

## Chapter 1. 버전관리

## Chapter 1-1. 버전관리

버전관리 시스템(형상관리)

- Configuration Management Systems
- Version Control Systems

## 1-1. 버전관리

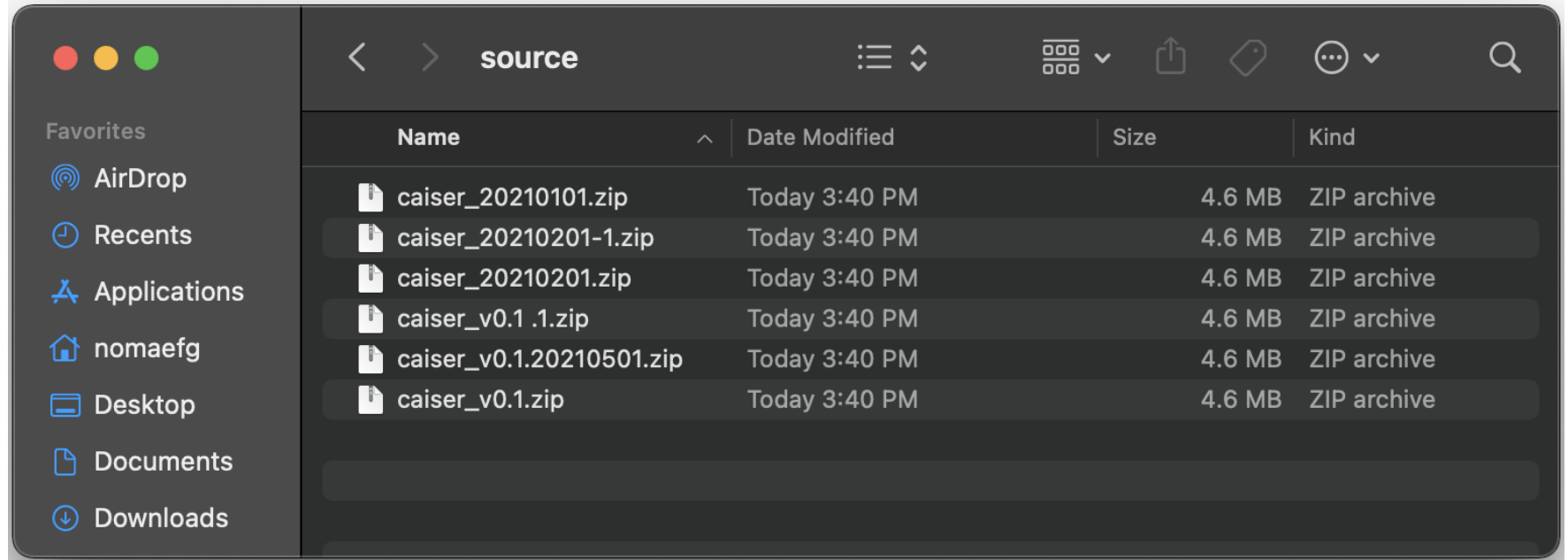
### 버전관리

- Source Data + History
- 협업, 작업추적, 복구 등이 가능

## Chapter 1-2.     Git 등장 배경

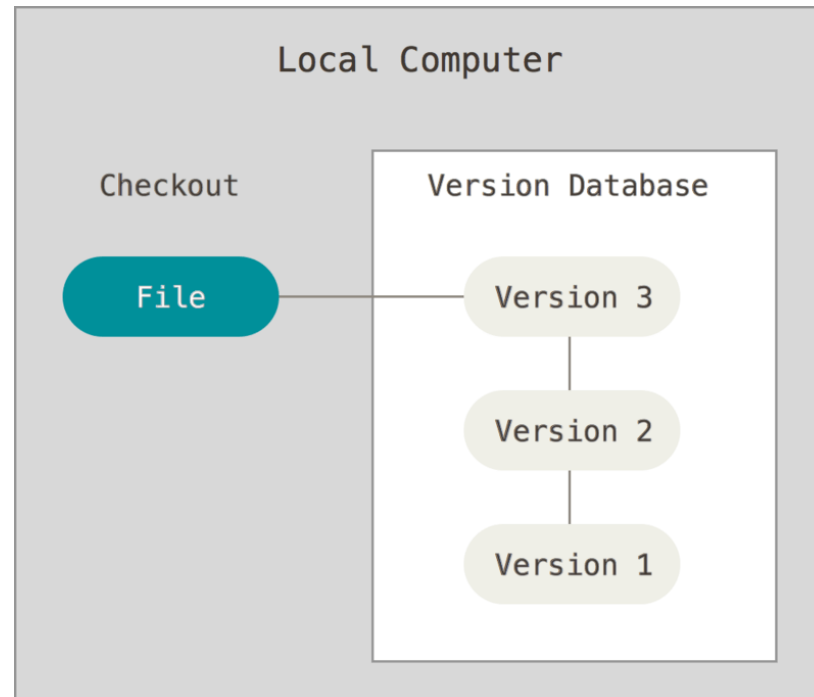


아주 오래전...



- Source Folder + 실행파일을 버전별로 카피하여 관리
- 하루종일 개발한 코드가 컴퓨터가 다운 되면서 날아가버림..(멘붕)

그래서 파일의 버전을 관리하려고...



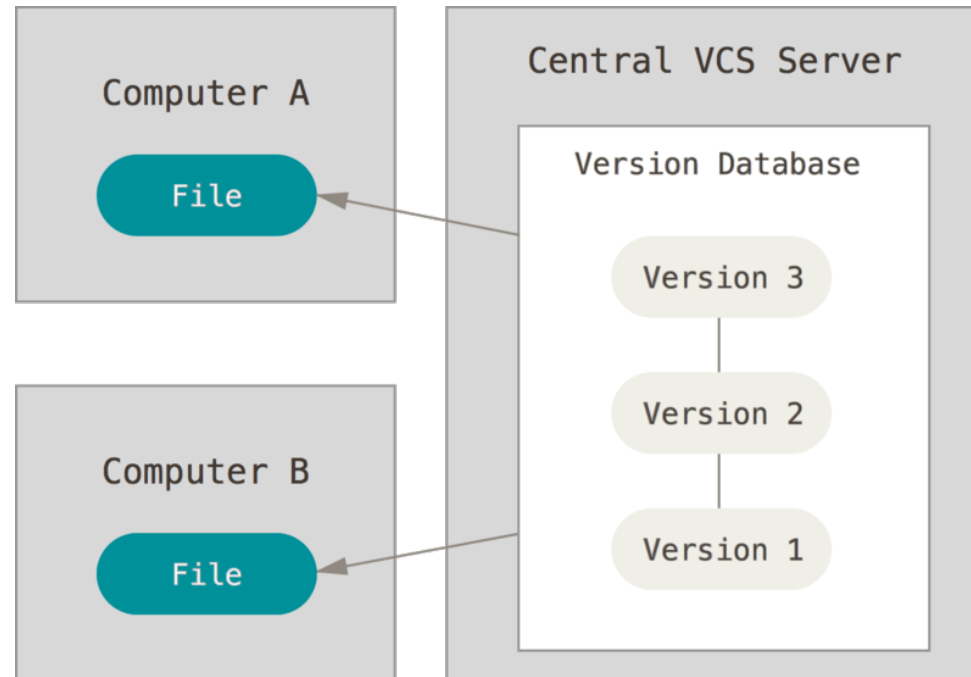
***Local Version Control Systems***

### Local Version Control Systems

- 내 컴퓨터에서 버전 관리 가능 → 내 컴퓨터 하드가 날아가면 전체 코드 사라짐
- 버전은 관리되지만, 협업은 여전히 어려움



그래서 중앙에서 관리해보자

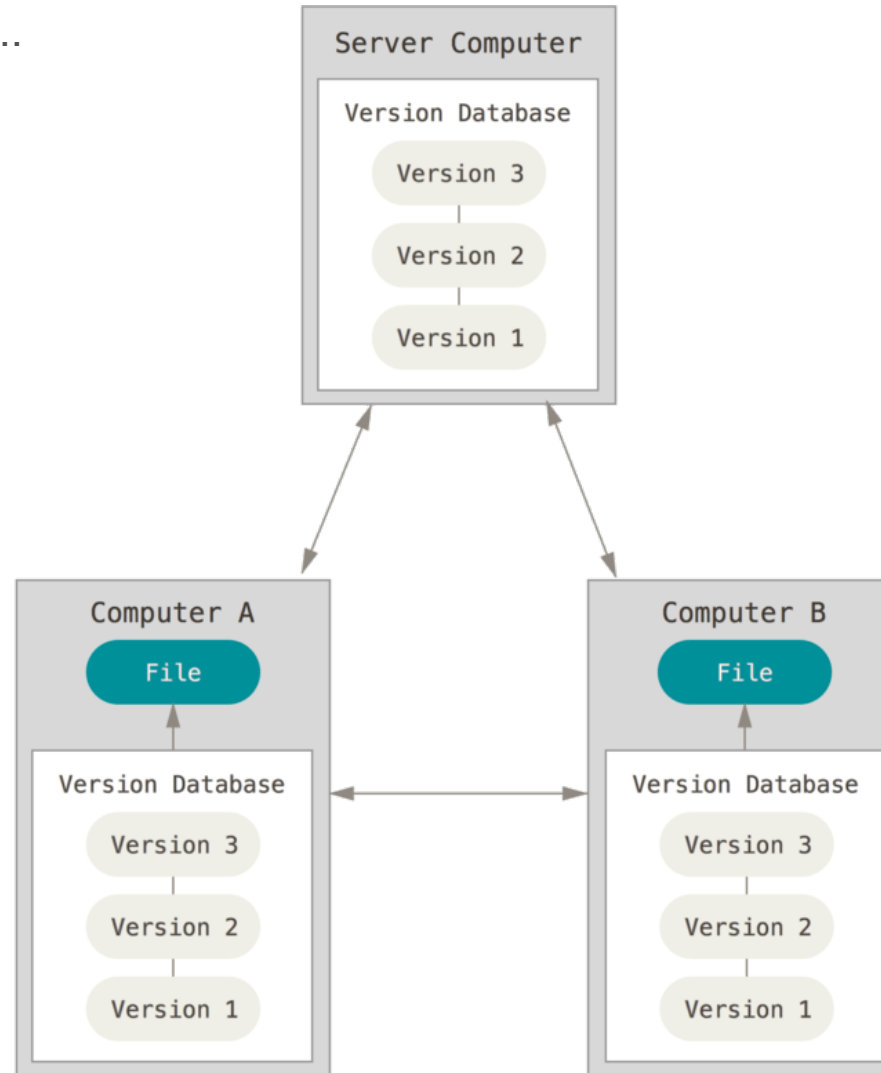


***Centralized Version Control Systems***

### Centralized Version Control Systems

- 협업이 가능해짐
- commit 하는 순간 배포되어 다수에게 버그 유발 가능 (서버로 바로 commit)
- 인터넷이 안되면 작업이 불가능
- 자신만의 version history를 가질 수 없음

그래서 개발되었다는...



### Distributed Version Control Systems

- commit 하더라도 개인저장소 내에 적용됨 (다른 개발자에게 영향 없음)
- 원하는 순간에 배포(Push) 가능
- 오프라인에서도 작업 가능
- 자신만의 version history를 가짐

## Chapter 1-3. 버전관리 시스템의 종류

## 1-3. 버전관리 시스템의 종류

### 버전관리 시스템의 종류

- CVCS - CVS, SVN, etc.,
- DVCS - Mercurial, Git, etc.,

## CVS

- 1980년대 만들어진 형상관리 시스템
- commit 중 오류 발생 시 Rollback 이 되지 않는 등의 문제
- 이후 SVN 으로 대체됨

## SVN

- <https://subversion.apache.org/>
- 2000년대 만들어졌고, 현재까지 두루 사용 중



### Git

- <https://git-scm.com/>
- SVN 보다 빠른 속도와 많은 기능을 지원
- 현재 많은 기업이 사용 중

요즘 기업들은...

- 대부분 SVN 혹은 Git 사용 중

## Chapter 1-4.     Git 기반의 서비스

### Github

- <https://github.com>
- Git 을 호스팅 해주는 웹 서비스, 협업을 위한 기능을 제공
- 참고 - 소스코드 보안이 중요한 경우 사용을 기피함

### Gitlab

- <https://gitlab.com>
- 설치형 버전관리 시스템 - 소스코드 보안이 중요한 기업에서 주로 사용
- 클라우드 버전 관리 시스템 - 10명 이하 무료 (Github 와 유사)
- Issue tracker, Git Remote Repository, API, Team, Group 기능 제공

zero-base /

Life Changing Education

Good Job!