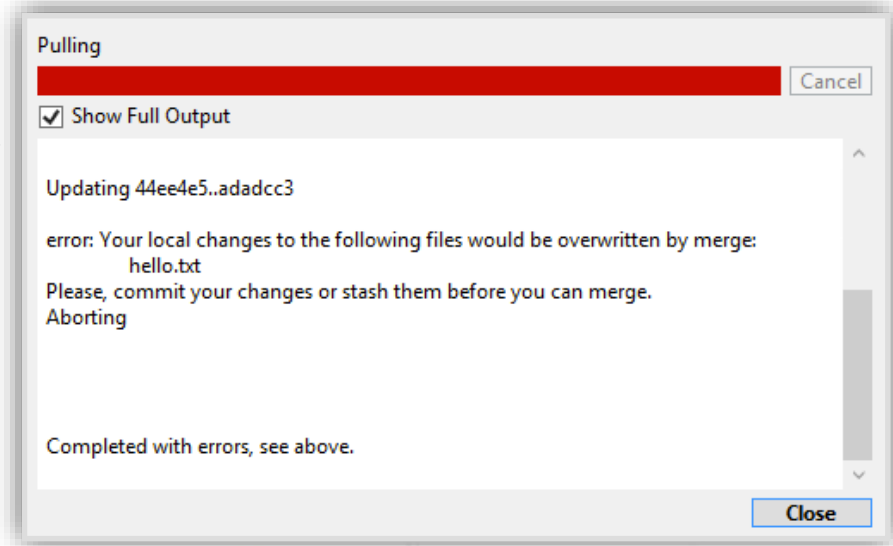
Xử lý tranh chấp (conflict)



Xử lý tranh chấp (conflict)

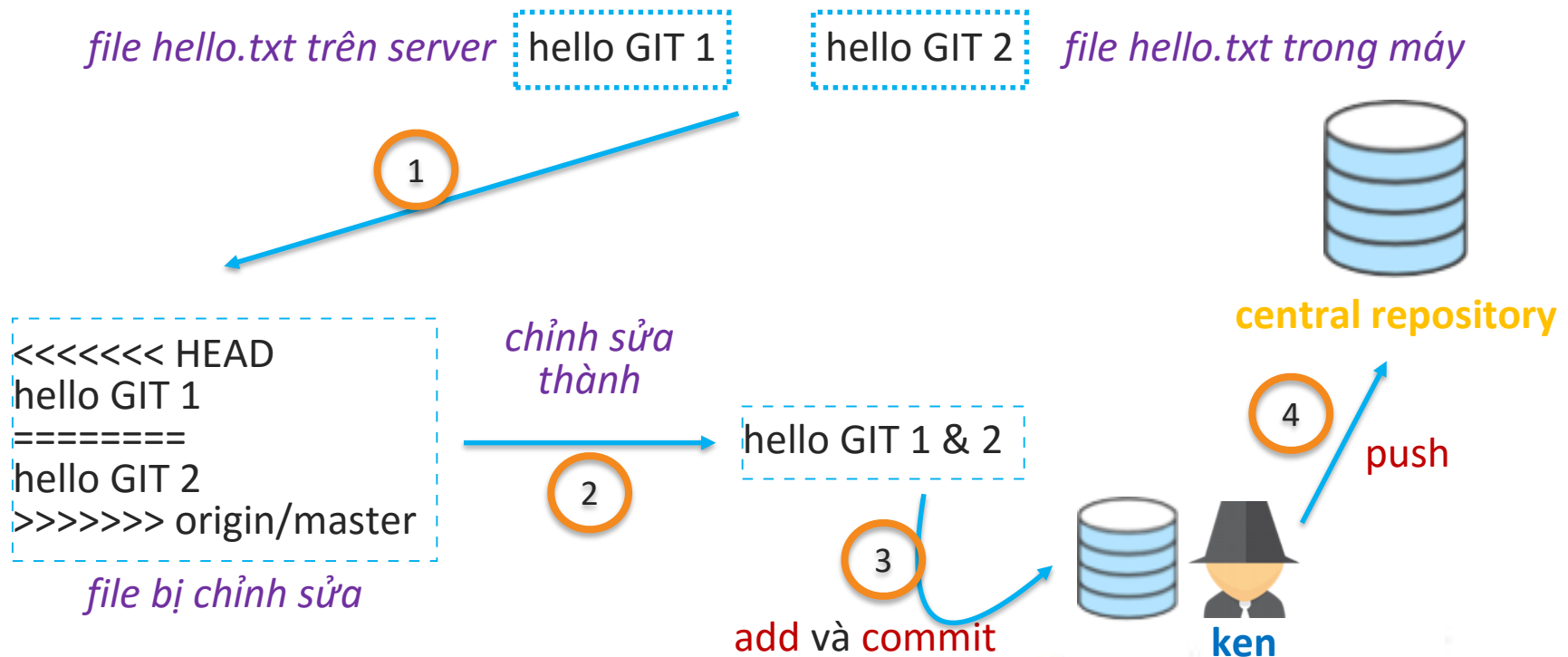
❖ *Khi Pull xảy ra conflict, sourcetree sẽ báo lỗi và tự tạo branch mới cho phần chức năng trên repository.*

❖ *Lúc này, cần merge branch hiện tại (thường là master) và branch mới*



Xử lý tranh chấp (conflict)

❖ Sau khi chọn merge, các tập tin bị conflict sẽ bị thay đổi và đánh dấu phần bị conflict



Các vấn đề khác

◆ *Phục hồi:*

<https://www.atlassian.com/git/tutorials/undoing-changes>

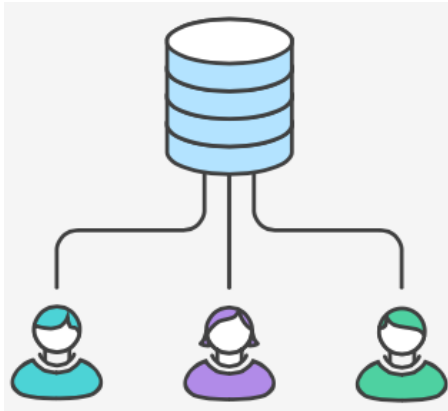
◆ *So sánh các lệnh: fetch, pull*

◆ *Merging và Rebase:*

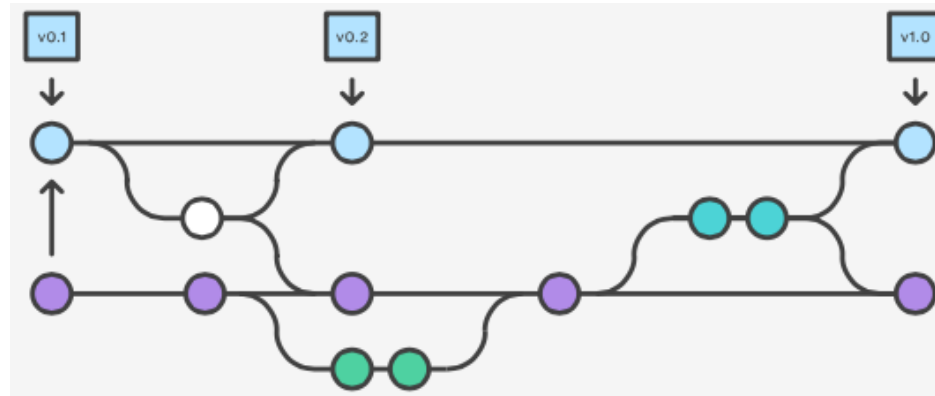
<https://www.atlassian.com/git/tutorials/merging-vs-rebasing>

Một số workflow cho GIT

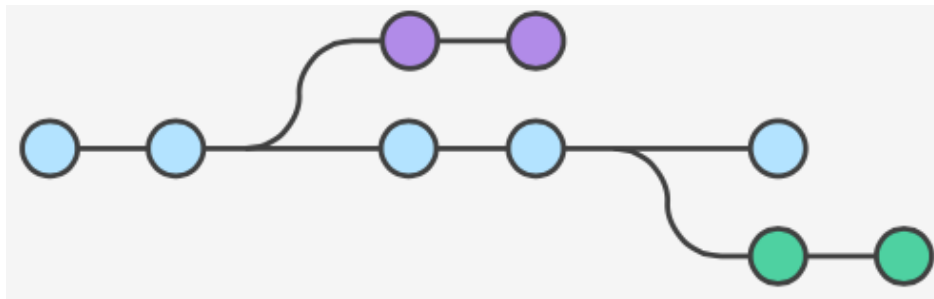
Centralized Workflow



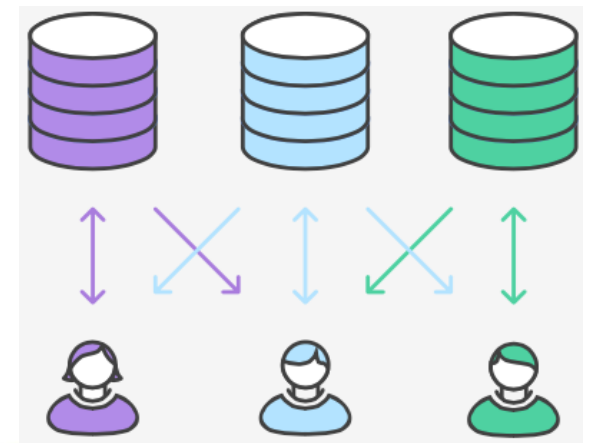
Gitflow Workflow



Feature Branch Workflow



Forking Workflow



<https://www.atlassian.com/git/tutorials/comparing-workflows>

Tài liệu tham khảo

◆ <https://www.atlassian.com/git/tutorials/>

◆ <http://git-scm.com/doc>



CẢM ƠN CÁC BẠN ĐÃ QUAN TÂM THEO DÕI