

형상 관리를 위한 Git & Github

Chapter 1

Git을 이용한 포트폴리오 형상관리

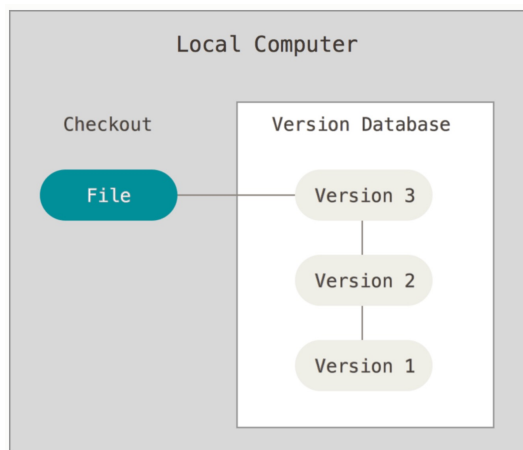
Chapter 1

Git을 이용한 포트폴리오 형상관리

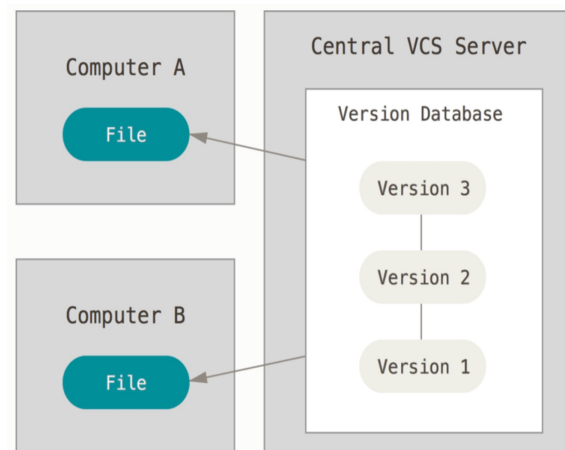
Git & Github 개요

버전 관리

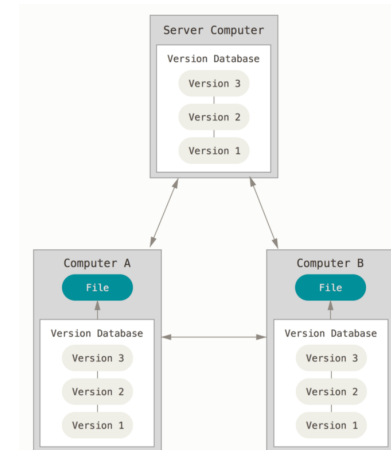
- 파일의 변화를 시간에 따라 기록하여 특정 시점의 버전을 꺼내 오는 시스템



로컬 버전관리



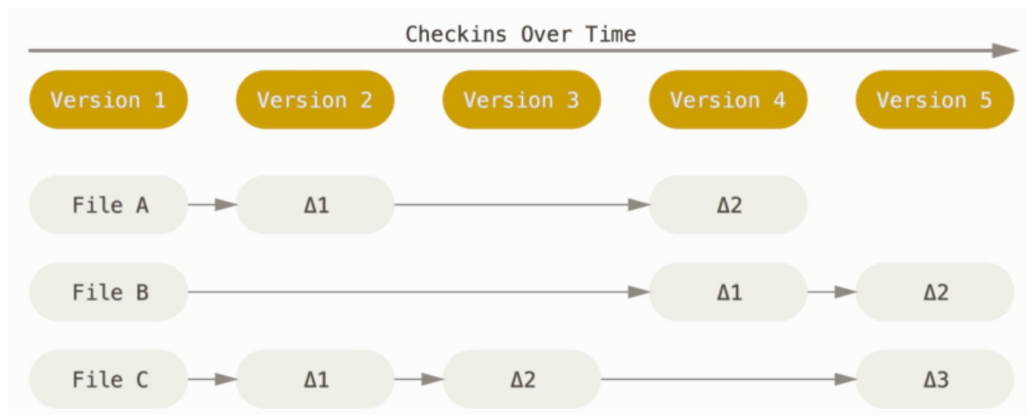
중앙 집중식 버전관리



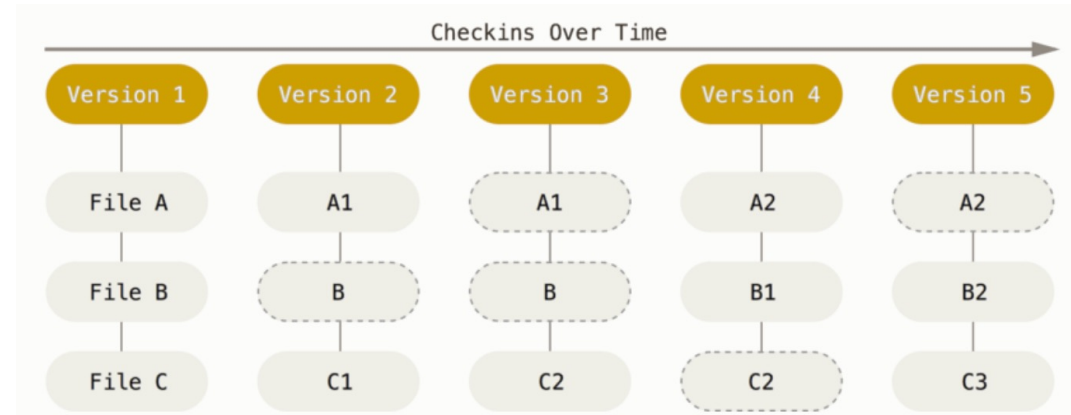
분산 버전 관리

Git

- 기존 VCS : 파일의 메타데이터와 변경 정보를 별도로 관리
- Git : 해당 해당 변경 시점의 최종 시점을 저장하고 해당 스냅샷을 포인터로 관리



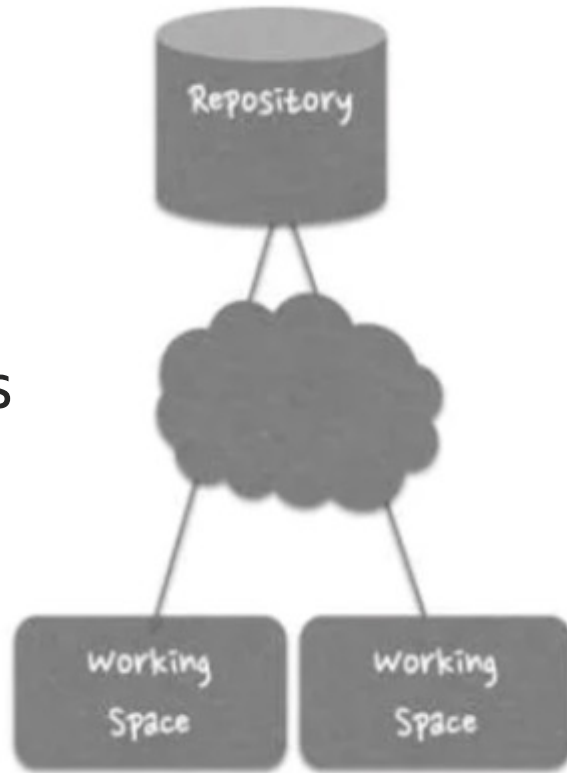
기존 VCS



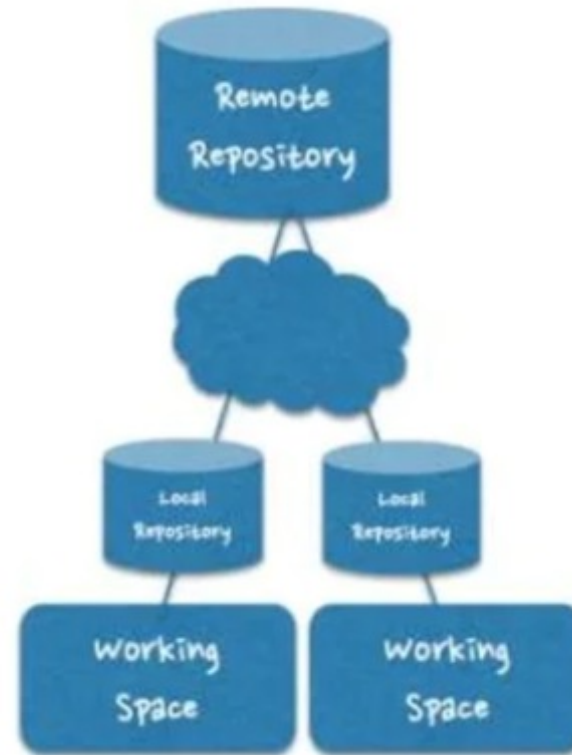
Git

Git 구조

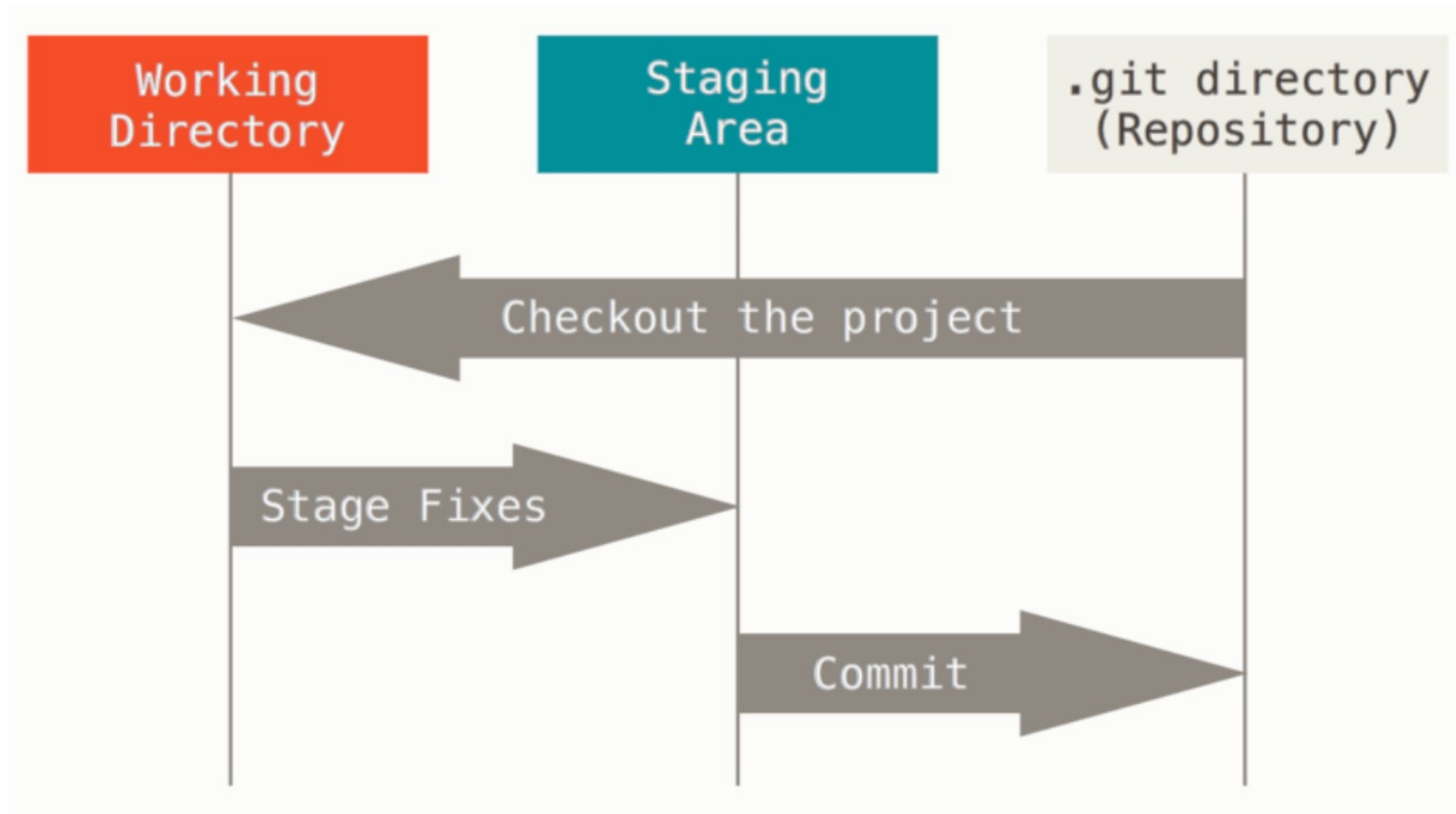
기존 VCS



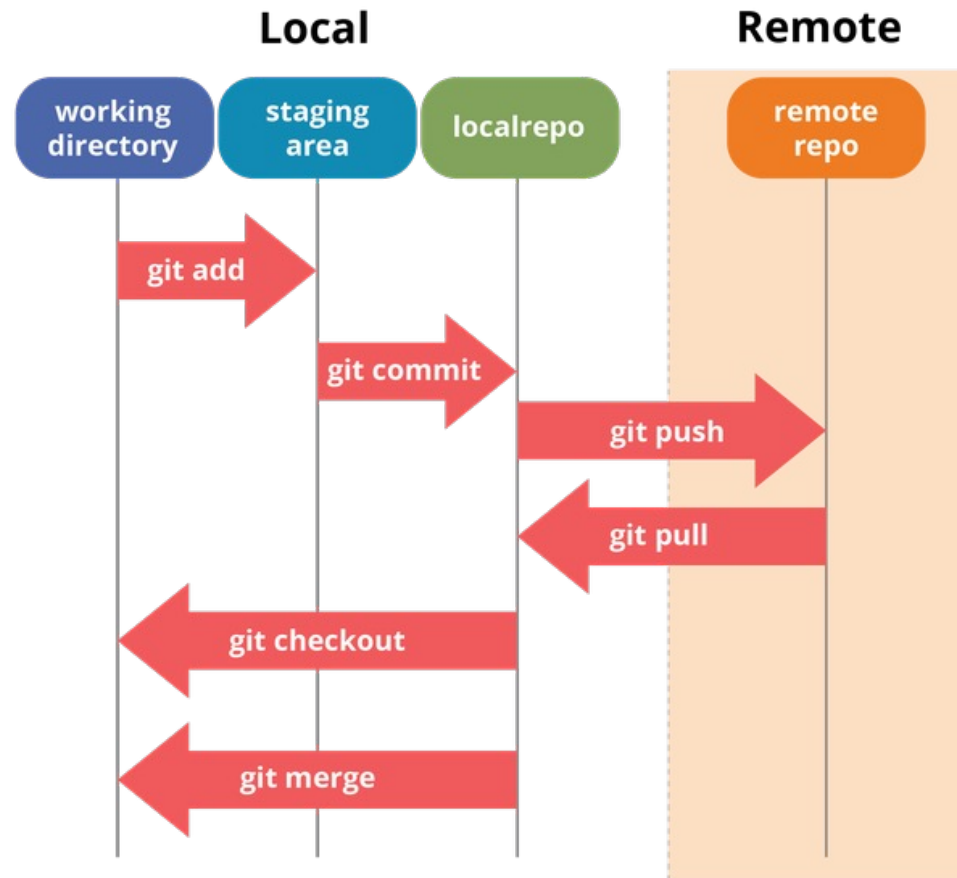
Git



Git 동작 원리



Git Command (1)



Git Command (2)

- `git branch` : 브랜치 목록 보기
- `git merge` : 현재 브랜치에서 다른 브랜치의 수정사항 병합
- `git reset` : 특정 commit을 삭제
- `git cherry-pick` : 다른 브랜치에서 특정 commit을 가져 옴
- `git status` : commit 전 현재 상태를 보여줌
- `git tag` : 특정 commit을 식별
- `git log` : 현재까지의 commit 이력을 보여줌