GIT

A distributed version control system

By:
Fahad P H
Broadsoft Design

GIT

- Why Git ?
- Download and install and Configure Git
- Introduction to Git
- Git common commands
- Q & A

Why Git ?

- Distributed version control system
- Have branches
- Stores content as metadata

Download and install Git

- Here’s the standard one GIT

  http://git-scm.com/downloads

- merge tool (meld)

  http://meldmerge.org

- GUI tool (sourcetree)

  https://www.sourcetreeapp.com/

Configure Git

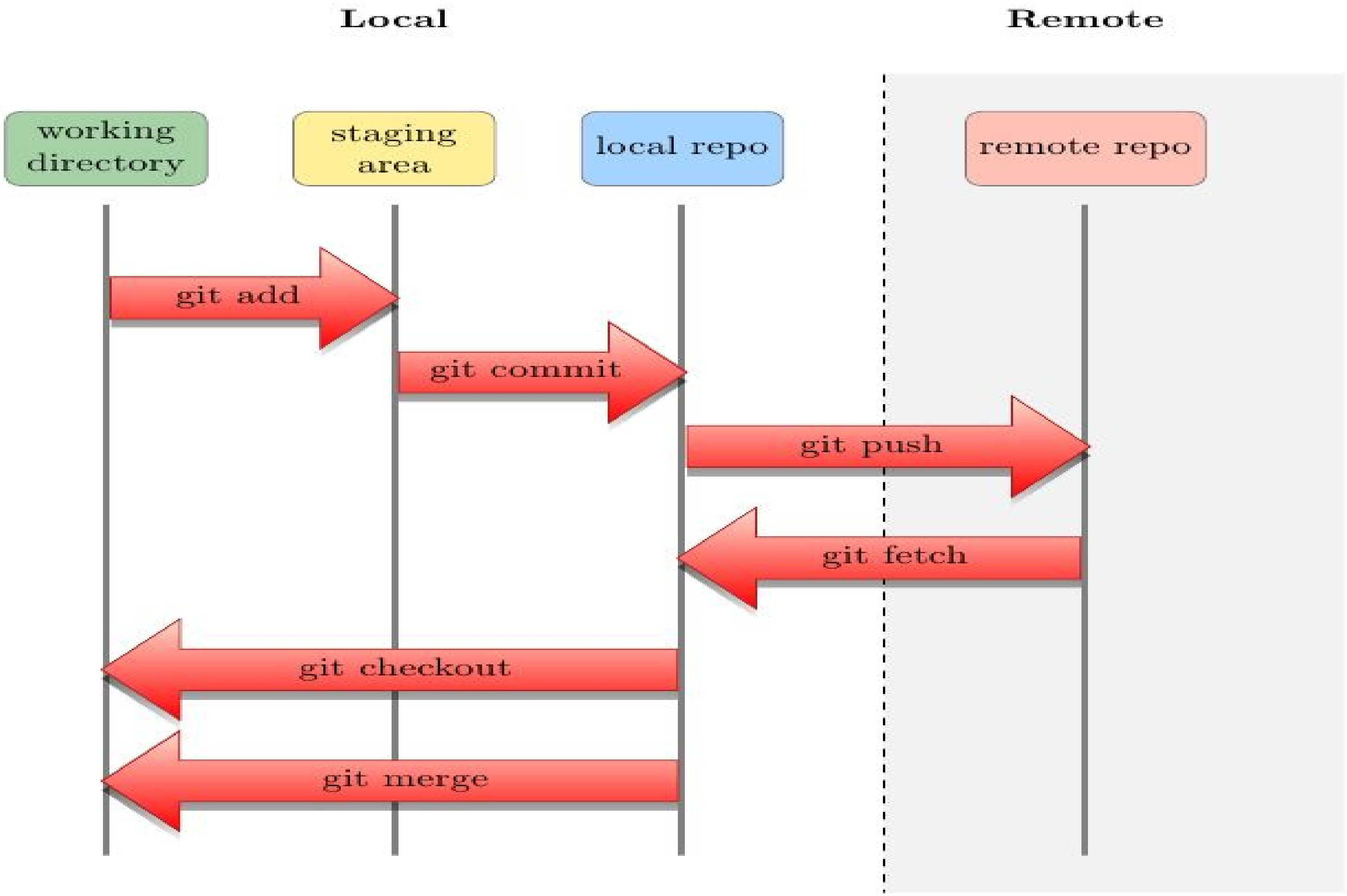
Enter these lines (with appropriate changes)

- git config --global user.name "fahad"
- git config --global user.email fhaneefa@broadsoft.com

If you want to use a different name/email address for a particular project, you can change it for just that project

- *cd* to the project directory
- Use the above commands, but leave out the --global

Introduction to Git



Working with command-line

Create and fill a repository

1. cd to the project directory you want to use

2. Type in git init
   note : This creates the repository (a directory named .git)

3. Type in git add . or git add <file-name>
   note : git add . adds all your current files to the repository

4. Type in git commit –m "Initial commit"
   note : You can use a different commit message, if you like

Working with Branches

1. Type in git branch branch1

   note: to create branch type git branch <branch-name>

2. Add some changes to the file add commit those changes

   note: git add <file-name> - for adding

   git commit -m”1st commit in branch1” - for commit

3. Type in git checkout master

   note: git checkout <branch-name> - for going to corresponding branch

4. Type git pull . branch1

   note:- git pull . <branch-name> - for getting all code from local branch broadsoft

Common git commands

- git init :- for initialization
- git clone :- for clone from remote url
- git add <file-name> :- for adding file
- git commit -m”” :- for commit
- git log :- for log
- git log --stat : for detailed log
- git reflog :- for detailed log with revert info
- git checkout :- for moving to branch and removing file change
- git branch :- for listing branch and creating
- git pull :- for get the changes
- git push :- for pushing chnages
- git revert :- for revert the commit and changes
- git reset :- for reset the change
- git stash :- for saving temp info
- git stash pop :- for getting stash info
- git difftool :- for getting file difference
- git mergetool :- for merging

References

- https://www.atlassian.com/git/
- https://www.sourcetreeapp.com/
- https://www.siteground.com/tutorials/git/commands.htm

Q
&
A

Thank
You