

소프트웨어공학



강의노트

프로젝트 형상관리

❖ 학습안내

이번 시간의 학습내용과 학습목표를 확인해보세요.

■ 학습내용

- 형상관리의 정의
- 형상관리의 실행
- 형상관리 관련 도구

■ 학습목표

- 프로젝트의 구성원으로서 형상관리를 성공적으로 진행할 수 있다.
- 형상관리를 위하여 형상항목식별,형상통제,형상심사 등을 수행할 수 있다.
- 형상관리, 배포관리, 변경관리를 이해하고 관련 자동화 tool을 활용할 수 있다.



❖ 학습내용

[1] 형상관리의 정의

1. 형상관리의 정의

- ◆ 형상관리
 - 프로젝트 과정에 변화되는 소프트웨어의 각종 산출물들을 체계적으로 관리
 - 소프트웨어에 가시성과 추적 가능성을 부여하여 소프트웨어 관리를 강화하고 품질보증을 도모하기 위한 관리기법
 - 프로젝트 진행 전반에 있어서 그때그때 달성될 목표에 대하여 변동 없이 달성하고자 하는 관리기법
- ◆ 형상관리의 중요성

형상항목 요소

IT 프로젝트를 진행하는데 있어서 구간별로 달성해야 할 목표와 기능수준

형상통제의 목적

부분의 변경을 위해서는 전체 과정의 영향을 고려하여 고민하고 수행하여야 함

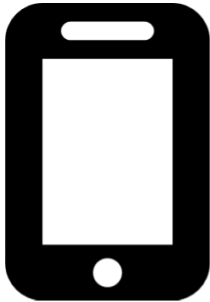
1. 형상통제의 실패 사례 - 휴대폰 개발

1



마케팅부서

전화번호가 100개까지 저장되게 해주세요.



전화번호목록을 50개까지 저장



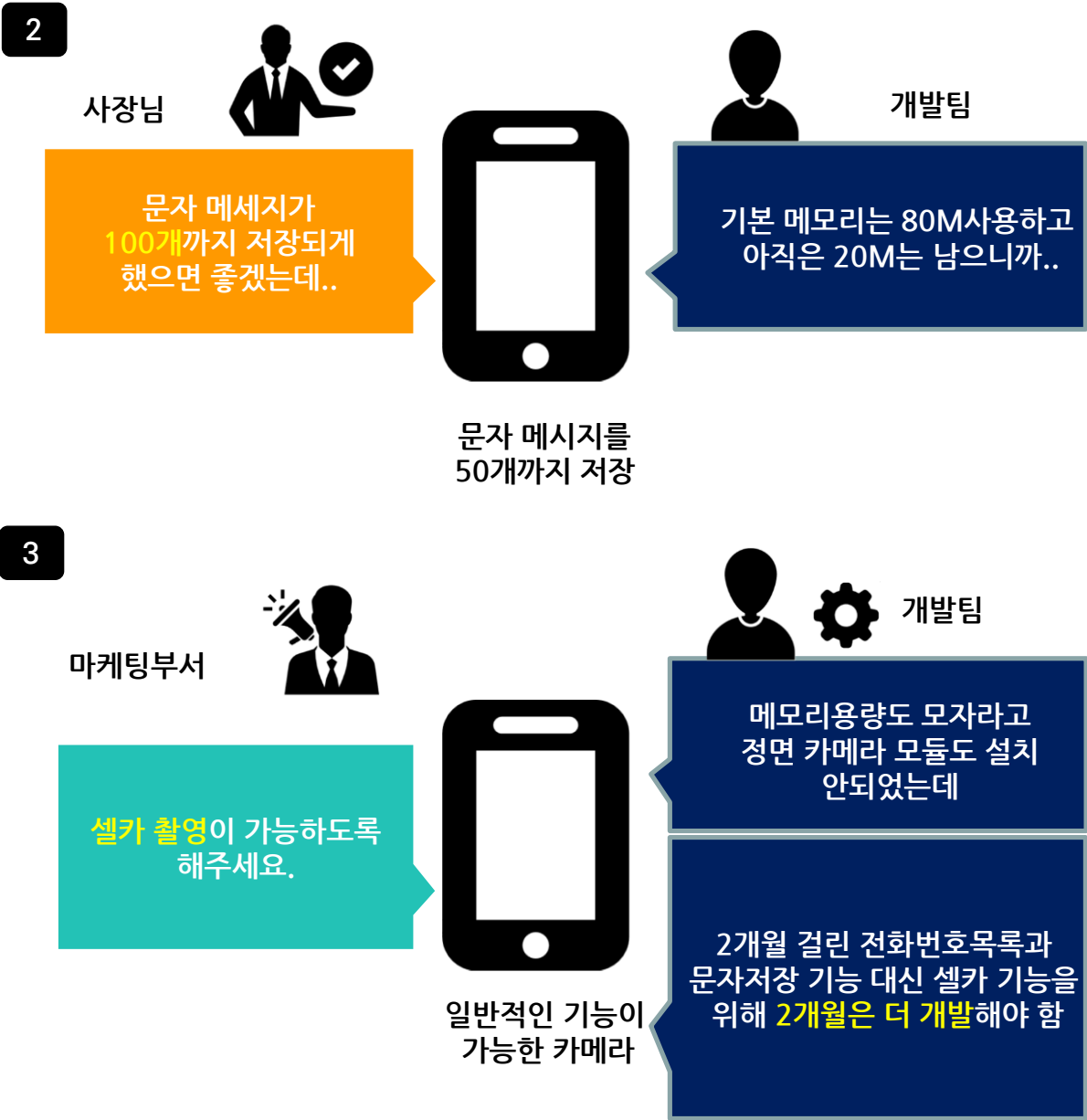
개발팀

휴대폰 메모리는 100M인데 100개의 주소를 저장해도 30M이니 충분하겠지?

❖ 학습내용

[1] 형상관리의 정의

1. 형상통제의 실패 사례 - 휴대폰 개발 (계속)



❖ 학습내용

[1] 형상관리의 정의

1. 형상관리의 정의(계속)

- ◆ 형상관리의 목적
 - 소프트웨어 변경으로 인한 사용자의 **충격 및 전체비용**의 최소화
 - 소프트웨어 공학 과정상의 **생산물 확인, 통계, 변경 상태 등 기록**
 - 생산성 극대화, 변경용이성 향상, 노력 감소
 - SDLC 전 주기에 걸쳐, SW 형상물의 **일관성 및 무결성을 체계적**으로 관리·통제하여 **생산성을 극대화**
 - 무형적 SW의 가시성, 체계적 통제, 추적성 확보
 - 무절제·비관리 변경 통제, Side-effect 최소화 및 품질 향상

2. 형상항목 및 기준점

- ◆ 형상관리 항목
 - 소프트웨어 개발 중 작성되는 **문서 및 산출물**은 형상관리의 대상
 - 프로젝트에서 공식적으로 식별, 관리되는 대상(환경, 문서, 산출물)

형상항목

시스템 명세서, 프로젝트 계획서, 설계 명세서, 원시코드 목록, 사용자 매뉴얼, 변경 요청서 등

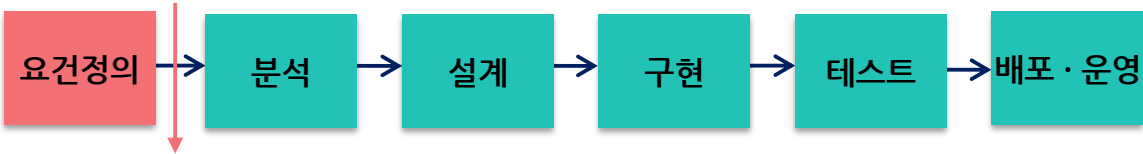
- ◆ 기준점(Baseline)
 - 프로젝트, 소프트웨어 개발 등 처리절차 진행 중 **형상 마감**이 이루어지는 기준점이 필요
 - 소프트웨어 개발 생명주기 모델에서 **각 단계 종료시점**은 하나의 **기준점**으로 판단하여 형상관리를 하여야 함
 - 기술적 통제 시점의 **중요결정 위원회, 공식 변경 절차**를 통해서만 변경
 - 역할: 다음 공정 기초, 변경에 대한 통제 기준, 변경 착오에 의한 이전 시점 회복 기준

❖ 학습내용

[1] 형상관리의 정의

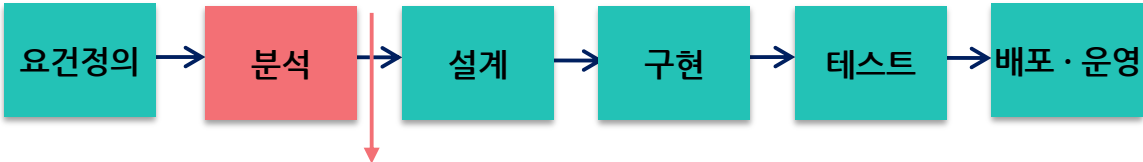
2. 형상항목 및 기준점(계속)

◆ 정보시스템 개발 프로젝트 단계별(SDLC) 기준점



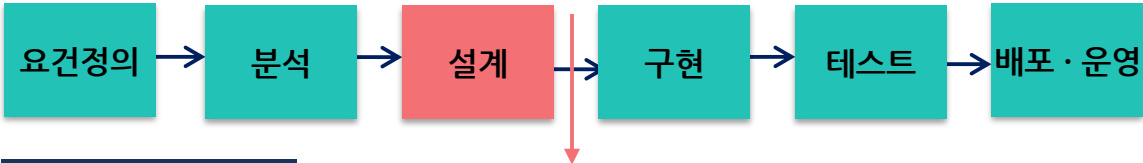
요건정의 단계 Baseline

- 요건정의 단계 산출물은 이후 단계에서 변경되는 것이 통제되어야 함
- 요구사항 정의서



분석 단계 Baseline

- 기존(AS-IS) 시스템 분석서
- 기존 프로세스 분석서 등



설계 단계 Baseline

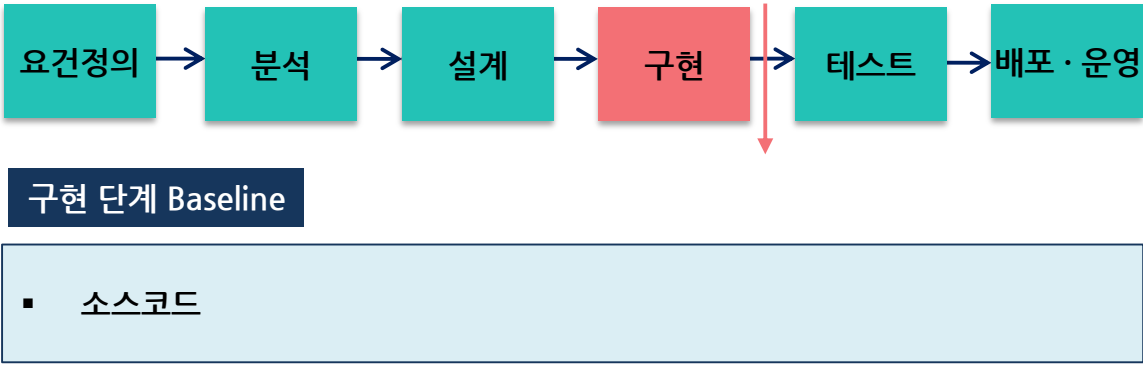
- 설계서

❖ 학습내용

[1] 형상관리의 정의

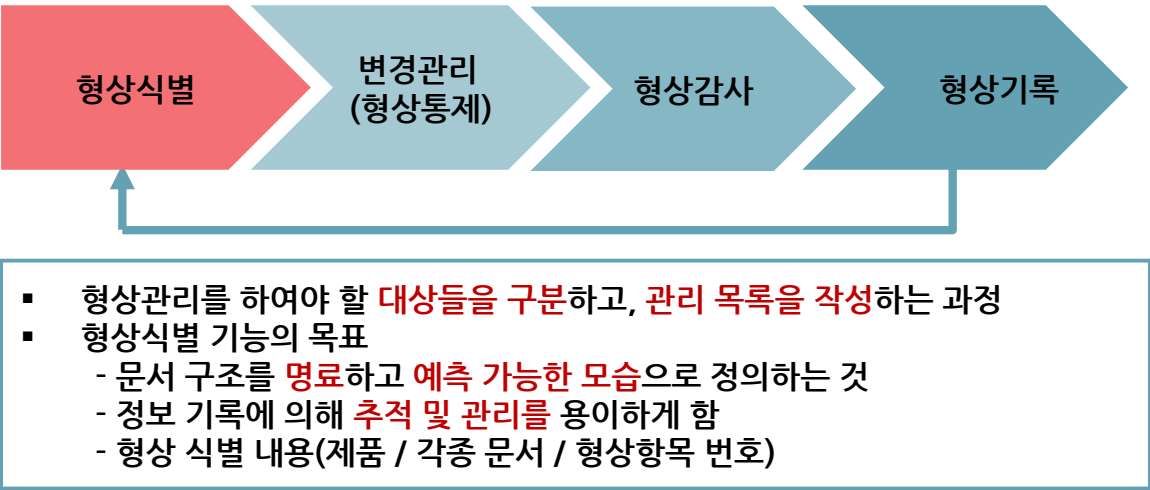
2. 형상항목 및 기준점(계속)

◆ 정보시스템 개발 프로젝트 단계별(SDLC) 기준점



3. 형상관리의 기능

◆ 형상관리 절차

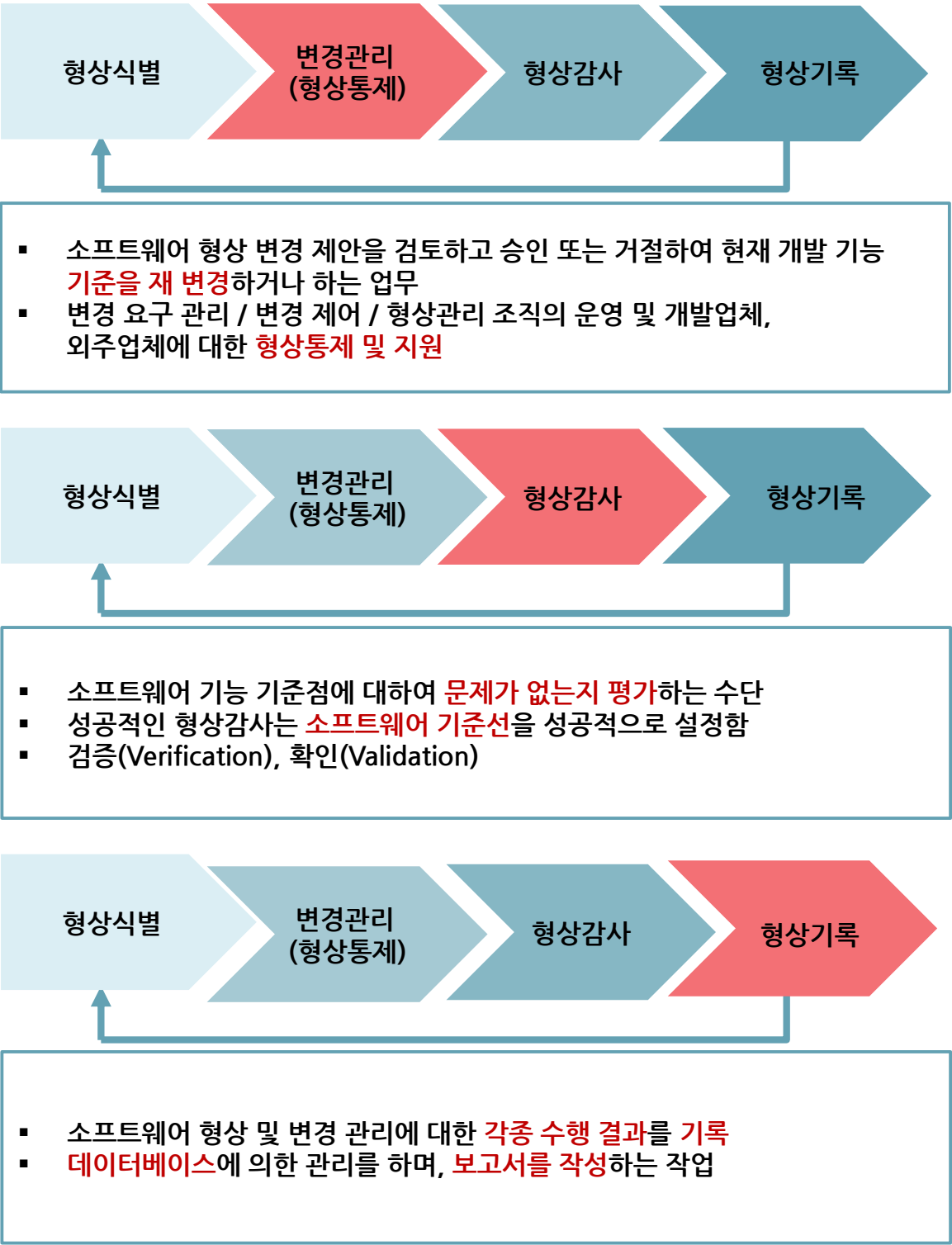


❖ 학습내용

[1] 형상관리의 정의

3. 형상관리의 기능(계속)

◆ 형상관리 절차(계속)

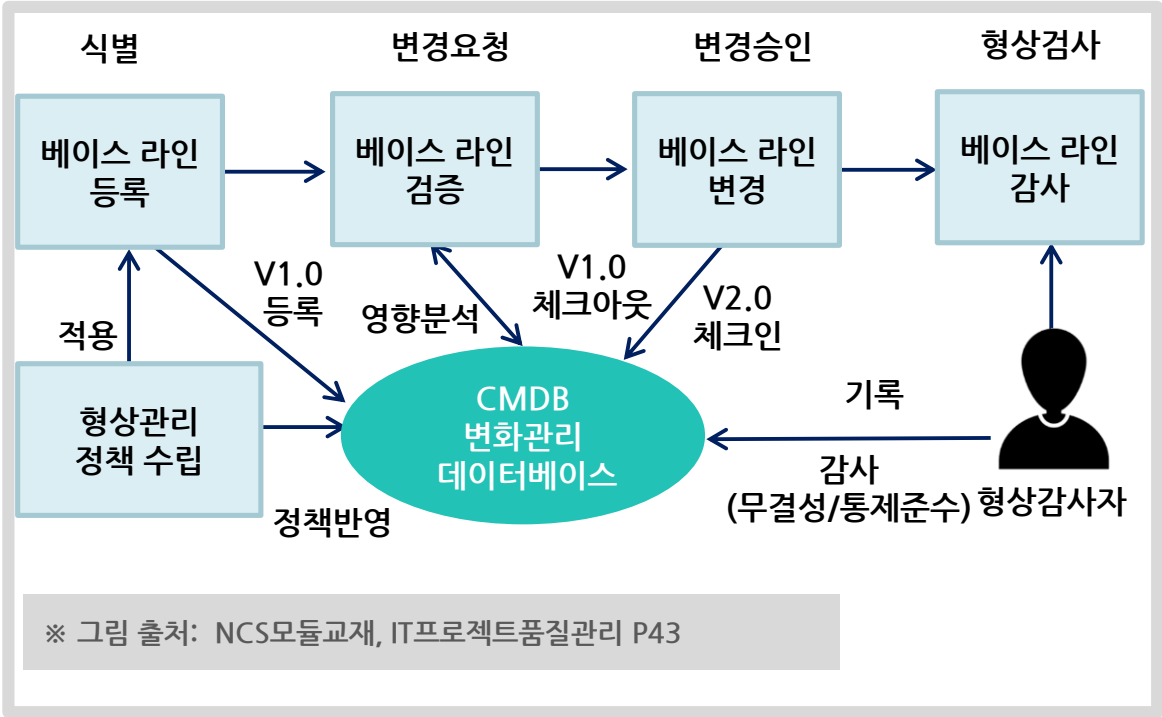


❖ 학습내용

[1] 형상관리의 정의

3. 형상관리의 기능(계속)

◆ 형상관리를 수행하는 프로세스



◆ 형상통제 위원회(CCB: Configuration Control Board)

- 소프트웨어 기준선에 대한 **설정 권한**과 이에 대한 **관리 기능**을 가지는 조직
- 역할

기준선의 **설정**

기준선에 대한 **변경사항** 관리

산출물의 형상관리 **변경 절차**

❖ 학습내용

[1] 형상관리의 정의

3. 형상관리의 기능(계속)

- ◆ 형상통제의 효과
 - 소프트웨어 라이프사이클(프로젝트 기간)동안 문서화, 코드, 인터페이스, 데이터베이스 등을 **관리(Control)**하고 **식별(Identify)**하는 구조를 제공
 - 요구사항, 표준, 정책, 조직 구조들을 만족하는 선택되어진 **개발 및 유지보수 방법론**을 지원
 - 베이스라인의 상태, 변경 관리, 테스트, 릴리즈, 감사 등에 관련된 **생산 정보와 관리(Management)**를 제공
 - 소프트웨어의 **품질과 신뢰성**을 높여줌

[2] 형상관리의 실행

1. 형상항목 식별 및 계획수립

- ◆ 전체 형상항목을 식별
 - 프로젝트 단계별 **기준선(Baseline)**을 통해 전체적인 **형상항목**을 식별

1. 형상항목 식별 및 계획수립 예시 - 프로젝트 단계별 기준선

종류	기준선	주요형상항목
계획 단계	기능적 기준선	<ul style="list-style-type: none">▪ 사용자의 요구사항 명세서 또는 시스템 기능 요구 정의서 검토▪ 프로젝트 계획서, 개발 표준 및 프로세스
요구분석 단계	분배적 기준선	<ul style="list-style-type: none">▪ 사용자의 요구 기능이 하위시스템 사이에 어떻게 분배되는가를 정의하는 기본 설계 명세서를 검토하는 시점▪ 요구사항 정의서, 기능분해도, 작업 흐름도, 자료 흐름도
설계 단계	설계 기준선	<ul style="list-style-type: none">▪ 프로그래밍에 임하기 위한 설계 명세서를 검토하는 시점▪ 기본 설계: 화면·보고서 명세서▪ 상세 설계: ERD, 아키텍처 설계서,▪ 프로그램 설계서

❖ 학습내용

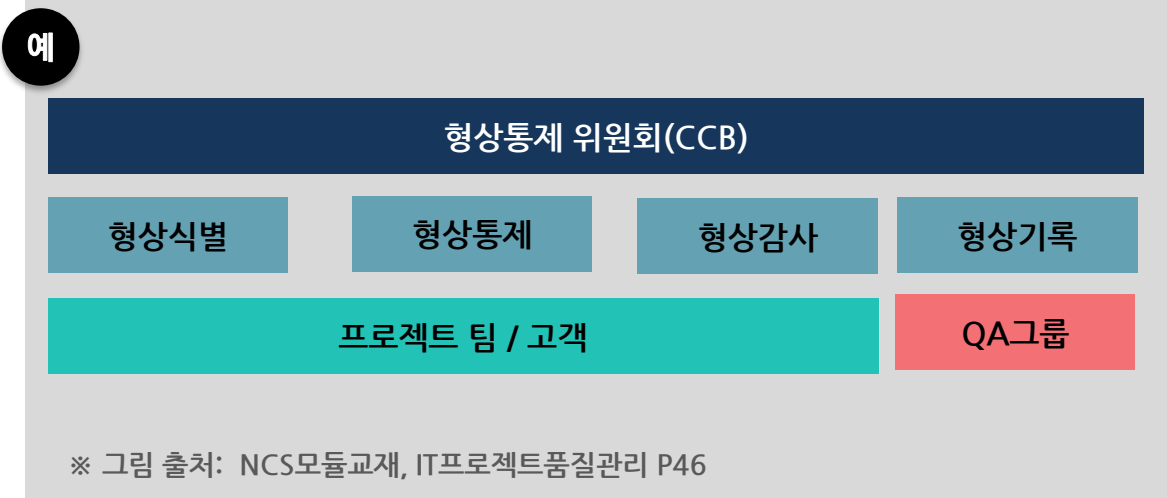
[2] 형상관리의 실행

1. 형상항목 식별 및 계획수립 예시 - 프로젝트 단계별 기준선(계속)

종류	기준선	주요형상항목
구현 단계	시험 기준선	<ul style="list-style-type: none">소프트웨어의 기능 수행과 성능 충족도를 평가할 수 있는 원시 코드, 실행 코드 및 시험 사례를 포함한 시험 계획서를 검토하는 시점원시 코드, 목적 코드, 실행코드, 단위 시험 보고서
시스템 통합 및 시험 단계	제품 기준선	<ul style="list-style-type: none">하나의 시스템으로 개발 완료된 제품과 그의 품질을 보증하는 시점통합 시험: 통합 시험 계획서, 케이스시스템 시험: 케이스 및 보고서
설치 및 운영 단계	운용 기준선	<ul style="list-style-type: none">사용자 환경에 설치되고 운용되기 시작한 소프트웨어나 품질을 사용자 입장에서 평가하는 시점운영자 지침서, 사용자 지침서, 이관 소스

1. 형상항목 식별 및 계획수립(계속)

- ◆ 형상관리 조직 구성 및 역할 정의
 - 형상관리 조직 구성



❖ 학습내용

[2] 형상관리의 실행

1. 형상항목 식별 및 계획수립(계속)

- ◆ 형상관리 조직 구성 및 역할 정의(계속)
- ◆ 형상관리 조직의 역할

예

변경 요청서, 서비스 요청서, 결함 보고서 등을 받아 공식·비공식 회의를 거쳐 다음 **사항의 승인 여부**를 결정

- 형상항목에 대한 기술적 변경
- 성능의 변화가 허용 한도를 초과하는 변경
- 인터페이스에 영향을 주는 변경
- 안전과 호환성 등에 영향을 주는 변경
- 운영 환경에 영향을 주는 변경

2. 형상통제 실시

- 1

변경 관리 및 배포 관리와 연동하여 효율적인 형상관리를 수행
- 2

소프트웨어 변경 관리에 대한 별도 절차를 수립

❖ 학습내용

[2] 형상관리의 실행

2. 형상통제 실시 예시 - 변경 관리 절차

Activity	설 명
변경 요청	개발 부서나 현업 사용 부서 서비스 관리 부서에서 변경 사항 이 발생하면 해당자의 확인을 받은 변경 요청서를 통해 변경을 요청
변경 결정	변경 관리자는 변경 요청서를 검토하여 변경의 적합성을 여과, 필요하지 않을 때는 변경 요청자에게 회송, 적합하다고 판단 시 변경 모델 결정
단순 변경	변경 관리자가 변경 내용의 영향도가 적다고 판단될 시 단순 변경 가능
CAB/CCB 개최	변경의 영향도가 클 경우, 변경의 긴급성을 판단하기 위해 CAB 개최 CAB의 결정에 따라 긴급 변경이나 일반 변경을 결정함
긴급 변경	긴급 시험 실시, 성공 시 변경 요청, 긴급 변경 수행 변경 후 사후 보고 및 사후 형상관리 감사 필요
일반 변경	변경 수행 및 철회 계획 수립, 변경사항 사전 시험 변경 조정 및 변경 수행

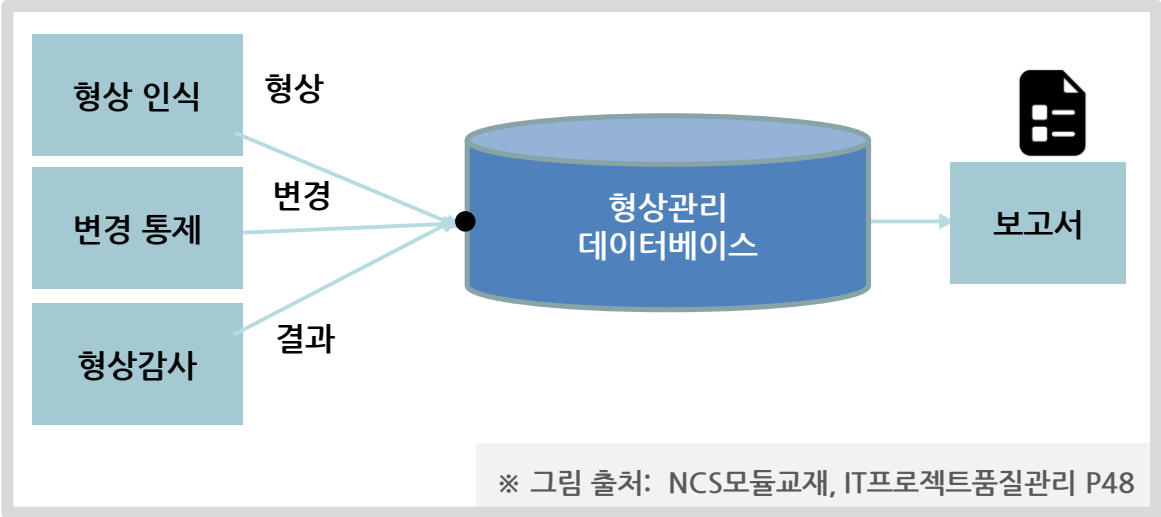
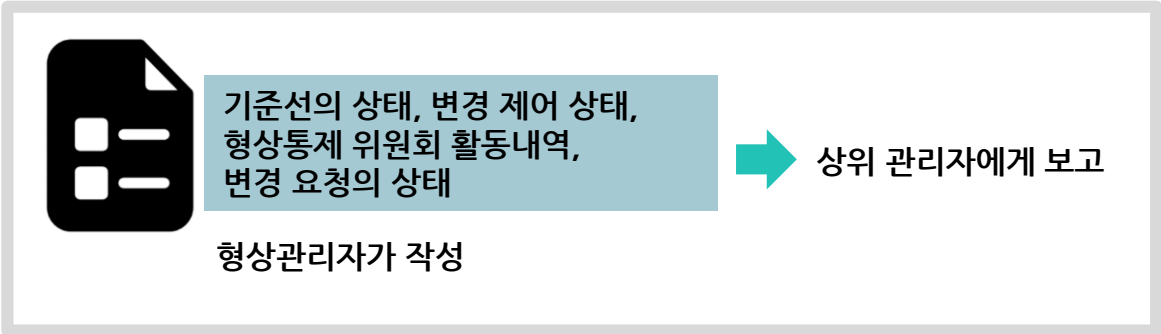
- CCB(Change Control Board): 변경관리 위원회(형상통제 위원회), PMBOK에서 사용하는 용어
- CAB(Change Advisory Board): 변경자문위원회, ITIL에서 사용하는 용어

❖ 학습내용

[2] 형상관리의 실행

3. 형상관리 기록 유지 및 감사

- ◆ 형상관리 기록 유지 및 배포
 - 형상 상태 보고(Configuration Status Accounting)
 - 기준선의 **현재 상태 및 변경 항목**이 제대로 반영되었는지 여부를 보고
 - 기준으로 설정된 형상항목의 구조와 변경 상태를 기록하여 보고함으로써 **형상항목의 개발 상태**에 대한 **가시성 제공**이 가능
 - 형상 상태 보고서 제출
 - 형상관리 계획서에서 정한 **주기**대로 작성 및 제출



❖ 학습내용

[2] 형상관리의 실행

3. 형상관리 기록 유지 및 감사(계속)

◆ 형상감사 실시

- 형상감사(Configuration Audit)를 위한 **사전 검토 활동**을 수행



형상항목이 **요구사항**에 맞도록 잘 변경되었는지 확인



형상관리 계획서대로 형상관리가 진행되고 있는지,
요구사항 문서대로 제품이 제작되었는지를 확인



프로젝트 기준선의 **무결성**을 검증


- 형상감사 활동을 수행
- 형상관리 계획서를 기반으로 **형상관리자**가 실시
 - 모든 승인된 변경의 반영 여부
 - 관련된 항목의 갱신 여부
 - 승인되지 않은 변경의 반영 여부
 - 인도될 항목과 요구사항, 디자인과의 일치 여부
 - 모든 알려진 소프트웨어의 문제가 변경 승인에 의해 검토되었는지 여부

❖ 학습내용

[3] 형상관리 관련 도구

1. 형상관리 관련 개념

- ◆ 정보시스템 개발 프로젝트에서 형상관리
 - 프로젝트에서 형상관리는 **변화관리의 측면**이 강함
 - 요건정의, 분석, 설계, 구현, 테스트의 각 단계 종료시점에서 산출물의 검토가 중요한 요소
 - ↳ Baseline
 - ↓ 문서, 프로그램소스, 데이터자료 등

요건정의 단계	사용자 또는 개발요청자의 요청사항이 최종 정의 되었는지 관리
분석, 설계, 개발 단계	해당 진행되어야 할 범위가 변동되지 않도록 하여 단계별 종료 후 다음 단계의 진행 이 이루어질 수 있도록 관리하는 것이 중요사항
구현 단계	상당기간의 많은 개발 인력이 투입되어 개발되어야 하는 경우가 많아서 협업관리를 지원 하는 도구(Tool)를 많이 사용함
 <div>PMS(Project Management System) 프로젝트 관리 도구관리 시스템(Tool)을 사용하여 전체 프로그램 관리 표준화</div>	

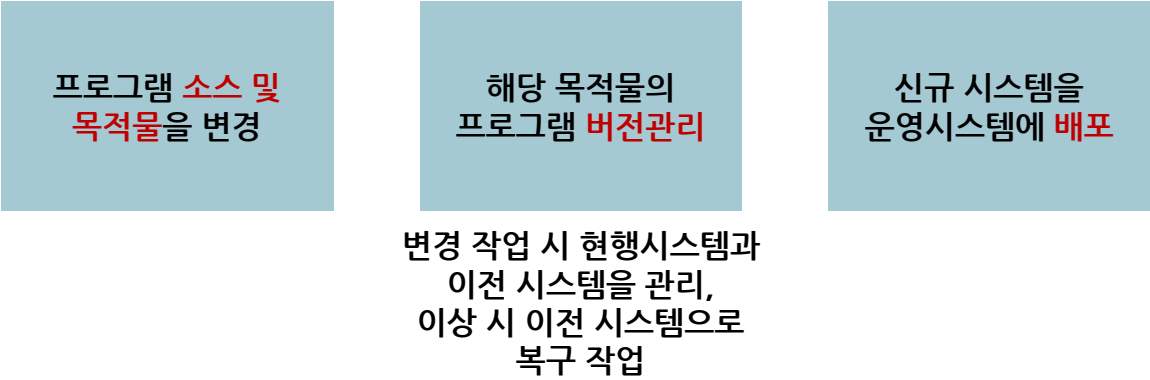
- 정보시스템의 운영 단계에서도 사용자의 요구의 변경, 제도, 시스템 성능향상의 필요성 등 다양한 **요청**에 의한 **시스템의 변경, 수정, 개선작업**이 수시로 이루어짐
- 정보시스템을 변경을 분석
- 다른 시스템의 영향을 고려하고 기존 시스템의 **운영에 연속성을 보장**하는 상태에서 시스템의 **변경, 수정, 개선작업**을 진행
- 리스크 없이 **시스템 이관 가능**한 형상관리가 필요함

❖ 학습내용

[3] 형상관리 관련 도구

1. 형상관리 관련 개념(계속)

- ◆ 정보시스템 개발 프로젝트에서 형상관리(계속)
 - ITSM(IT Service Management)시스템
 - ITSM시스템 내부의 서브 시스템



- ◆ 형상관리 시스템
 - 단순히 프로그램변경, 버전관리, 배포관리만이 형상관리가 아니라 PMO (Project Management Office)차원의 관리활동을 의미하는 큰 개념임
 - 버전관리 시스템, 배포관리 시스템은 개발 목적물을 최신 상태로 유지하는 버전의 관리나 운영시스템으로 배포하는 관리를 담당하는 시스템임
 - 버전관리 시스템과 배포관리 시스템은 형상관리 시스템의 작은 부분 중 하나임 (시중에서 Git, SVN을 형상관리 시스템이라고 부르는 것은 잘못된 사항)
 - 여러 사람이 동시에 하나의 프로그램 체계를 가지고 개발 시 프로그램 개발 협업관리 시스템이 많이 사용됨

예

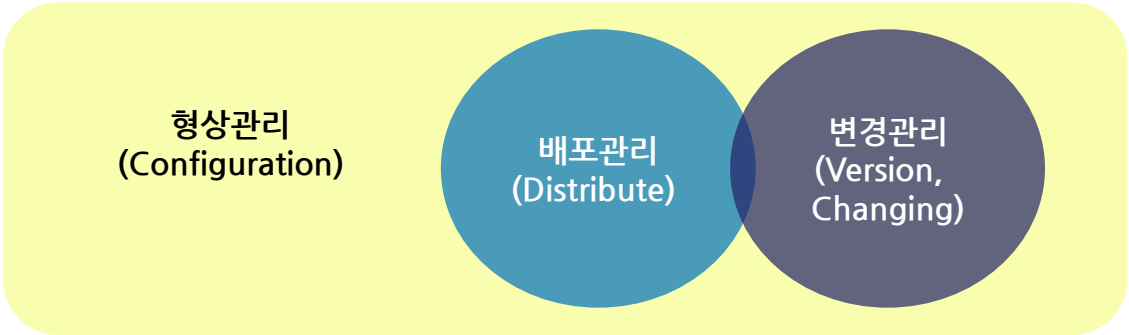
- SVN: 소프트웨어의 버전과 수정을 관리하는데 유용한 프로그램으로 많이 사용됨
- Git: 여러 곳에 분산되어 있는 소프트웨어의 버전을 관리하는데 유용한 프로그램으로 무료 오픈 소스로 많이 사용됨
- Developer Team Suite: 마이크로소프트의 협업관리를 소프트웨어로 유료(상용) 프로그램

❖ 학습내용

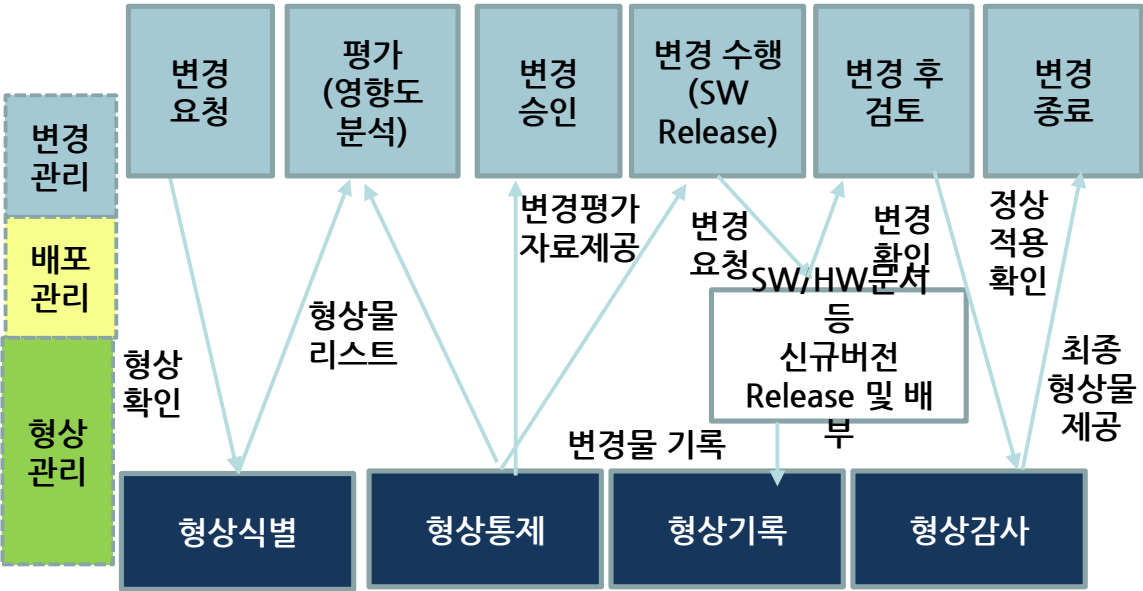
[3] 형상관리 관련 도구

1. 형상관리 관련 개념(계속)

◆ 형상관리 시스템(계속)



- 배포관리, 변경관리의 활동을 포함한 개념
- 형상관리는 품질관리와 밀접한 관계를 가지고 있음



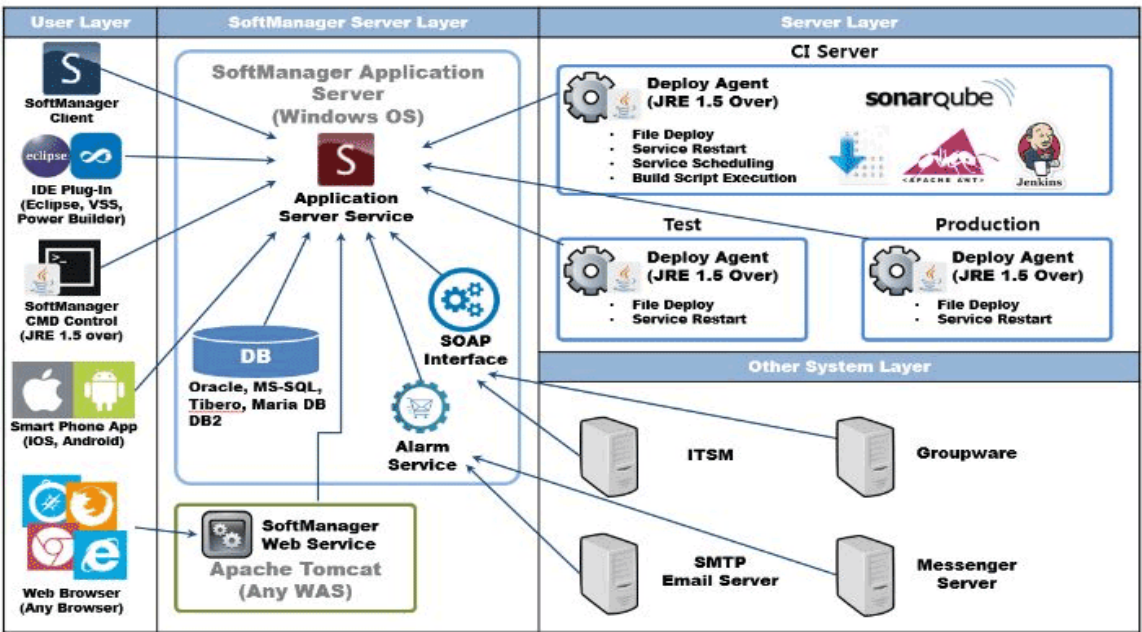
❖ 학습내용

[3] 형상관리 관련 도구

2. 형상관리 시스템

- ◆ 상용 형상관리 시스템
 - 형상관리 시스템은 **ALM(Application Lifecycle Management)**라고도 함
 - 형상관리 솔루션의 기능에서는 형상관리에서 관리할 주요 포인트 또는 사람의 정성적인 판단보다는 **자동화하여 관리**하여야 하는 요소들을 모아서 지원하고 있음
 - 실제 시중에서 판매되는 **형상관리 솔루션을 교육목적으로 소개**함

2. 형상관리 시스템 예시 - Wbjsoft SoftManager



※ 그림 출처: <http://wbjsoft.com/>

- 버그트래킹, 소스 및 문서 버전관리, 배포관리 등의 **통합 형상관리 솔루션**임
- 개발용 도구, 기존 업무시스템(Groupware, ITSM)등과 **연동**되어 있음

❖ 학습내용

[3] 형상관리 관련 도구

2. 형상관리 시스템 예시 - Wbjsoft SoftManager (계속)

◆ 특징

기능	설명
형상관리 기능	<ul style="list-style-type: none">모든 형상 자원(4G 이상 파일도 지원)에 대한 변경 이력관리 기능소스, 이미지, 산출물 및 다양한 포맷의 형상자원 버전 관리승인 절차에 따라 자동으로 버전 태깅되어 승인된 소스만 공유 작업 관리 제공특정 소스 라인 추적 검색 기능 제공: 누가 소스라인을 어떠한 이유 때문에 수정했는지 추적성 제공다양한 개발툴 연동 지원(Eclipse, Visual Studio, PowerBuilder, VB 6.0, ProframeStudio 등)
이슈관리 기능	<ul style="list-style-type: none">업무 흐름도 관리를 통한 직관적이고 편리한 관리 기능 제공변경요청서와 소스라인 수준의 양방향 추적성 제공모바일앱에서 승인 / 결재 등 다양한 업무 처리 기능 제공효율적인 업무 분장을 위한 업무 일정관리 기능 제공
배포관리 기능	<ul style="list-style-type: none">다양한 OS 및 배포 환경에 배포 자동화 기능 제공스케줄 배포 및 비인가 접근에 대한 모니터링 등 운영관점에 사고 방지를 위한 기능 제공배포 수행 내역 및 결과 실시간 모니터링 및 자동 대응 방안 설정 기능 제공스크립트 수행으로 WAS/WEB 등 On/Off 및 빌드 쉘 수행으로 빌드관리 지원매시간 / 매일 / 매주 등 다양한 빌드 스케줄링 수행 기능 제공

❖ 학습내용

[3] 형상관리 관련 도구

2. 형상관리 시스템 예시 - Wbjsoft SoftManager (계속)

◆ 특징(계속)

기능	설명
기타 기능	<ul style="list-style-type: none">통합된 관리툴에서만 제공될 수 있는 다양한 통계 및 통계 디자이너 제공개발 업무와 밀접한 테스트관리, 통제관리(RDP/SSH/DB서버), 성능관리 제공업무 흐름도 관리 기능을 제공하여 다양한 변경 업무 절차 구성문서폼 관리 기능으로 변경 업무에 필요한 전자폼 Template 화최신의 개발 방법론과 업무 절차 등 라이브러리화 해서 지속적으로 제공

◆ 주요 기능 분석

- 저장소에 **이전, 현재 소스 및 수정자 정보** 등을 기록하여 관리

형상자원의 최신 정보 및 상태 조회

형상자원 구조 그대로 구성 되는 저장소(Repository)

※ 그림 출처: <http://wbjsoft.com/>

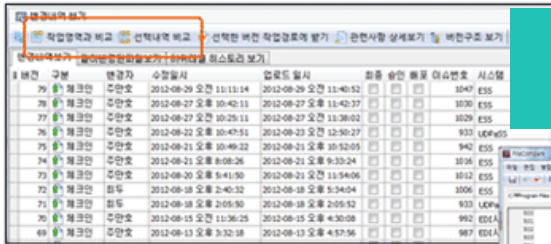
❖ 학습내용

[3] 형상관리 관련 도구

2. 형상관리 시스템 예시 - Wbjsoft SoftManager (계속)

◆ 주요 기능 분석(계속)

- 개발자가 사용하는 **개발 툴**(예: Eclipse, Visual Studio 등)과 **연동**, 소스 변동부분 등을 보여줌



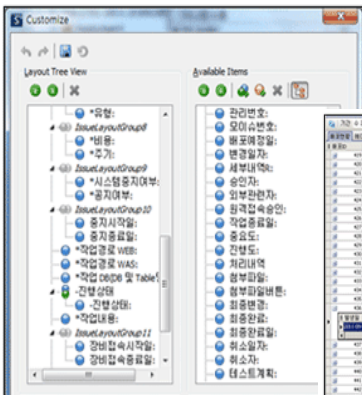
이름	구분	단위명	수정일시	입력일시	활동	상태	이슈번호	시스템
79	개발자	주안호	2012-08-29 오전 11:11:14	2012-08-29 오전 11:40:52			1047	E55
76	개발자	주안호	2012-08-27 오후 10:42:11	2012-08-27 오후 11:42:37			1030	E55
77	개발자	주안호	2012-08-27 오후 10:25:11	2012-08-27 오후 11:30:02			1029	E55
76	개발자	주안호	2012-08-25 오후 10:47:51	2012-08-25 오전 12:30:27			933	E4955
75	개발자	주안호	2012-08-21 오후 10:40:22	2012-08-21 오후 10:52:05			942	E55
74	개발자	주안호	2012-08-21 오후 8:08:26	2012-08-21 오후 8:33:24			1036	E55
72	개발자	주안호	2012-08-20 오후 4:41:30	2012-08-21 오전 11:54:06			1012	E55
71	개발자	최두	2012-08-18 오후 2:40:32	2012-08-18 오후 5:34:04			1006	E55
71	개발자	최두	2012-08-18 오후 2:05:50	2012-08-18 오후 2:05:52			933	UD55
70	개발자	주안호	2012-08-15 오전 11:36:25	2012-08-15 오후 4:30:08			992	ED55
69	개발자	주안호	2012-08-13 오후 3:32:18	2012-08-13 오후 4:57:36			987	ED55

원하는 버전 간 혹은 로컬과 특정 버전의 비교 수행

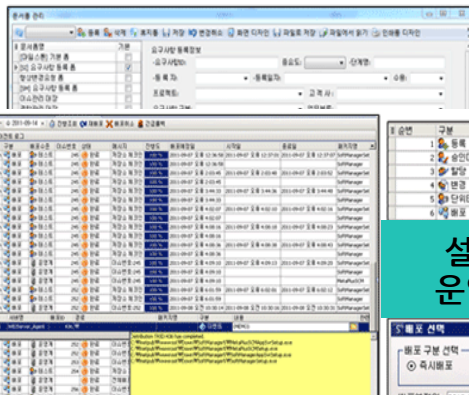
비교 툴을 통하여 서로 상이한 부분을 표시

※ 그림 출처: <http://wbjsoft.com/>

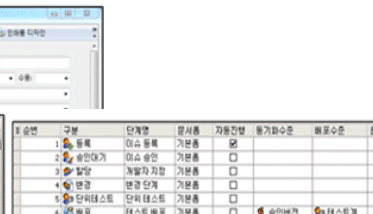
- 완료된 구현물을 실 운영 서버 등 **운영시스템으로 이관 및 배포**해 주고, 배포상황을 **모니터링**



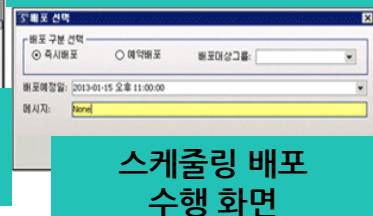
다양한 필드 제공



다양한 원격 서버에 수행되는 배포



설정을 통한 테스트 / 운영 배포 단계적 배포



스케줄링 배포 수행 화면

※ 그림 출처: <http://wbjsoft.com/>

❖ 학습내용

[3] 형상관리 관련 도구

3. 버전관리SW와 형상관리 시스템

◆ 버전관리(팀 개발) 도구의 필요성

소스의 버전관리

실수로 소스를 삭제하거나, 아니면 수정하기 전으로 돌아가야 되는 경우가 발생 시 원복 필요

잠금 기능 또는 파일명을 변경하는 기능

프로젝트나 유지보수를 하다 보면, 단순히 버전관리 뿐만 아니라, 다른 사람들이 임의로 변경하는 것을 막음

팀 공동 개발

한 소스를 가지고 새로운 기능을 추가하는 팀과 이전의 버그를 수정하는 팀이 동시에 있을 수 있으며, 이 두 작업이 끝나면 다시 병합해야 하는 경우도 있음

유용한 오픈 소프트웨어: Git, SVN, CVS등이 있음
Git를 가장 많이 사용함

Comparison of Version Control Software은 위의 오픈 소프트웨어의 성격을 정확히 표현한 용어

❖ 학습내용

[3] 형상관리 관련 도구

3. 버전관리SW와 형상관리 시스템

- ◆ Concurrent Versions System(CVS)
 - GNU license로 공개되었으며 사용자가 **작업할 코드를 “check out”**하고, **변경된 코드를 “check in”** 하는 시스템
 - 처음에는 오로지 최종 버전의 코드만 작업이 되고 수정을 허용함으로써 두 프로그래머간의 충돌을 방지
 - **먼저 온 것이 먼저 처리**되는 시스템으로 사용자는 변경을 빨리 적용, 다른 사용자들 이 변경 적용으로 선수치는 것을 막아야 했음
 - CVS 서버는 **유닉스 종류 시스템**에서 구동이 되고, Client Software 다양한 운영체제에서 구동이 됨
 - 가장 **안정된 버전 제어 시스템**으로써, 오랫동안 사용되고 있음

장점

단점

- **오랫동안 사용**이 되어 왔고, **안정된 것**으로 고려 됨

장점

단점

- 파일 이동이나 이름변경은 **버전 변경**이 미발생
- 파일 **Symbolic 링크**에 대한 보안 문제
- 소스 충돌에 대한 세부 운영에 대한 **지원 및 안내 미지원**
- **버전 분기**가 힘들고, 장기간 분기된 버전 운영에 대해서 설계되지 않았음

❖ 학습내용

[3] 형상관리 관련 도구

3. 버전관리SW와 형상관리 시스템(계속)

- ◆ Apache Subversion(SVN)
 - SVN은 CVS와 높은 호환성을 유지, 약간의 버그를 수정하면서 대체 시스템으로 개발

기능	SVN	CVS
Database 오류 발생	소스에 모든 변경이 적용이 되거나 아무것도 적용이 안 되는 경우가 발생	원본 소스를 변경하는 문제점을 해결
분기 사용	대규모의 분기된 프로젝트에서 유용하게 사용	CVS의 분기 버전이 사용이 어렵고, 분기 버전 자체는 프로젝트에서 장기 운영에 유용하지 않음

장점

- CVS를 기초한 좀 더 새로운 시스템으로 세부적인 조작이 가능
- 효율적인 분기 버전 운영
- SVN를 지원하는 많은 IDE용 플러그인 툴, Peer-to-peer 모델을 사용하지 않음

장점

- 파일과 디렉토리 변경에 관련된 버그
- 불충분한 저장소 관리 명령어
- 더 느려진 상대적인 속도

단점

단점

❖ 학습내용

[3] 형상관리 관련 도구

3. 버전관리SW와 형상관리 시스템(계속)

◆ Git

- CVS에 관행과 관례를 공개적으로 개선으로 보다 빠른 분산된 **버전 제어 시스템**을 지원
- 주로 **리눅스용**으로 개발이 되고 있고, 리눅스 상에서 **가장 빠른 속도**를 보임
- **유닉스 종류**도 지원하며, Windows용 Git 설치프로그램인 msysgit도 있음



- 중앙 집중화된 서버가 없기 때문에, **개발자 개인 프로젝트나 작은 팀에게도** 유용하게 사용
- 저장소가 없는 컴퓨터를 사용할 때는 코드가 항상 이용 가능하지 않을 수 있기 때문에 적절한 방식으로 **버전정보를 계속 교환**함
- 사용자 이력 시스템을 탐색할 수 있는 다양한 툴 제공, 각각의 소스본은 이력 전체를 포함하여 **인터넷 연결이 없을 때** 유용함

❖ 학습내용

[3] 형상관리 관련 도구

3. 버전관리SW와 형상관리 시스템(계속)

◆ Git(계속)

장점

단점

- CVS/SVN를 싫어하는 사람들에게도 유용함
- 조작 **속도**의 큰 개선, 분기 버전의 **효율적인 운영**
- **오프라인**에서도 전체 이력이 이용 가능함
- 분산된 Peer-to-peer Model

장점

단점

- SVN 사용자들이 배우기 어려움
- 개별 개발자들에게는 적절치 않음
- 리눅스에 비해 제한된 Windows 지원

❖ 핵심정리

1. 형상관리의 정의

- 프로젝트 과정에 변화되어 가는 소프트웨어의 각종 **산출물**들을 체계적으로 관리하고, 소프트웨어에 **가시성과 추적가능성**을 부여하여 소프트웨어 관리를 강화하고 품질보증을 도모하기 위한 관리기법
- 형상관리는 **형상식별, 형상통제, 형상감사, 형상기록의 절차**로 진행됨

2. 형상관리의 실행

- 프로젝트 **단계별 기준선(Baseline)**을 통해 전체적인 형상항목을 식별함
- **형상통제**를 실시하며, **형상관리 기록 유지 및 감사**를 진행

3. 형상관리 관련 도구

- 형상관리는 **배포관리, 변경관리의 활동**을 포함한 개념
- 형상관리를 솔루션 도구를 이용하여 자동화 하는 시스템을 **형상관리 시스템 또는 ALM**이라고 함
- 개발 프로그램의 버전을 관리해주는 오픈 소스로 **Git, SVN**이 많이 사용됨