



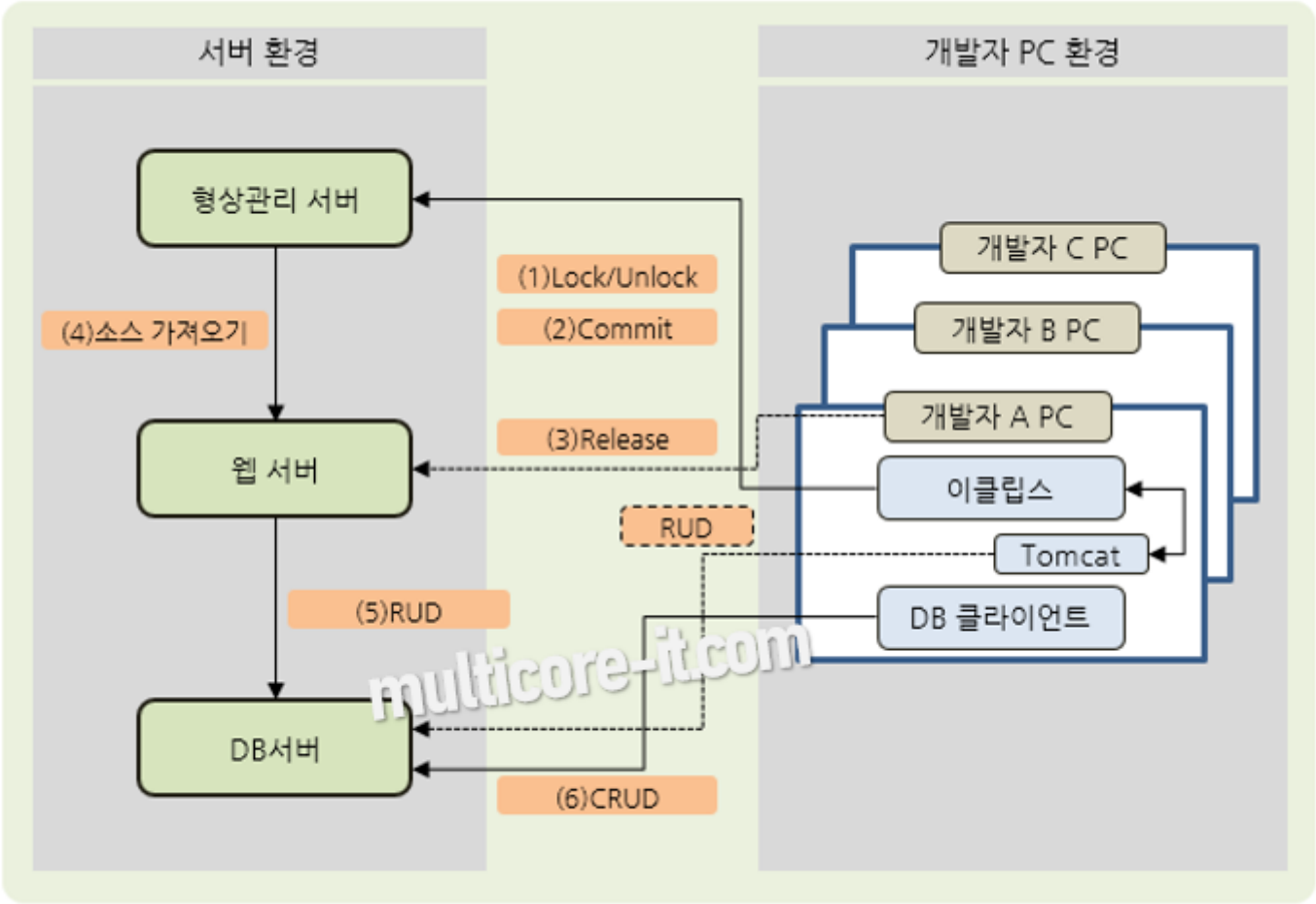
개발환경 구축하기

2018.11.19 15:22

#개발환경 #데이터베이스 #웹 서버 #형상관리

□ 개발환경 구축하기

○ 개발환경 구성



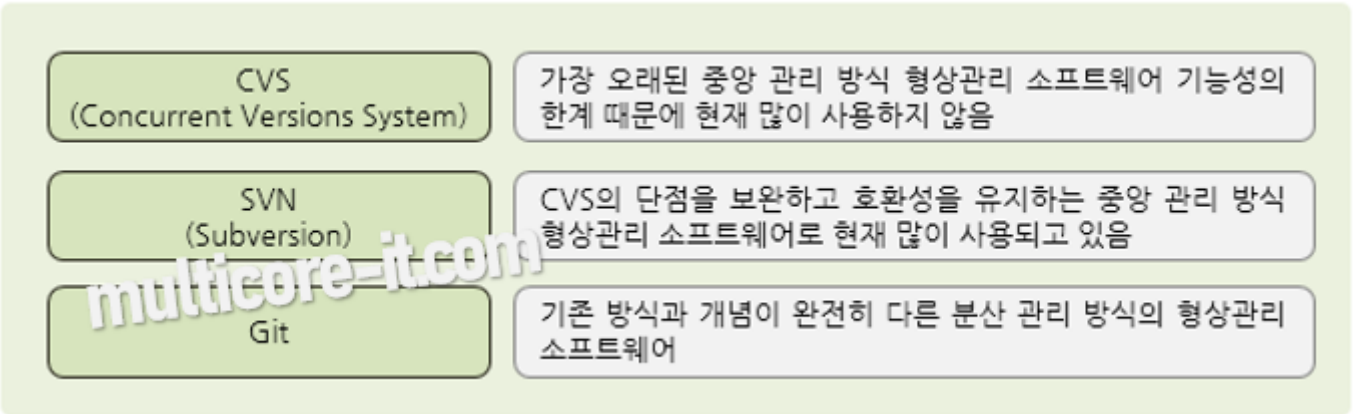
개발환경 구성

일반적인 프로젝트 개발 환경은 형상관리서버, 웹서버 그리고 DB 서버로 구성된다. 프로젝트 규모가 작은 경우에는 세 가지 시스템이 하나의 서버에 설치되기도 하지만 여력이 된다면 서로 기능과 역할이 다르기 때문에 별도의 서버로 구성하는 것이 좋다.

형상관리 서버는 여러 명의 프로그래머가 협업을 할 수 있도록 도와주고 프로그램과 산출물을 시계열적으로 관리하는 역할을 한다. 프로젝트에서는 하나의 프로그램을 한 명이 계속 개발하는 것이 아니라 다른 사람이 수정할 수 있기 때문에 프로그램에 대한 락(Lock)과 언락(Unlock) 기능을 필수적으로 지원해야 한다. (1) 공통 기능을 A 개발자가 수정하기 위해서는 먼저 프로그램에 대해 락을 걸로 수정해야 한다. 만일 프로그램에 대해 락이 걸렸다면 C와 D 개발자는 프로그램을 수정할 수 없다. (2) A 개발자가 프로그램 수정이 끝났다면 커밋(Commit)을 통해 내가 수정한 내역을 형상관리 서버에 저장해야 한다. (3) 커밋 후 언락이 되면 B와 C 개발자가 프로그램을 수정할 수 있다. A 개발자는 커밋을 하면서 자신이 어떤 내용을 수정했는지 텍스트로 기록해야 한다. 그래야 다른 개발자가 프로그램의 변경 내역을 확인하고 본인이 수정할 때 참고할 수 있다.

형상관리 서버의 또 다른 역할을 변경을 관리하는 것이다. 요구사항 명세서를 수정해서 형상관리 서버에 커밋하면서 무엇을 어떻게 고쳤는지 적어 놓으면 다른 사람이 요구사항 명세서를 참고할 때 변경 내역을 확인할 수 있다. 만일 수정되기 이전에 요구사항 명세서의 내용을 확인할 필요가 있다면 형상관리 서버에서는 커밋 날짜에 따른 요구사항 명세서를 모두 저장하고 있으므로 쉽게 찾아볼 수 있다. 특히 프로그램 같은 경우는 이전 버전과 현재 버전이 어떻게 차이가 나는지 프로그램에서 라인 단위로 비교해 주기 때문에 변경 내용을 아주 쉽게 파악할 수 있다.

○ 형상관리 소프트웨어

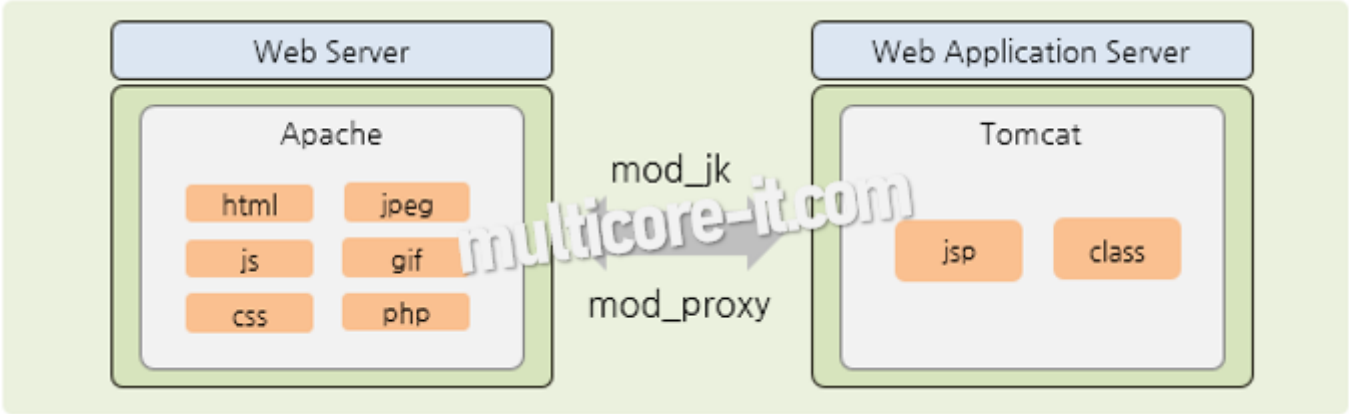


형상관리 소프트웨어

형상관리는 말 그대로 프로그램이나 문서의 형상을 기억하고 관리하는 것을 말한다. 형상을 기억한다는 것은 문서를 수정하기 전 버전과 수정 후 버전을 모두 저장하고 있다는 뜻이다. 즉 형상관리를 사용하면 언제든지 과거 버전으로 프로그램을 되돌릴 수 있다.

가장 먼저 만들어진 형상관리 소프트웨어는 CVS이다. 무료이고 쉽게 설치가 가능하지만 내장된 오류로 인해 SVN으로 빠르게 대체되었다. 최근에는 분산 관리 방식의 Git이 점차 확산되는 추세이다. SVN과 Git은 관리방식과 소프트웨어 동작 방식이 완전히 다르기 때문에 프로젝트에 참여한 개발자의 수전에 알맞은 형상관리 툴을 사용하는 것이 좋다. Git이 유행이라고 SVN에 익숙한 개발자에게 Git을 사용하라고 가용할 필요가 전혀 없다. 형상관리 소프트웨어를 사용하는 목적이 단순히 버전 관리를 위해서라면 SVN로 충분한 기능을 가지고 있기 때문이다.

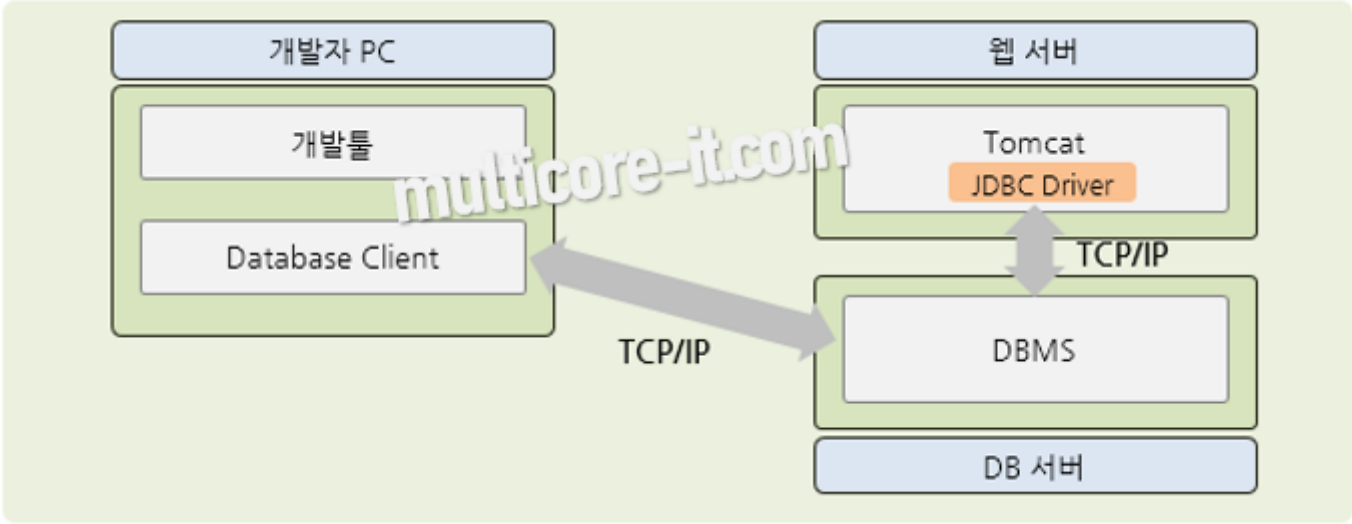
○ 웹 서버



웹 서버

웹 서버는 정적인 콘텐츠를 처리하는데 특성화 되어있다. 정적인 콘텐츠란 한번 만들어지면 변하지 않는 것을 말한다. 예를 들어 이미지, html, css, js 파일 등이 정적인 콘텐츠에 해당한다. 반면 웹 어플리케이션 서버는 동적인 콘텐츠를 주로 처리한다. jsp나 class 파일은 입력 값이 무엇이나에 따라 동작이 달라지기 때문에 웹서버 보다는 웹 어플리케이션 서버에서 처리한다. 우리가 가장 많이 사용하는 대표적인 웹 서버로는 아파치가 있고, 웹 어플리케이션 서버로는 톰캣이 있다. 웹 어플리케이션 서버는 다양한 상용 제품이 많이 나와 있지만 개발 환경에서는 톰캣을 활용해도 큰 문제가 없다. 자바 기반의 웹 어플리케이션 서버는 모두 j2ee 스펙을 지원하게 약속되어 있기 때문이다. 톰캣이 동적인 콘텐츠만 서비스할 수 있는 건 아니다 html과 이미지와 같은 정적인 콘텐츠 또한 처리가 가능하다. 하지만 웹서버에 비해 정적 콘텐츠 처리 성능이 떨어지기 때문에 아파치와 톰캣을 별도로 구성해서 사용하는 것이 좋다. 하지만 10억 이하의 중소규모 프로젝트에서는 개발서버에 톰캣만 설치해도 큰 문제가 되지 않는다.

○ 데이터베이스



데이터베이스

데이터베이스는 개발자 PC 환경에서는 전용 클라이언트를 통해 접근하고 웹서버 특히 톰캣에서는 JDBC Driver를 통해 접근한다. 데이터베이스는 운영 환경과 동일한 제품과 버전을 설치해야 한다. 웹서버와 달리 제품이 다르면 데이터베이스를 다루기 위한 SQL이 호환이 안되는 경우가 많고 제품이 같더라도 버전에 따라 세부 기능이 다르기 때문이다.

공감

구독하기

'더 팬찮은 개발자가 되기위한 > 시스템 개발하기' 카테고리의 다른 글	
대표적인 단위테스트 도구 JUnit (0)	2018.11.19
프로그램 단위 테스트 (0)	2018.11.19
효율적으로 코딩하기 (0)	2018.11.19
팀으로 개발하기 (0)	2018.11.19
개발환경 구축하기 (0)	

	2018.11.19
개발 표준 정하기 (0)	2018.11.19

NAME

PASSWORD

HOMEPAGE

http://

SECRET☐

WRITE

PREV

1

2

3

4

5

6

7

NEXT

+ Recent posts

9. 카트폴 DQN 모델 실행

8. 카트폴 DQN 강화학습

7. 텐서플로우 심층 신경망

6. 텐서플로우 단일 신경망