Page title

Text here

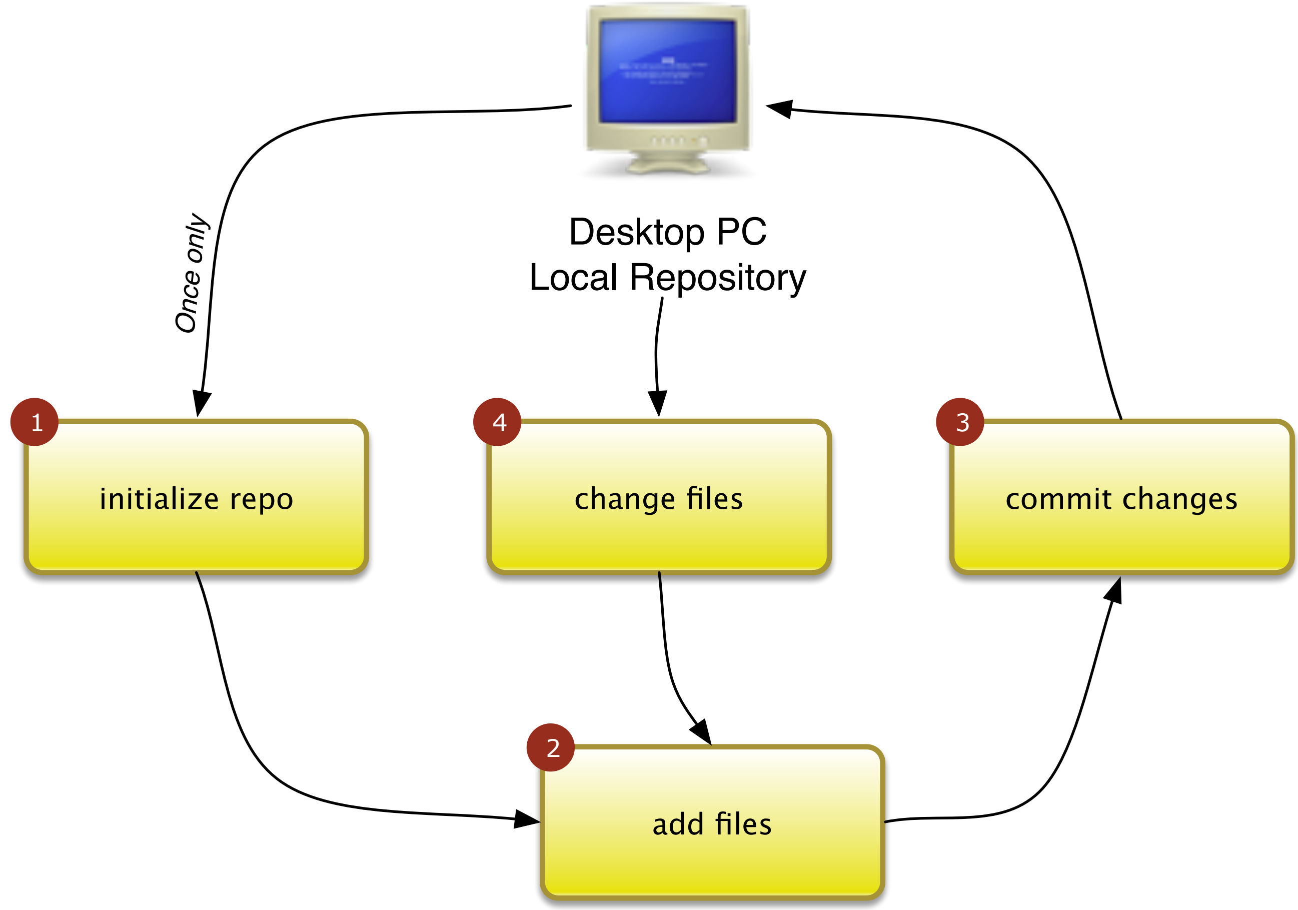
Text here

1 Some long description

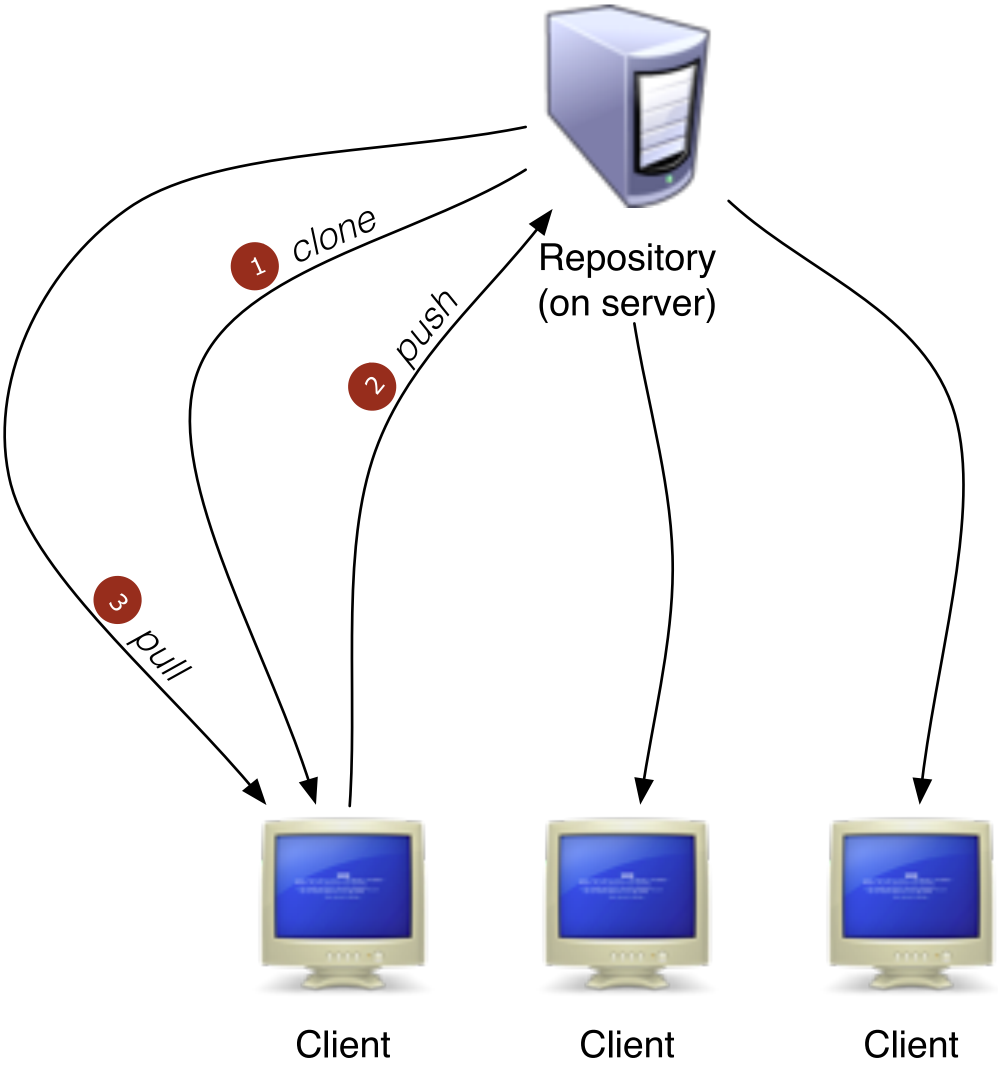
Commands

1 *action*

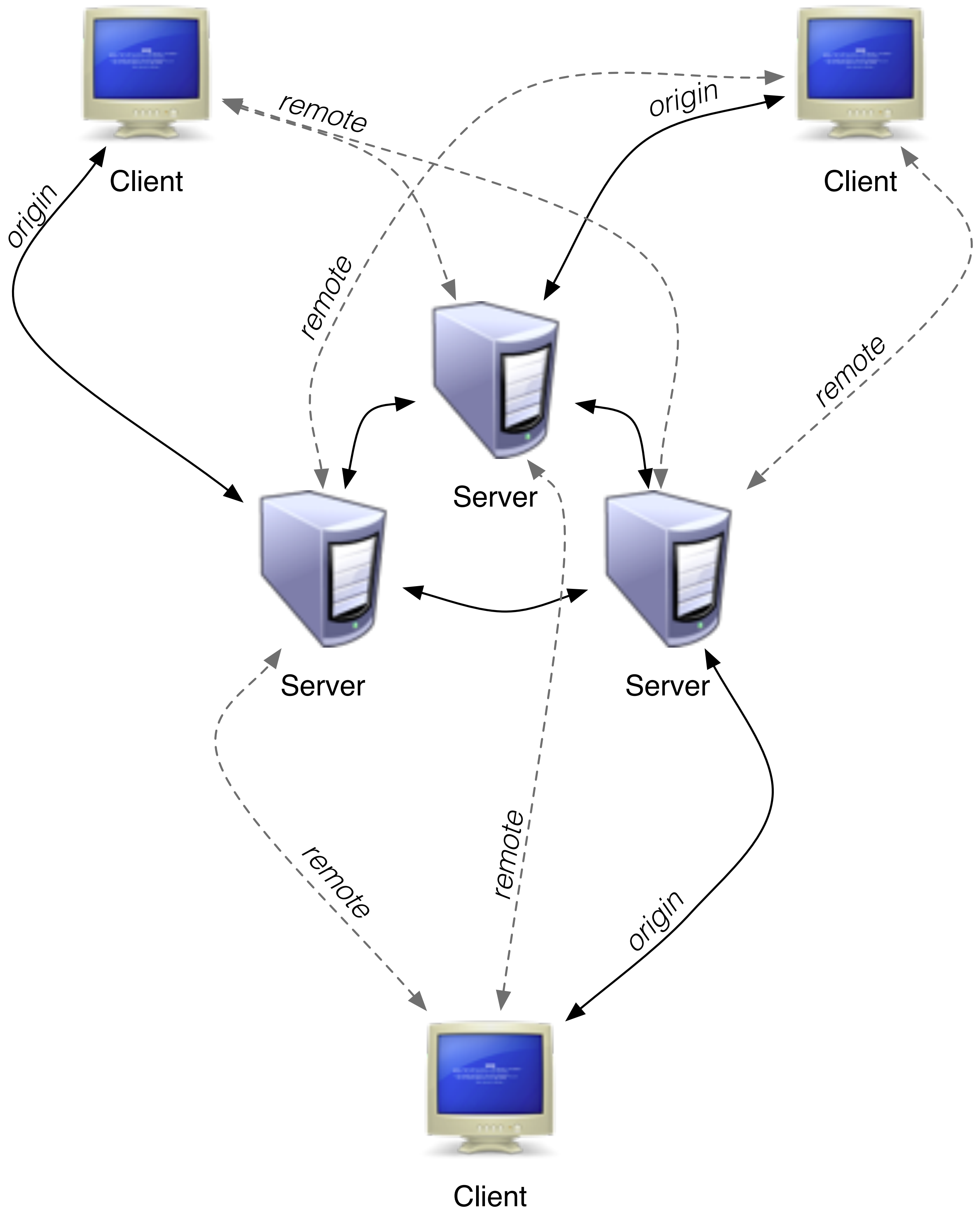
Working with a local repository



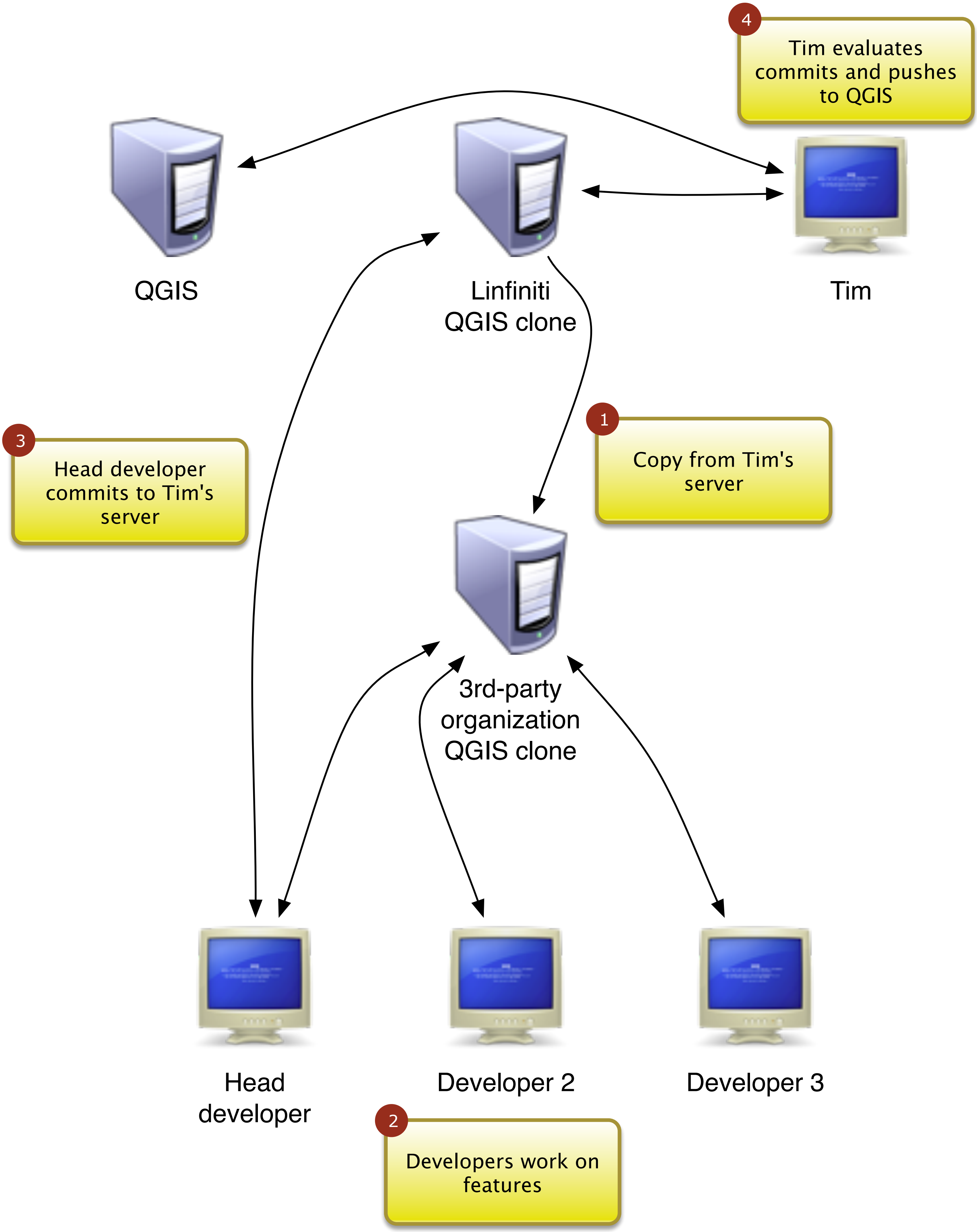
Working with a central repository



Working in a decentralized model



Working in a decentralized model (example)



QGIS

Linfiniti QGIS clone

Tim

3rd-party organization QGIS clone

Head developer

Developer 2

Developer 3

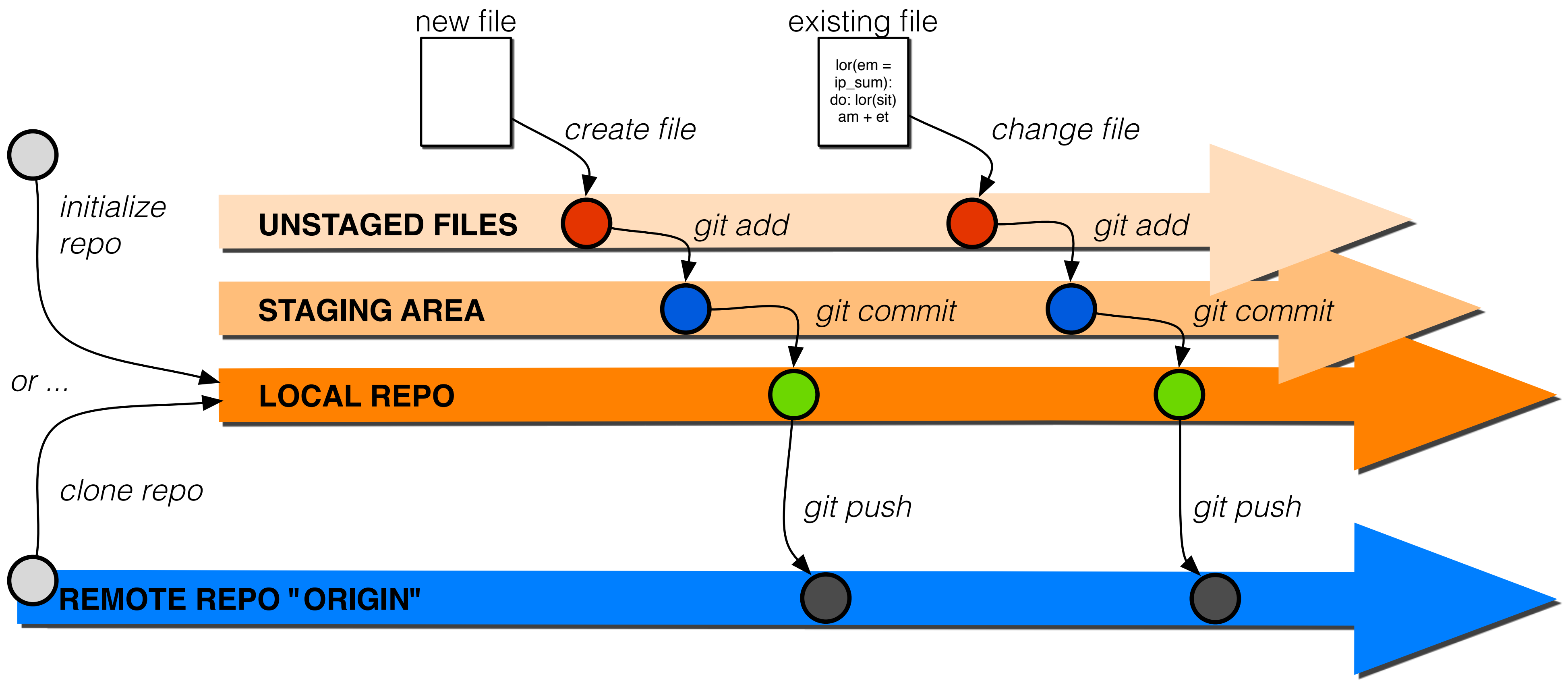
1 Copy from Tim's server

2 Developers work on features

3 Head developer commits to Tim's server

4 Tim evaluates commits and pushes to QGIS

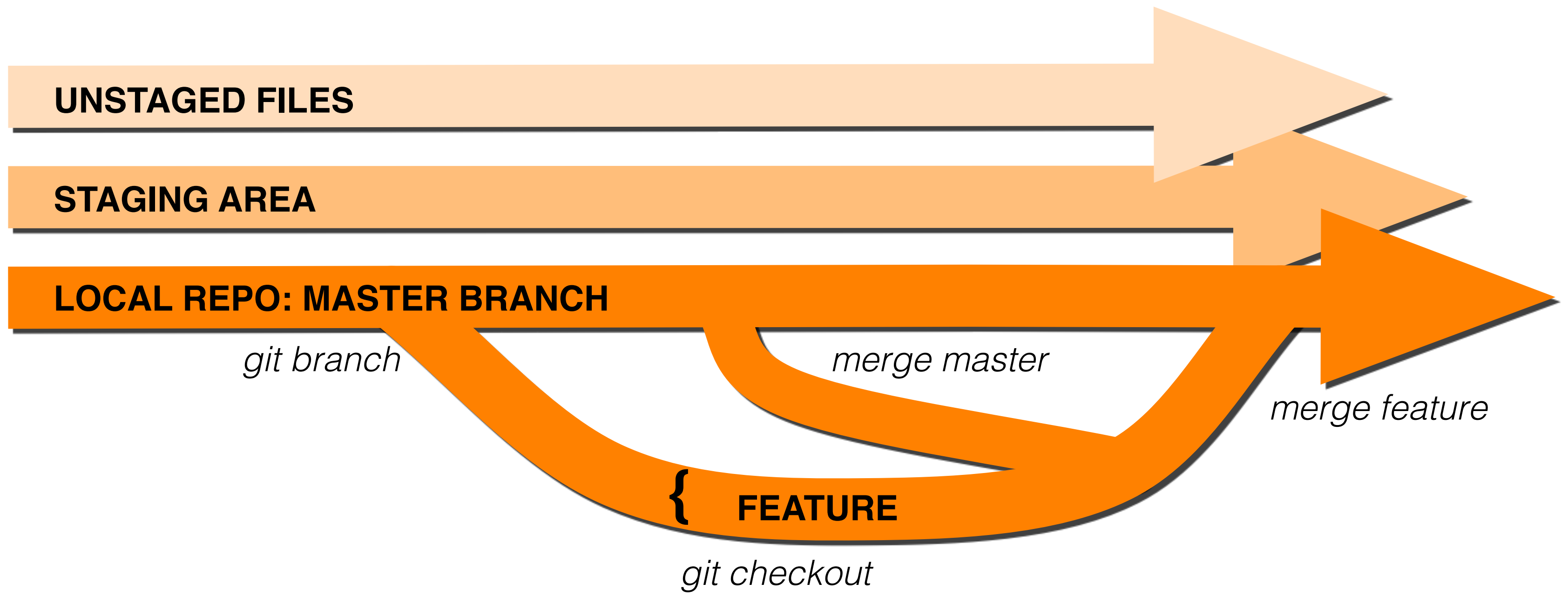
Git workflow: checkout, add, commit & push



Commands

git init: initialize a repository
git clone <url>: clone a repository to make a local copy
git add: stage a file/change
git commit: commit file to local repository
git push: push changes from local repository to origin
git status: see which files are unstaged/staged

Git workflow: branch & merge



UNSTAGED FILES

STAGING AREA

LOCAL REPO: MASTER BRANCH

git branch

merge master

merge feature

{ FEATURE

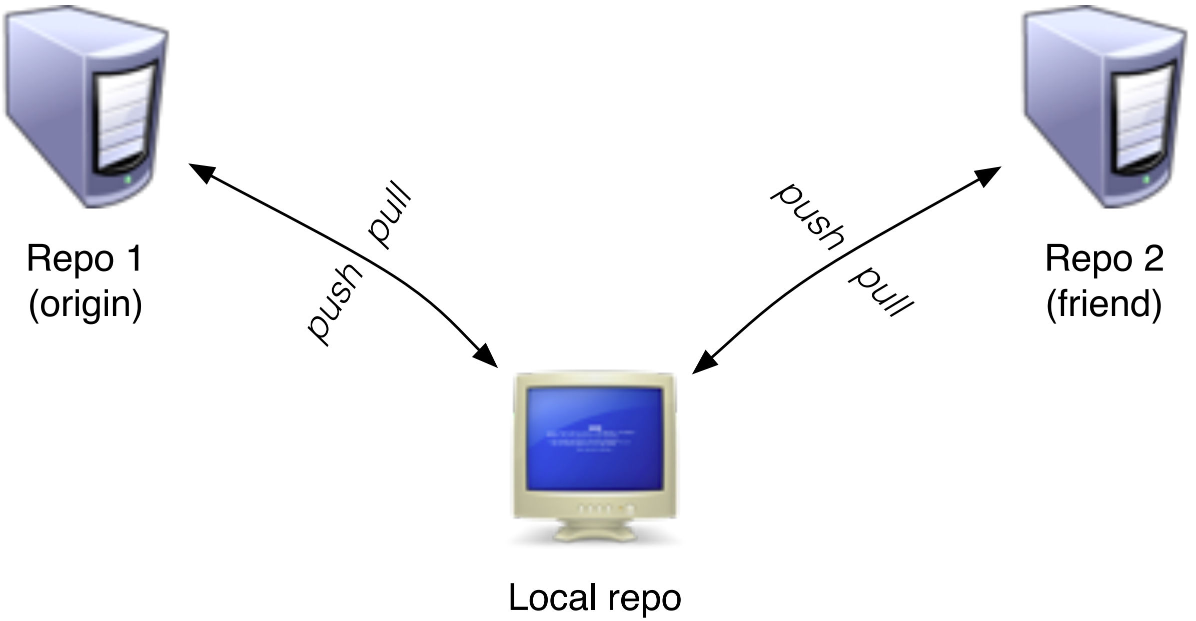
git checkout

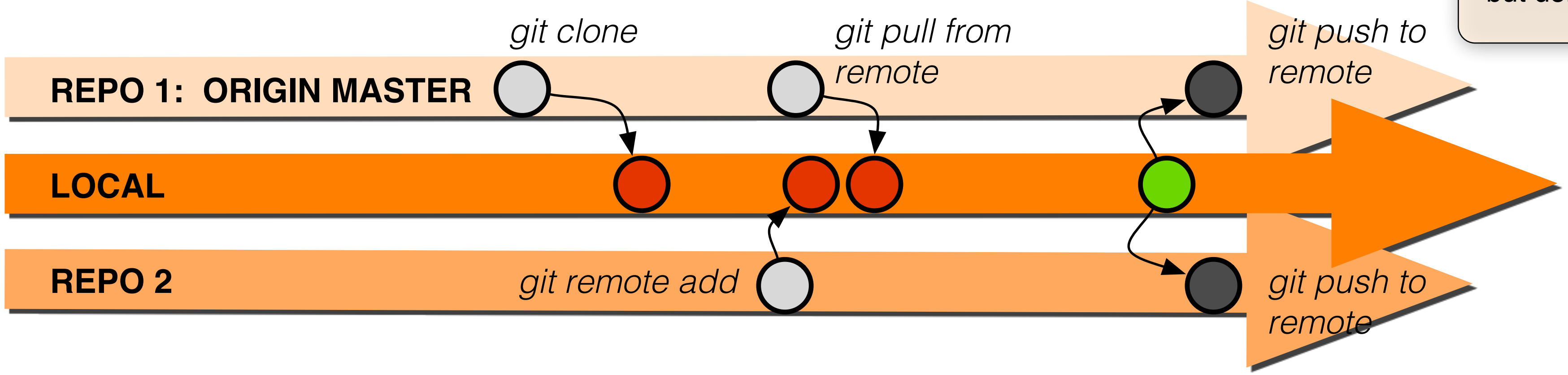
Commands

git branch: create a local branch in your repository
git checkout <branch>: start working in a branch
git checkout master: return to master branch (staging area must be clean)

git merge <other branch>: bring changes from "other branch" into your branch**git branch -D <unwanted branch>:** delete a branch
git branch: list branches (current shown with A)

Git workflow: remotes





Commands

git clone: copy remote repo

git remote add <nickname> <remote url>: register a new remote repo

git remote -v: list remote repos

git pull <nickname> <branch>: fetch and merge changes from remote repo

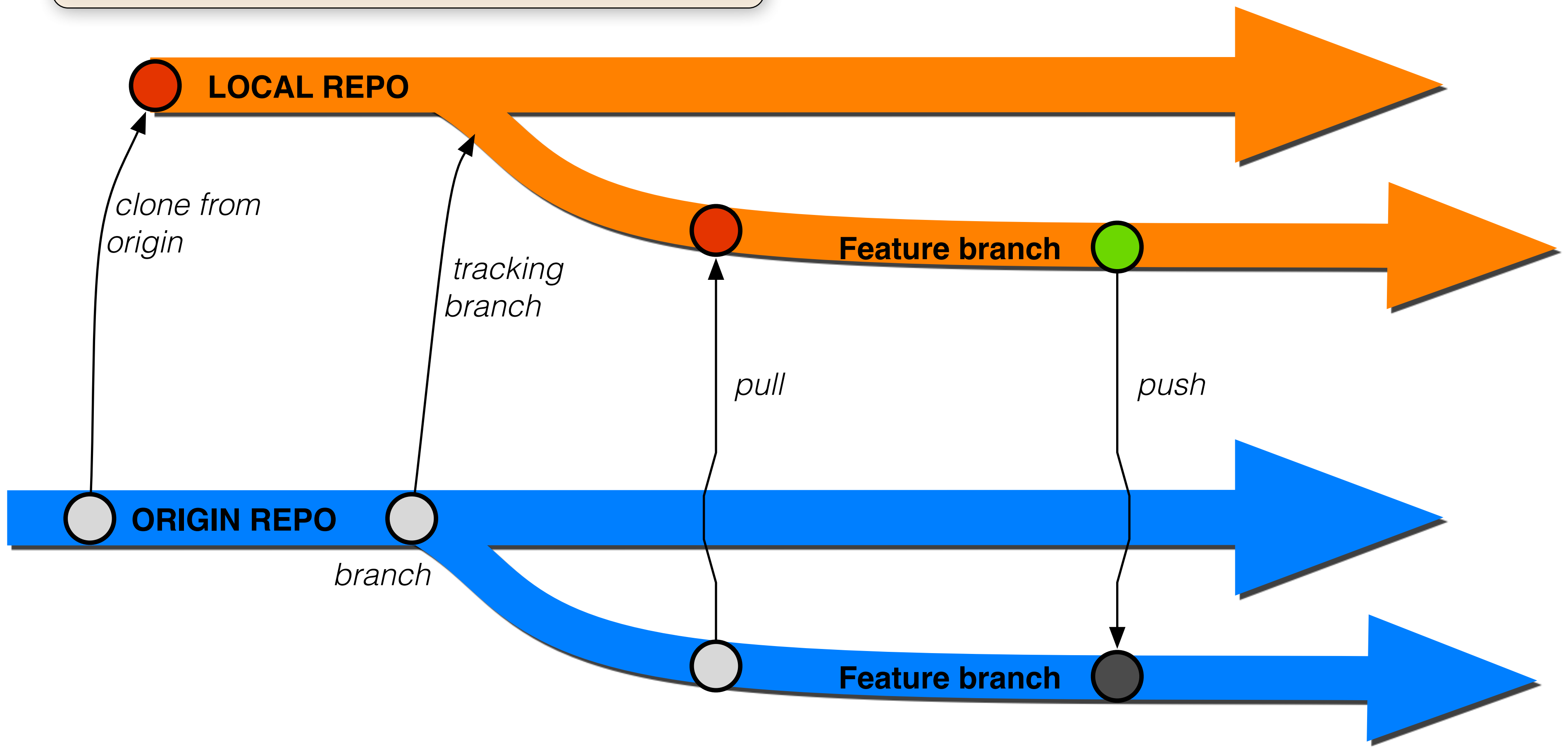
git push <nickname> <branch>: push local changes to remote repo

git fetch <nickname>: fetch history from remote but don't merge

Git workflow: tracking branches

Commands

git branch --track <remote nickname> <branch>:
create a branch and synchronize it with a remote branch
git branch -r: list all remote branches
git pull: fetch changes from remote branch
git push: put local changes into remote branch



Git handy commands

Commands

git add: put a file onto staging area or mark locally changed files for staging

git rm: delete a file locally and mark for removal from the repo

git mv: move a file locally and in the repo on next commit (retaining its history)

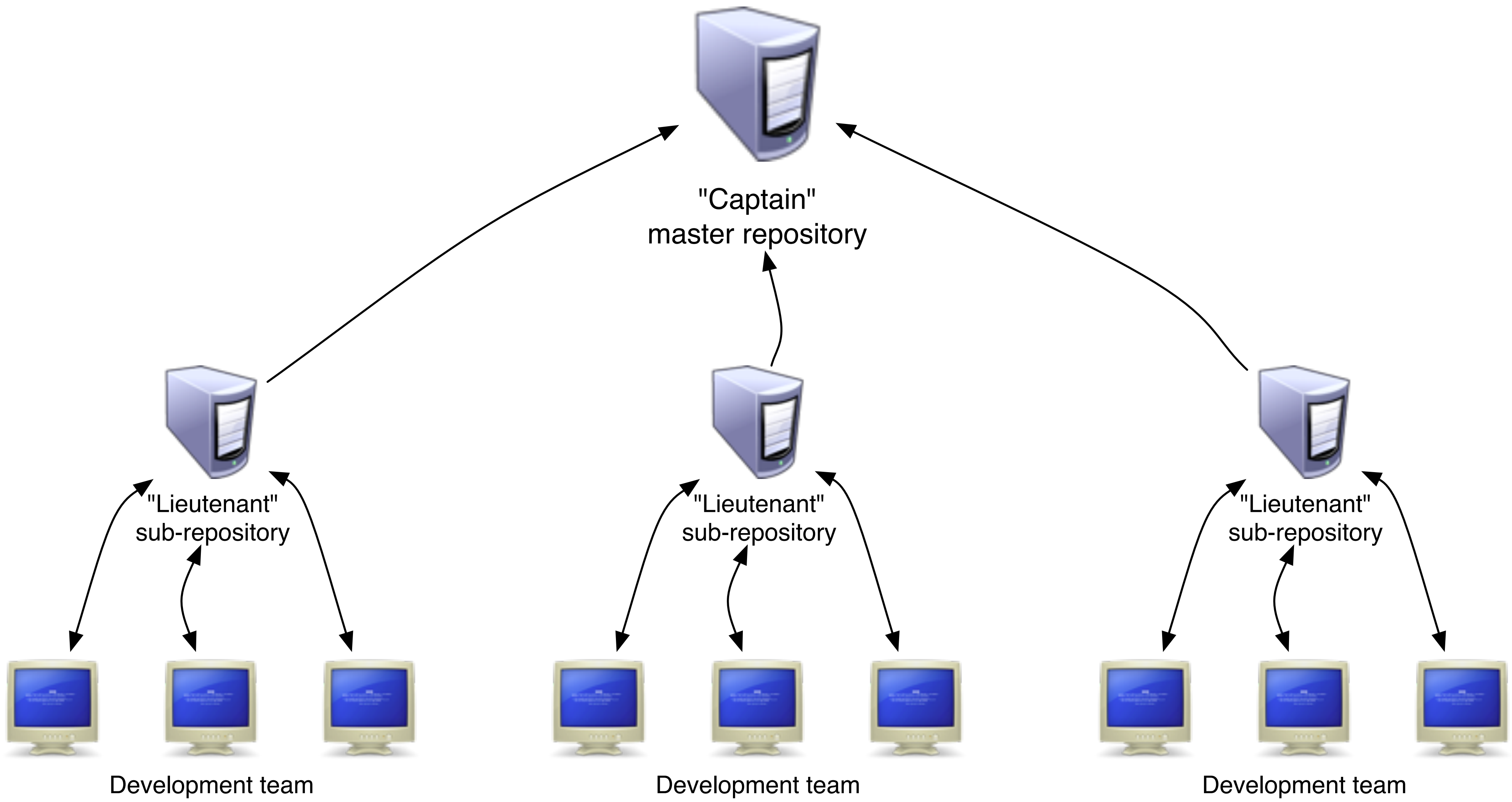
git diff: view difference between local staging and repo's version

git mergetool: use a graphical to view and resolve merge conflicts

git blame: see a log of who has changed a file

git log: see a history of changes

Working in a hierarchical model



"Captain"
master repository

"Lieutenant"
sub-repository

"Lieutenant"
sub-repository

"Lieutenant"
sub-repository

Development team

Development team

Development team