To-Be 개발 방법론 Overview

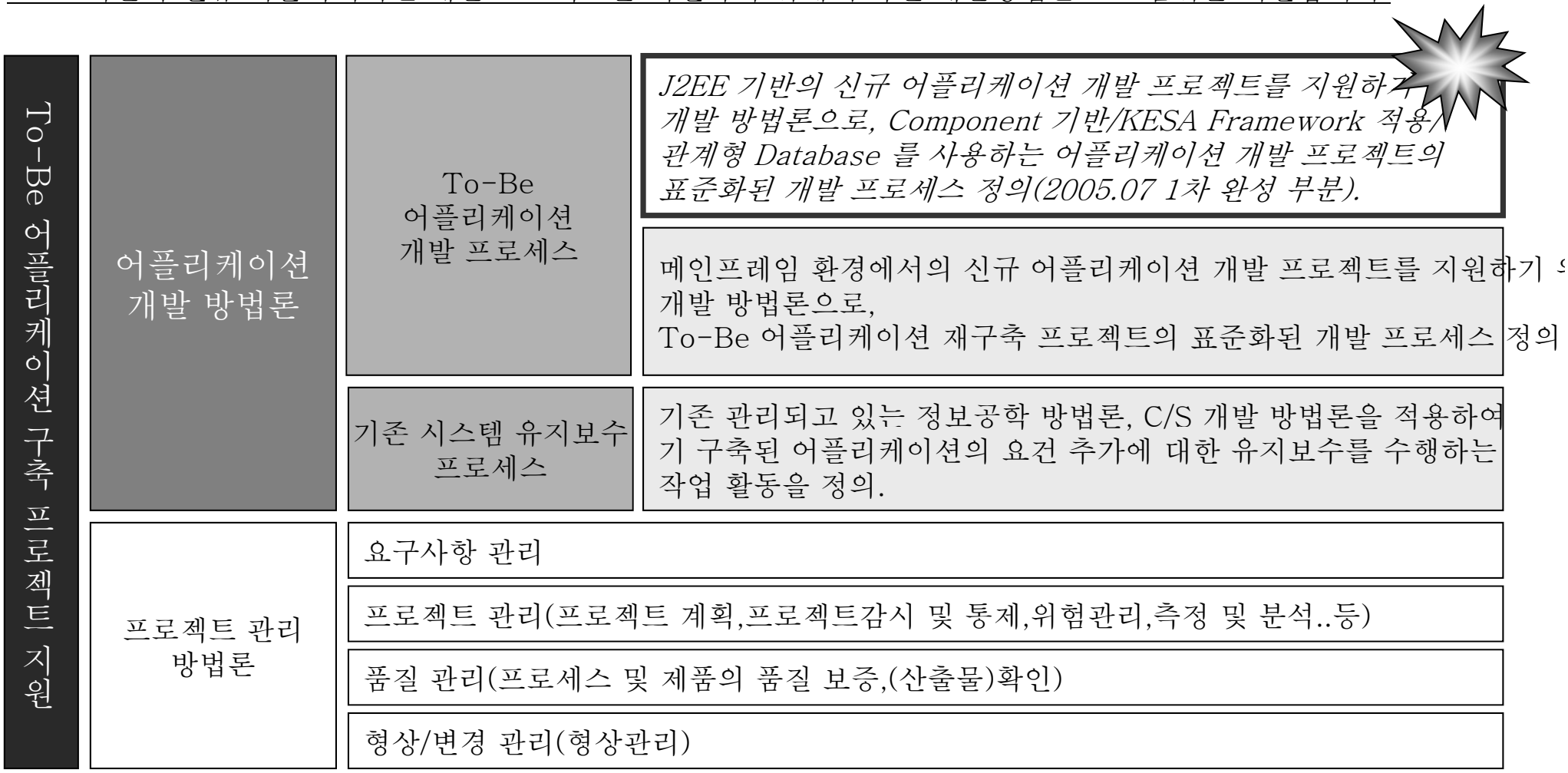
목 차

I. 개요
1. To-Be 개발 방법론 1차 구축의 범위
2. 소프트웨어 개발 프로세스의 정의
3. 개발 방법론과 관리 방법론의 관계 정의
4. 개발 방법론 구성요소

II. To-Be 개발 방법론 구성
1. To-Be 개발 방법론 절차 구성
2. To-Be 개발 방법론 절차 정의 기준
3. To-Be 개발 방법론 단계(Phase)별 활동 및 산출물 정의
4. Layer별 개발 절차 시나리오
5. To-Be 개발 방법론 적용 사례

To-Be 개발 방법론은 Hybrid 환경의 전행적인 어플리케이션 개발 프로젝트 지원을 범위로 하고 있습니다.

본 문서에서 소개하는 To-Be 개발 방법론은
J2EE 기반의 신규 어플리케이션 개발 프로젝트를 지원하기 위해 구축된 개발방법론으로 범위를 제한합니다.



소프트웨어 개발 프로세스는 효율적으로 시스템을 구축하기 위한 절차이다.

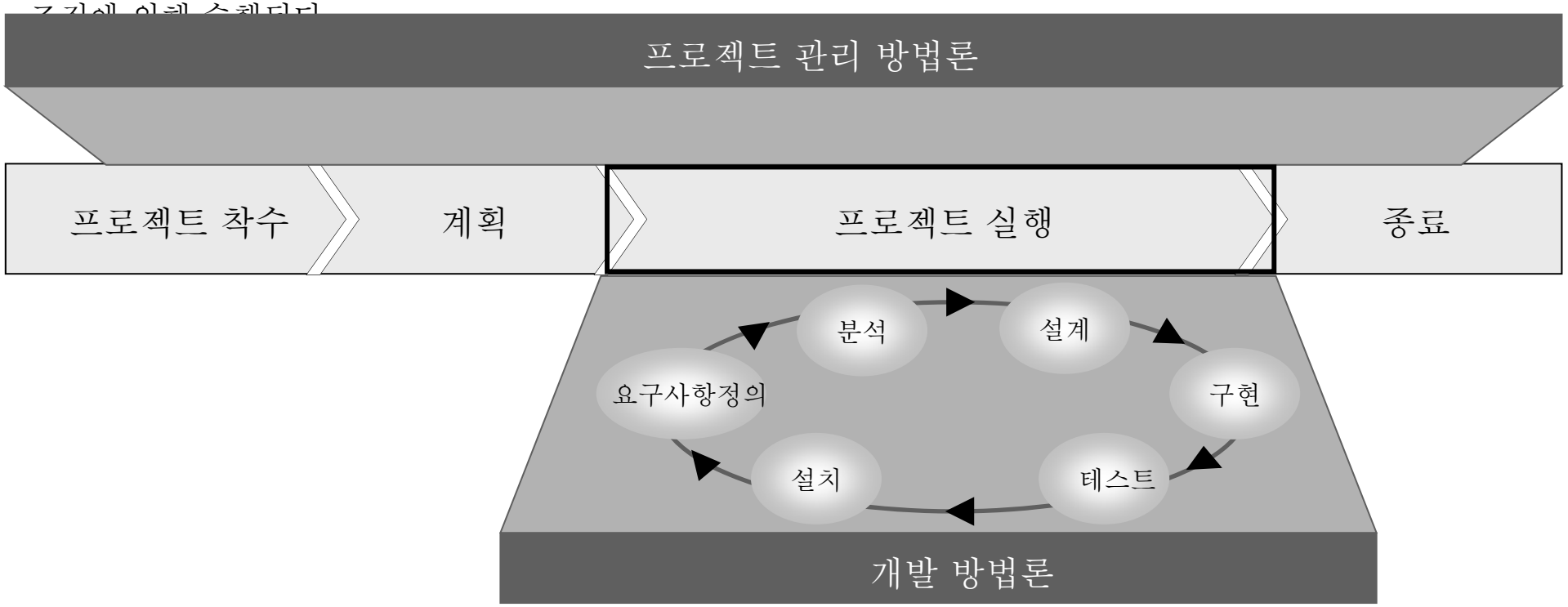
따라서, 누가 무엇을 수행하여 어떤 결과를 내놓아야 하는 가에 대하여 언급 되어야 한다.

- 시스템을 구축하기 위해 어떤 개발자들이 필요하며,
- 각 개발자가 수행해야 할 작업은 무엇이고,
- 이 작업의 수행 목표와 결과가 무엇인지 정의하여야 한다.

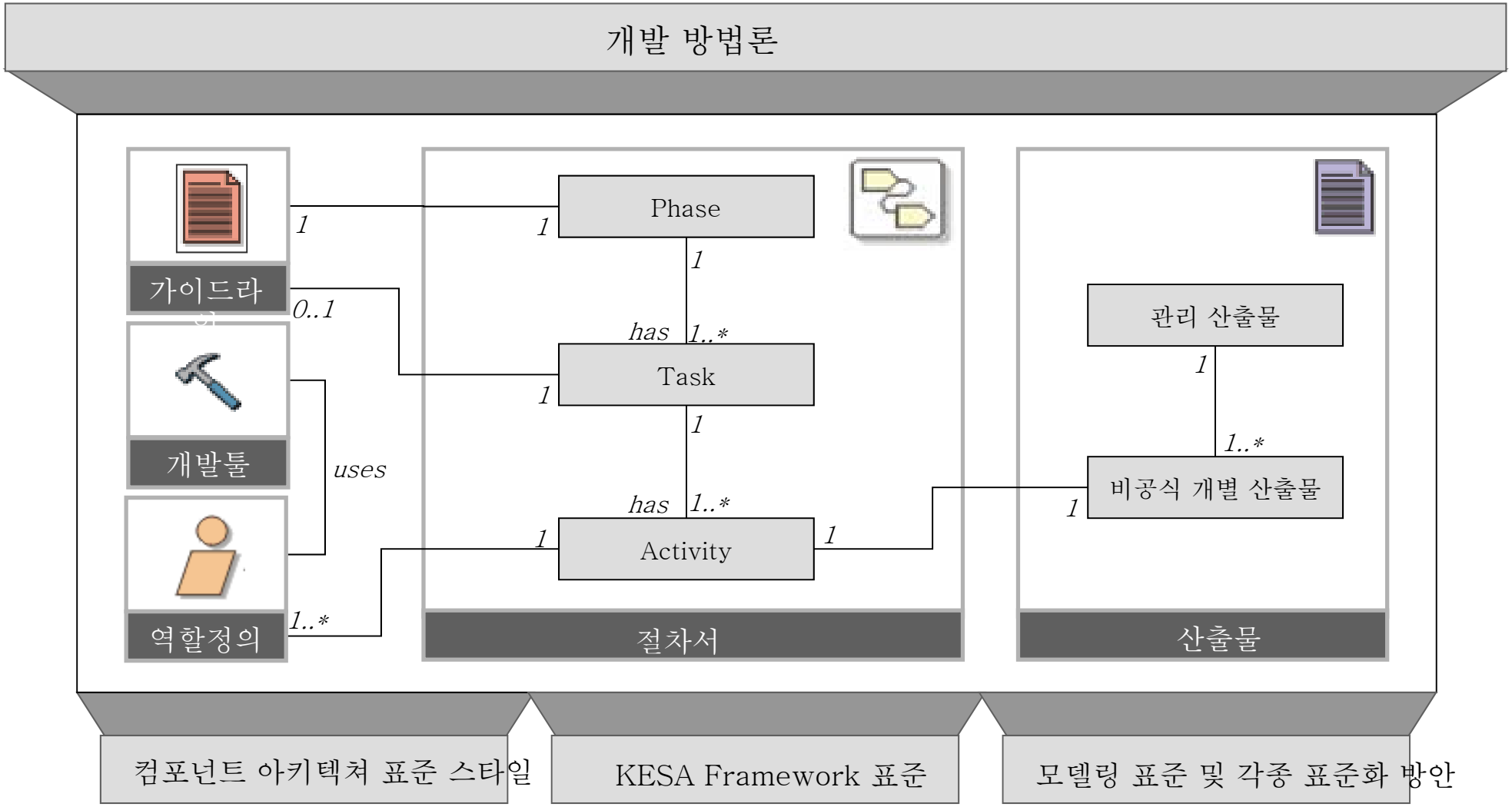
- The Unified Software Development Process (Addison Wesley,1999) 중에서 -

어플리케이션 개발 프로젝트에서 수행하는 작업은 실질적인 개발 활동과 프로젝트 관리 측면의 활동으로 구분 됩니다.

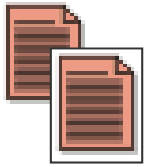
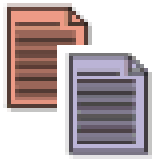
- 실질적인 어플리케이션 개발 활동을 정의하고 있는 개발 방법론은
  요구사항 정의, 분석, 설계, 프로그램 작성, 테스트 등과 같이 어플리케이션 구축에 직접적인 영향을 주는 활동으로 어플리케이션 개발 프로젝트 내의 개발자에 의해 수행된다.
- 어플리케이션 개발 프로젝트 관리 측면의 활동을 정의하고 있는 프로젝트 관리 방법론은
  프로젝트 진행 과정을 관리하기 위한 활동으로 어플리케이션 개발 프로젝트 관리자와 전행 프로젝트를 총괄 관리하는
  주기에 의해 수행된다.



개발 방법론은 다음과 같은 산출물로 구성되어 어플리케이션 개발 프로젝트에 지원 됩니다.



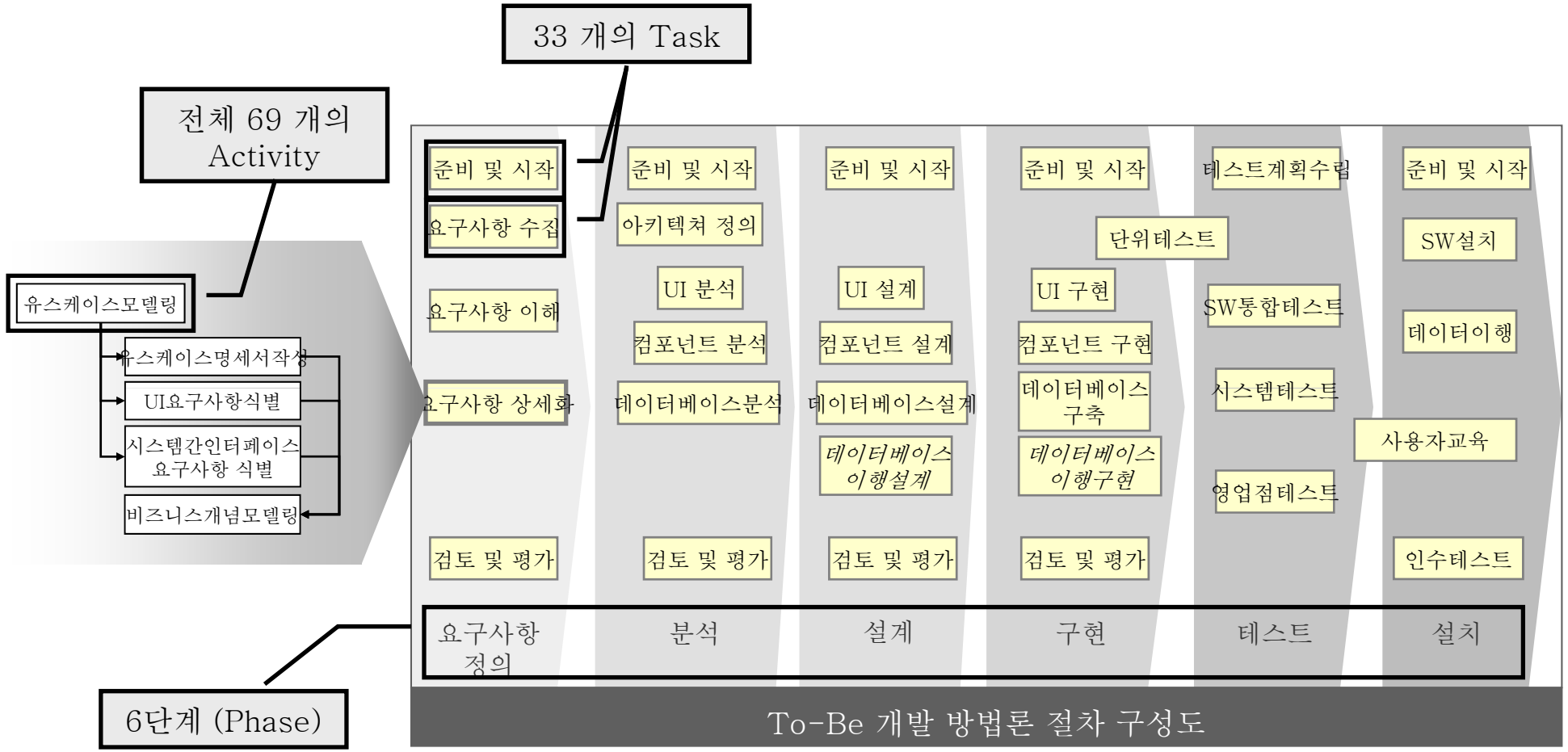
(: MS-Word : MS-PowerPoint : UML Model)

개발방법론 구성	세부 구성 내용
 절차서	❑ 개발 방법론 단계별 절차서(MS-Word) ▪ 요구사항 정의 단계 절차서 ▪ 분석 단계 절차서 ▪ 설계 단계 절차서 ▪ 구현 단계 절차서 ▪ 테스트 단계 절차서 ▪ 설치 단계 절차서 ❑ 어플리케이션 개발 프로젝트 수행 시 이루어 지는 작업 단계 체계 정의 ❑ 단계별 Task(관련 활동의 그룹), Task별 활동(Activity) 정의 ❑ 활동(Activity)별 상세 작업 절차, 역할, 입력물과 출력물, 관련 가이드 및 참고 자료 등을 기술
 **산출물 양식 및 예제**	❑ 관리 대상 공식 산출물 양식 및 예제 ▪ 요구사항 명세서 ▪ 컴포넌트 분석/설계서 ▪ 데이터 분석/설계서 ▪ 각 단계 결과 검토서 ▪ 테스트 계획서 ▪ 테스트 설계서 ▪ 테스트 결과서 ▪ 데이터 이행 계획서 ▪ 데이터 이행 설계서 ▪ 설치 계획 및 설계서 ▪ 운영자 가이드 ▪ 사용자 가이드 ❑ 활동(Activity)에 대한 작업 수행을 위한 비공식 작업 산출물 양식 및 예제 ❑ 요구사항 분석, 컴포넌트 분석 및 설계, 데이터 논리 및 물리 모델 예제
 **가이드 문서**	❑ 각 단계별 단계별 절차 가이드(MS-Word) ❑ 주요 모델링 가이드(MS-PowerPoint) ▪ 유스케이스 모델링 가이드 ▪ 비즈니스 타입 모델링 가이드 ▪ 데이터 모델링 가이드 ▪ 컴포넌트 분석 모델링 가이드 ▪ 컴포넌트 설계 모델링 가이드

(: MS-Word : MS-PowerPoi : UML Model)

개발방법론 구성	세부 구성 내용
툴 가이드	❑ 툴 설치 및 사용자 가이드 ▪ 요구사항 관리 툴 설치 및 사용자 가이드 ▪ 모델링 툴 설치 및 사용자 가이드 ▪ 문서 자동화 툴 설치 및 사용자 가이드 ▪ 개발 툴 설치 및 사용자 가이드 ▪ 형상관리 툴 설치 및 사용자 가이드 ▪ 테스팅 툴 설치 및 사용자 가이드
각종 표준	❑ 요구사항 분석, 컴포넌트 분석 및 설계 모델 표준 ▪ 모델 구성요소 명명 규칙(유스케이스ID, 컴포넌트 및 인터페이스 명명 규칙 등) ▪ 모델 Package 구성 표준 등 ❑ 구현 기술에 따른 표준 및 소스파일 명명 규칙 ❑ 설치 패키지 구성 및 명명 규칙 ❑ 어플리케이션 개발 프로젝트 지원 정보 공유를 위한 서버 Repository 관리 방안 ❑ 형상관리,품질관리 대상 산출물 정의 ❑ 그 외의 UI Design 표준, 데이터베이스 테이블 및 컬럼 명명 규칙 표준 등 관련 담당 파트에서 지원

To-Be 개발 방법론 프로세스는 어플리케이션 구축을 위한 개발자들의 수행 활동을 정의 합니다.
개발 절차의 단계(요구사항 정의, 분석, 설계, 구현, 테스트, 설치)는 관련 작업 활동의 그룹(Task)으로 구성되었으며, 관련 작업 활동(Task)은 상세 수행 활동(Activity)으로 정의되어 있습니다.
- 6 단계, 33 개의 Task, 69 개의 Activity 로 구성



개발 방법론의 단계(Phase)는 크게 6단계로 개발자의 활동을 정의하고 있습니다.

요구사항정의	개발 대상 시스템에 대한 요구사항을 수집 및 상세화 작업을 수행 ❑ 관련 활동을 5개의 Task 로 그룹화 ❑ 11개의 세부 활동으로 정의
분석	시스템에 요구되는 기능적 요구사항을 시스템화 하기 위한 분석 작업을 구현기술과 무관하게 ❑ 관련 활동을 6개의 Task 로 그룹화 ❑ 11개의 세부 활동으로 정의
설계	비기능적 요구사항과 시스템화 하기 위해 구현기술을 적용하여 설계 작업 수행 ❑ 관련 활동을 6개의 Task 로 그룹화 ❑ 12개의 세부 활동으로 정의
구현	실제 프로그램 소스코드를 작성하고, 컴파일 하여 시스템 구현 작업 수행 ❑ 관련 활동을 6개의 Task 로 그룹화 ❑ 8개의 세부 활동으로 정의
테스트	테스트계획 수립 및 소프트웨어 통합/ 시스템 통합/사용자(영업점)/최종 인 테스트를 수행 ❑ 관련 활동을 6개의 Task 로 그룹화 ❑ 22개의 세부 활동으로 정의
설치	실 운영 시스템에 소프트웨어 설치 및 데이터 이행, 사용자 교육 등을 수행 ❑ 관련 활동을 4개의 Task 로 그룹화 ❑ 5개의 세부 활동으로 정의

Phase

단계(Phase)의 정의에 대한 기준은 다음과 같습니다.

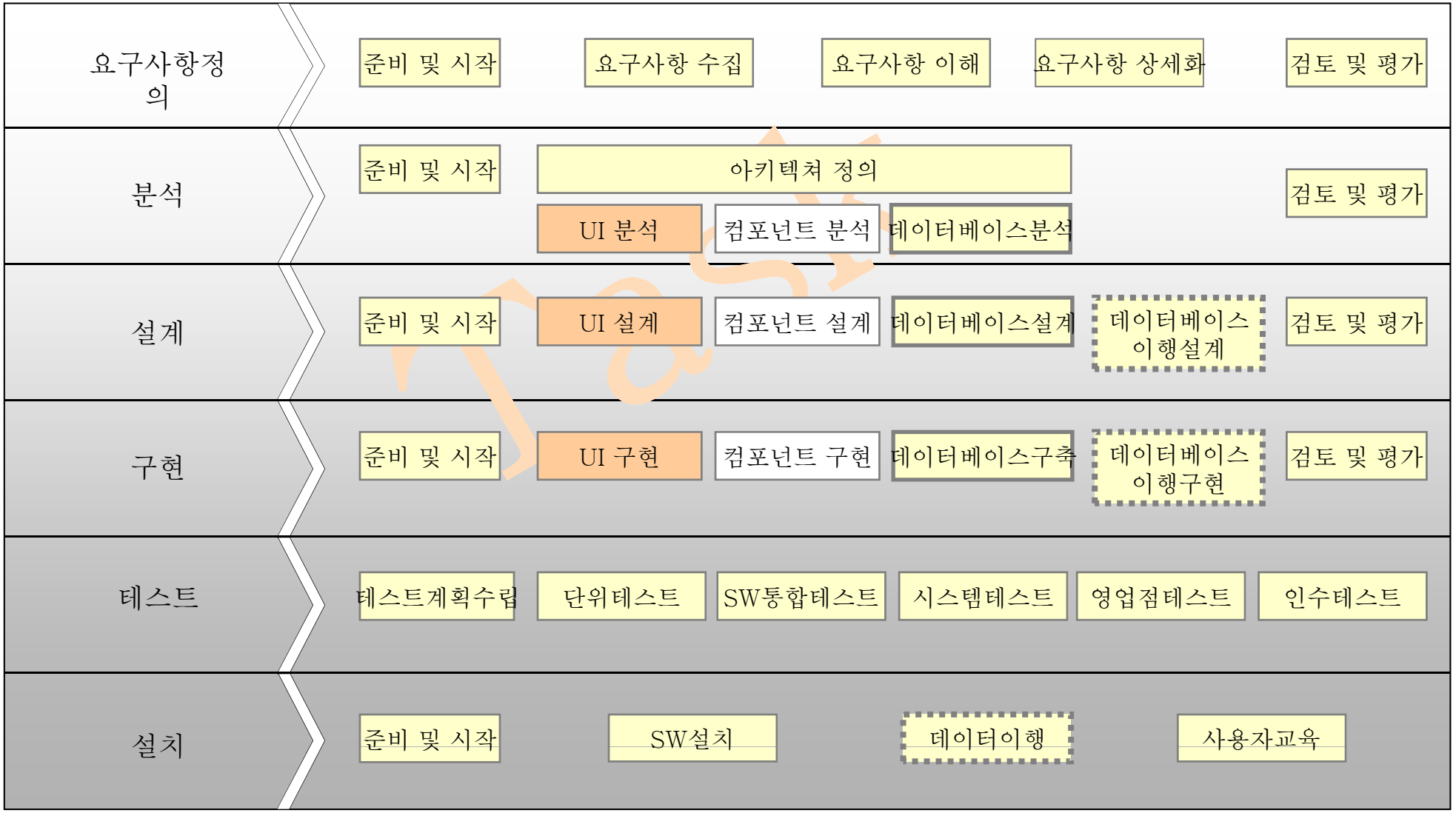
❑ Phase 단계 정의
- ▪ 방법론 구성의 최상위 단위이며, 개발 프로세스의 가장 큰 관리 단위
- ▪ 단계 의미와 목표에 부합하는 활동, 관련 활동의 집합인 Task로 구성

❑ 정의 기준
- ▪단순하면서 일반적으로 사용되는 명칭 및 개념도입(Software Life Cycle Processes)
- ▪소프트웨어 개발 방법론의 일반적인 개발 절차로, 정보공학방법론/RUP/CBD 등의 모든 방법론에서 적용하는
  공통적인 개발 단계
- ▪ 기존 소프트웨어 개발 경험이 있는 개발자 및 소프트웨어 개발 프로젝트 관리자에게 익숙한 개발 단계

❑ 고려 사항
- ▪ RUP의 비즈니스 모델링은 IT관점이 아닌 비즈니스 관점으로 수행되어야 할 활동이므로 제외한다.
- ▪ 소프트웨어 개발 프로젝트의 착수 단계에 해당하는 프로젝트 발주 및 사전 준비 활동은 개발 절차에서 제외, 프로젝트 관리
  프로세스를 준수한다. 또한, 형상관리/품질관리 등의 소프트웨어 개발 프로젝트의 관리, 평가 측면의 활동은 각 형상관리
  프로세스, 품질관리 프로세스를 준수한다.
- ▪ 테스트 단계의 상세 프로세스는 개발관리팀의 테스트 전담 파트에서 정의하되, 소프트웨어 개발 방법론 내에 포함되어
  개발 프로젝트에 지원된다.
- ▪시간적 수행 순서와 일치하지 않음(테스트 단계의 경우, 구현 작업 완료 이후 수행되는 것은 아님.) .
- ▪WBS 및 Iteration 계획을 제공하기 위한 단위기준 제시 .

Phase

개발 방법론의 각 단계(Phase)는 관련 활동의 집합인 Task로 다음과 같이 구성되어 있습니다.



Task 정의에 대한 기준은 다음과 같습니다.

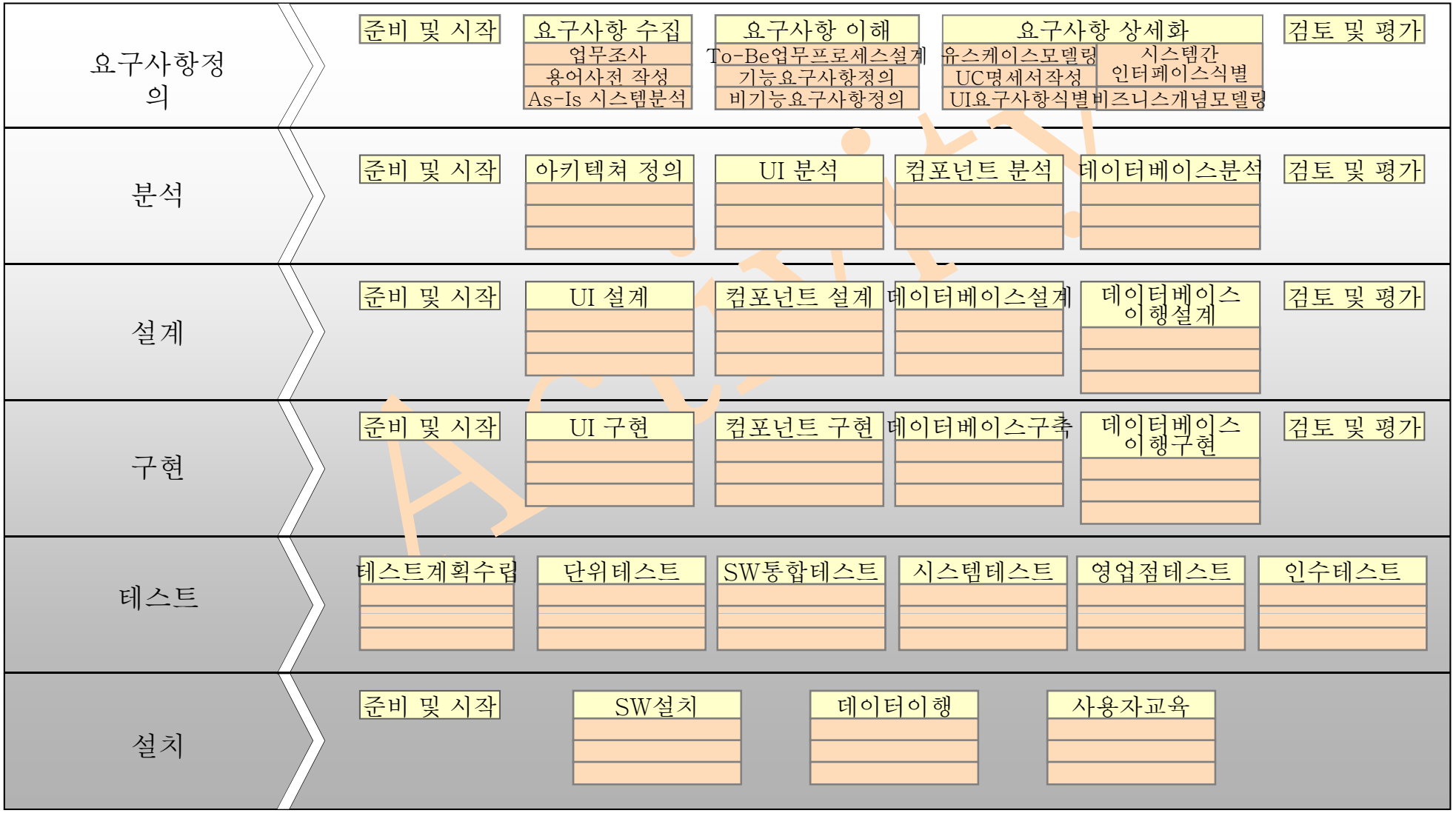
❑ Task 단계 정의
- Task는 단계(Phase) 내부의 논리적인 개발 활동 작업의 단위
- 소프트웨어 개발을 위한 개발자들의 상세 활동 중 동일한 목적을 위해 수행되는 활동을 그룹핑 하여 Task로 구성

❑ 정의 기준
- 동일한 목적의 상세 활동(Activity)을 그룹화 하여 Task 단위로 구성
- 분석, 설계, 구현 단계(Phase)를 구성하는 Task는 Layered Architecture 를 적용하여 명명의 일관성을 유지

❑ 고려 사항
- 소프트웨어 개발 프로젝트 환경에 따라 Task 단위 Customizing 하여 적용하도록 함
  (예. 신규 개발 시스템인 경우, 데이터 이행과 관련된 Task를 제외(*Optional로 구분*))
- 단계(Phase)는 “준비 및 시작”의 Task를 첫 활동으로 각 단계별 계획 및 품질평가 기준 사전 수립의 활동 수행토록 함

Task

개발 방법론의 각 Task는 개발자의 상세 활동(Activity)으로 구성되어 있습니다.



상세 활동(Activity) 정의에 대한 기준은 다음과 같습니다.

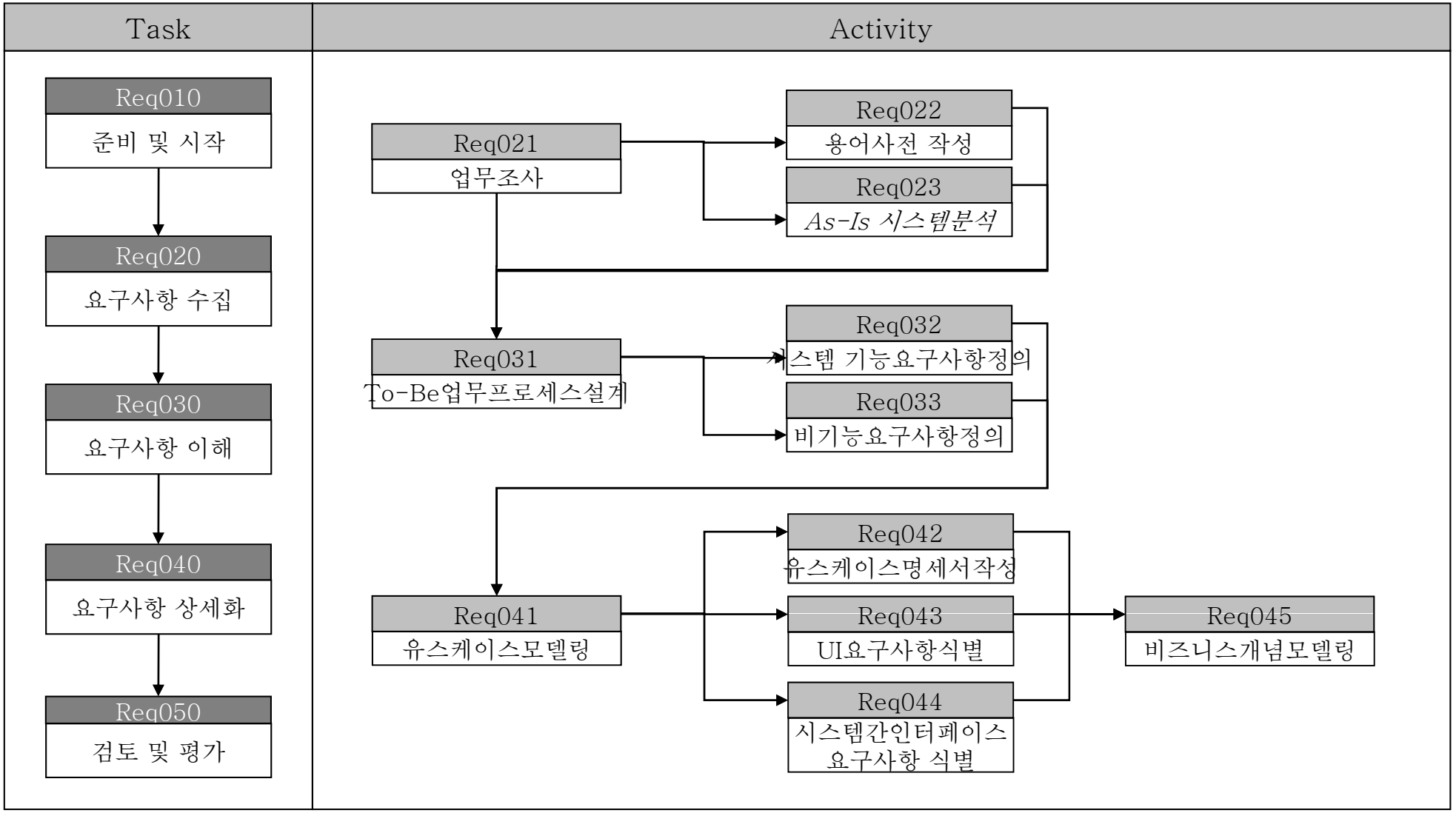
❑ Activity 단계 정의
- Task를 구성하는 상세 작업 활동의 단위 즉, 개발 방법론 절차의 최소 단위
- 작업 활동을 수행하기 위한 입력물과 수행 결과의 출력물의 정의 단위
- 모든 활동은 역할자가 정의됨.

❑ 정의 기준
- Activity는 어플리케이션 개발 프로젝트 내의 역할자가 수행하는 일의 단위로 구성하기 위함.
- 활동은 어플리케이션 개발을 위한 의미있는 결과를 제공하는 목적이 명확해야 함
- 어플리케이션 개발 프로젝트의 효율적인 관리를 위해 활동 수행의 기간은 최대 일주일을 초과하지 않는 단위로 정의
  (단, 어플리케이션 개발 프로젝트의 투입 인력에 따라 조정될 수 있음)

❑ 고려 사항
- 어플리케이션 개발 프로젝트 환경(프로젝트 규모, 기간, 투입 인력, 상용 패키지 연계, 외주 개발 등)에 따라 조정될 수 있음.
- 설계 및 구현 단계를 구성하는 작업 활동은 적용 플랫폼에 따라 Customizing 되어 적용될 수 있음
  (예. 신규 개발 시스템인 경우, 데이터 이행과 관련된 Activity를 제외(*Optional로 구분*))

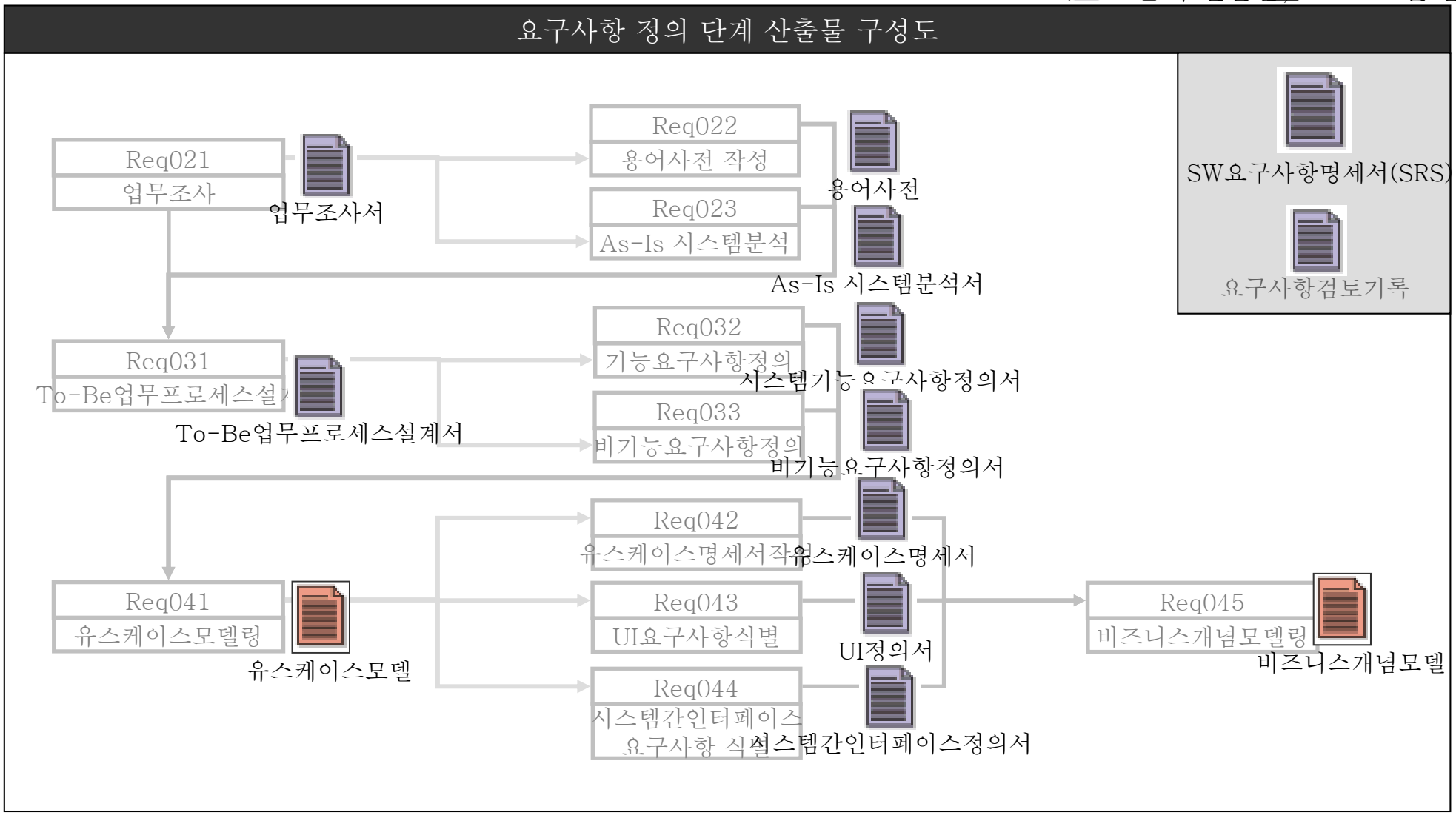
Activity

❑ Req000:요구사항 정의 단계의 개발 절차(Task→Activity) 구성

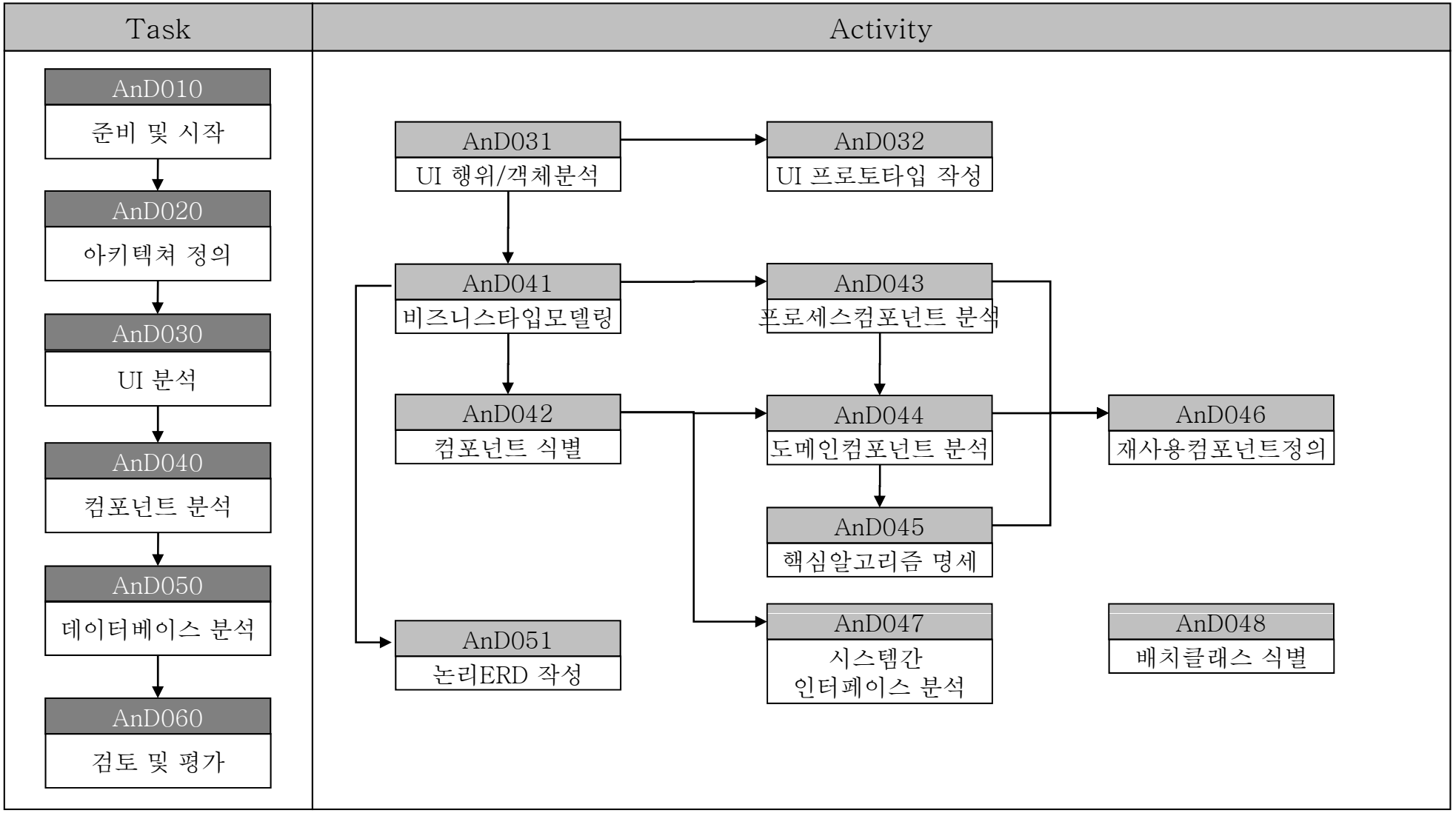


❑ Req000:요구사항 정의 단계의 산출물 구성

(- 문서 산출물, - 모델 산출물

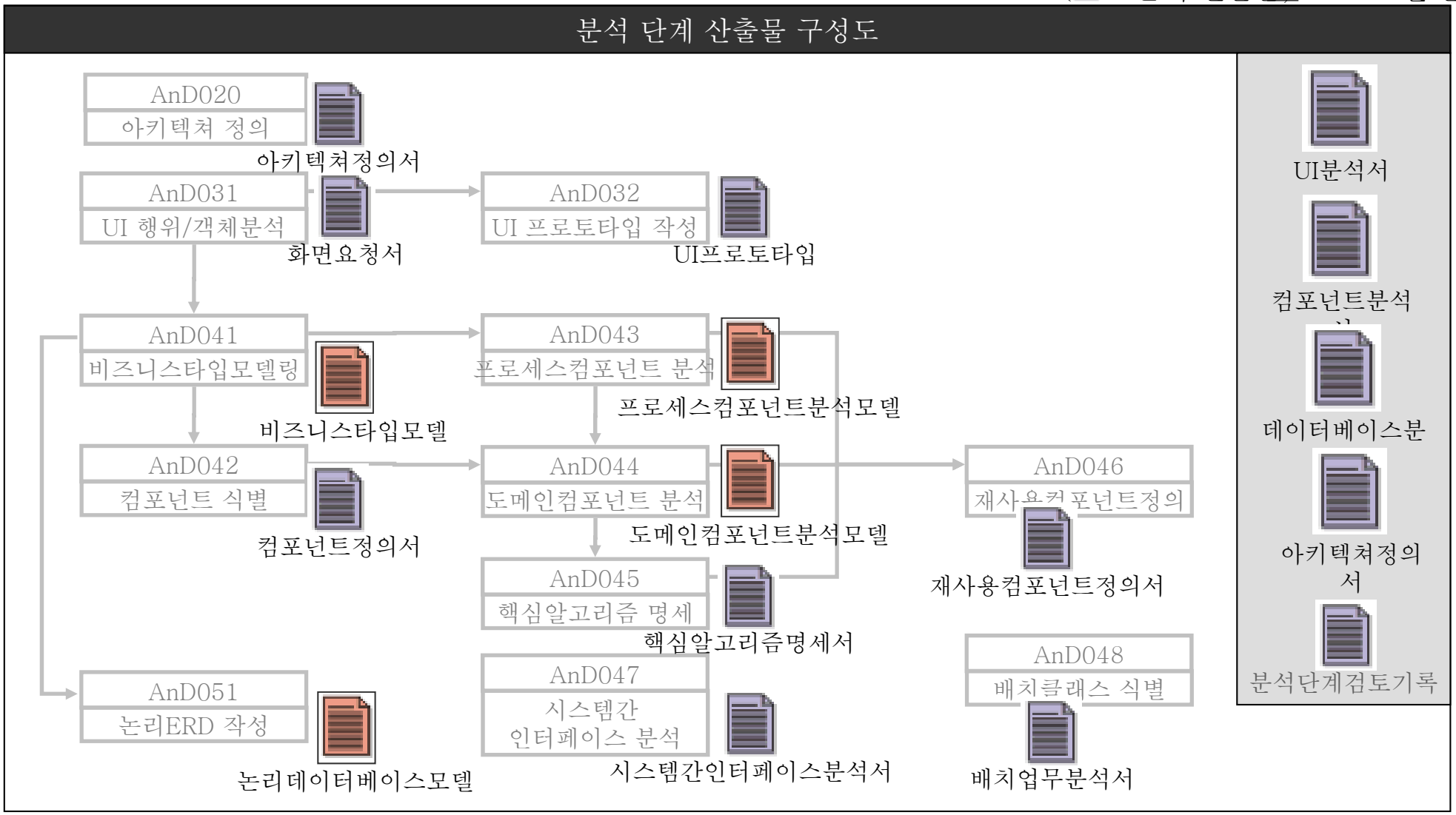


❑ AnD000:분석 단계의 개발 절차(Task→Activity)

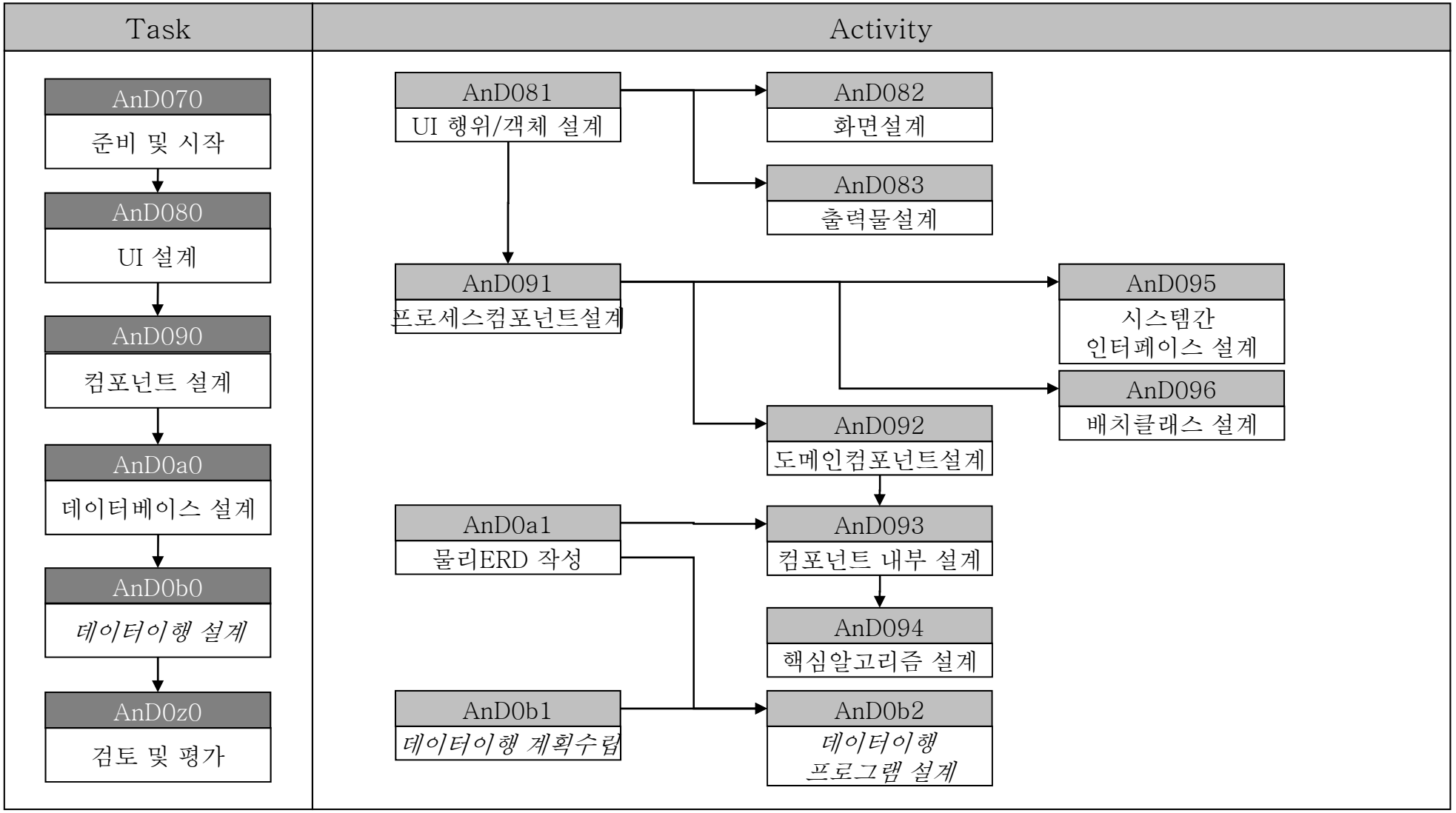


❑ AnD000:분석 단계의 산출물 구성

(- 문서 산출물, - 모델 산출물

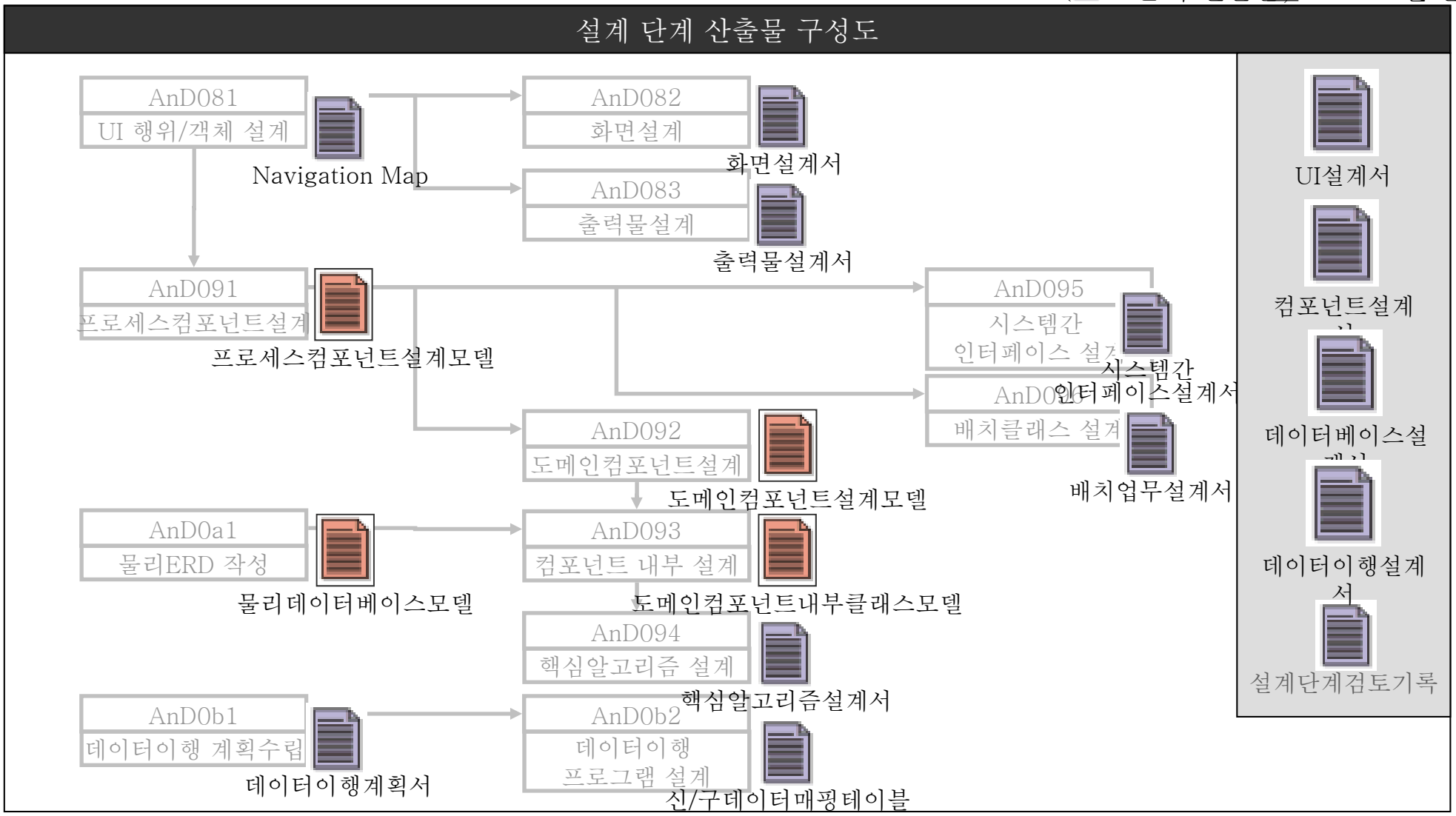


❑ AnD070:설계 단계의 개발 절차(Task→Activity)

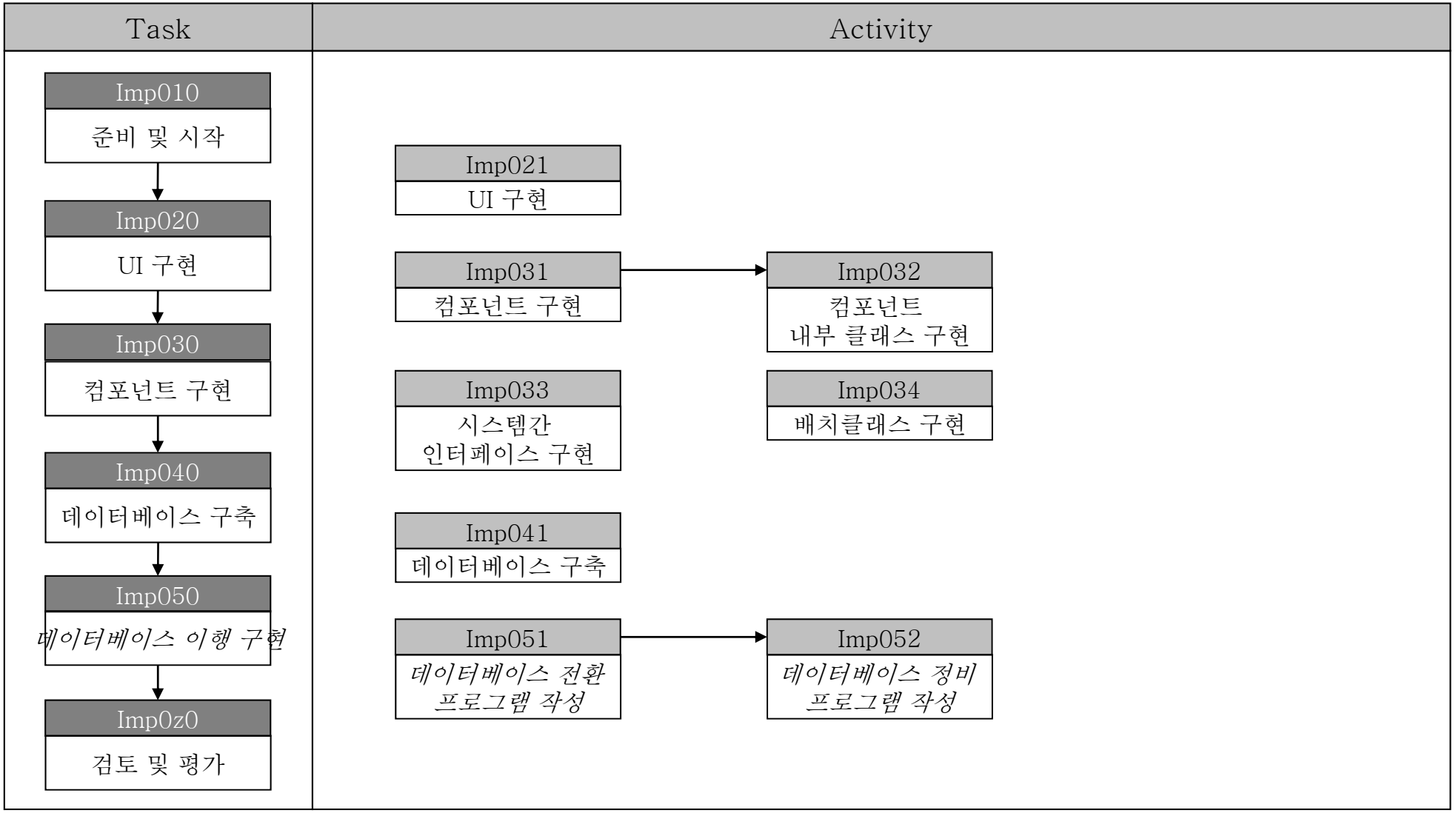


❑ AnD070:설계 단계의 산출물 구성

(- 문서 산출물, - 모델 산출물

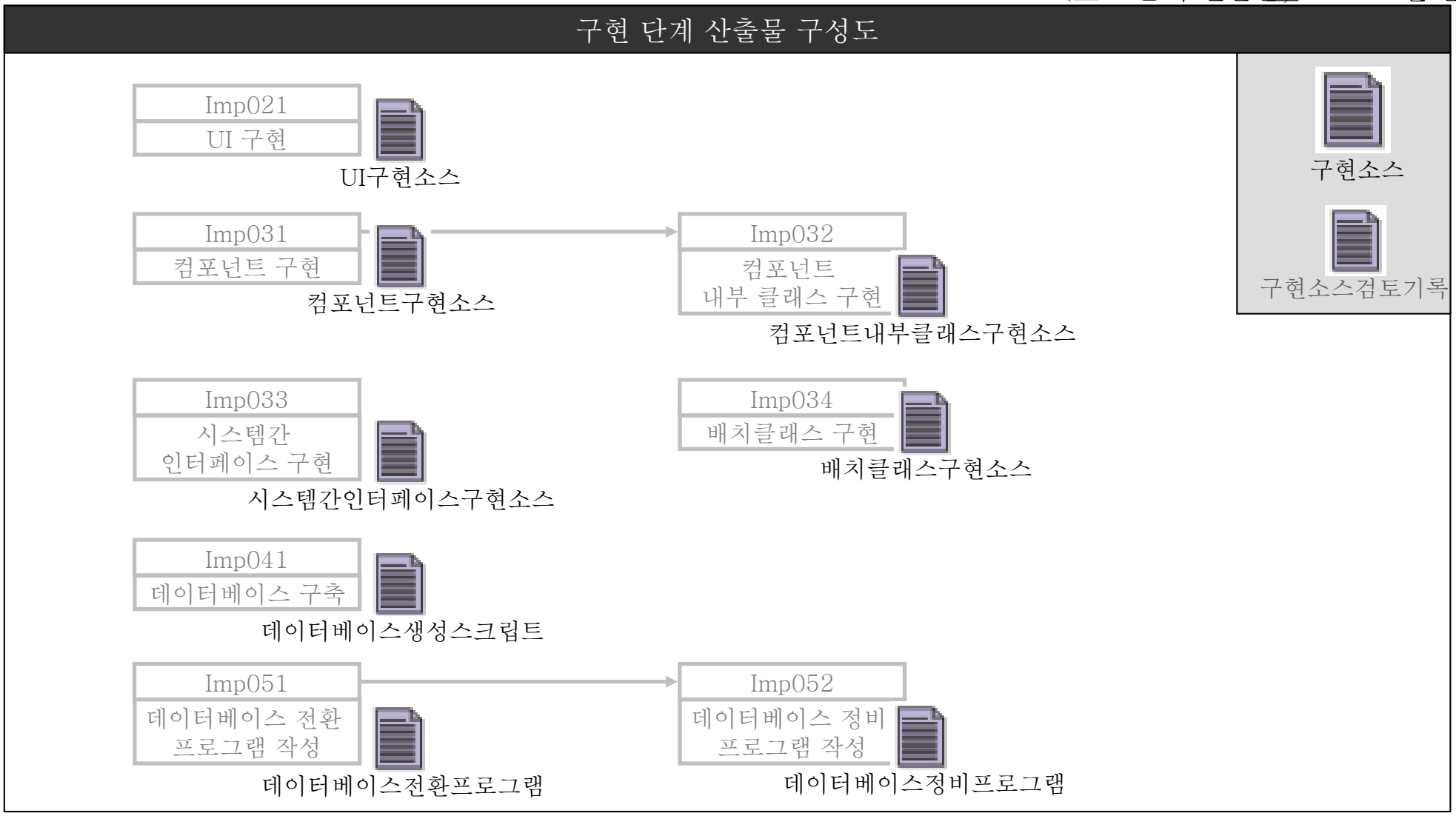


❑ Imp000:구현 단계의 개발 절차(Task→Activity)

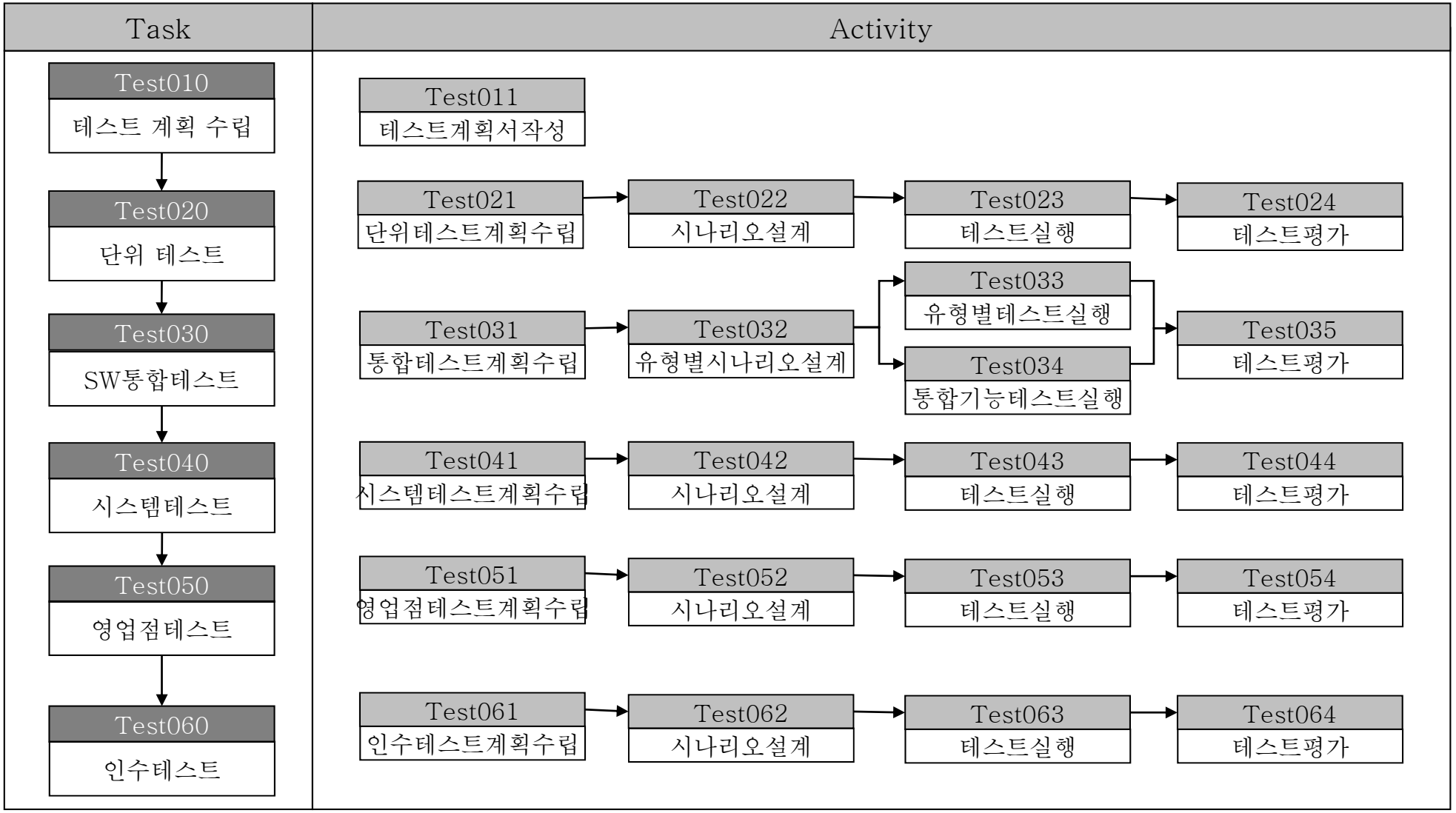


❑ Imp000:구현 단계의 산출물 구성

(- 문서 산출물, - 모델 산출물



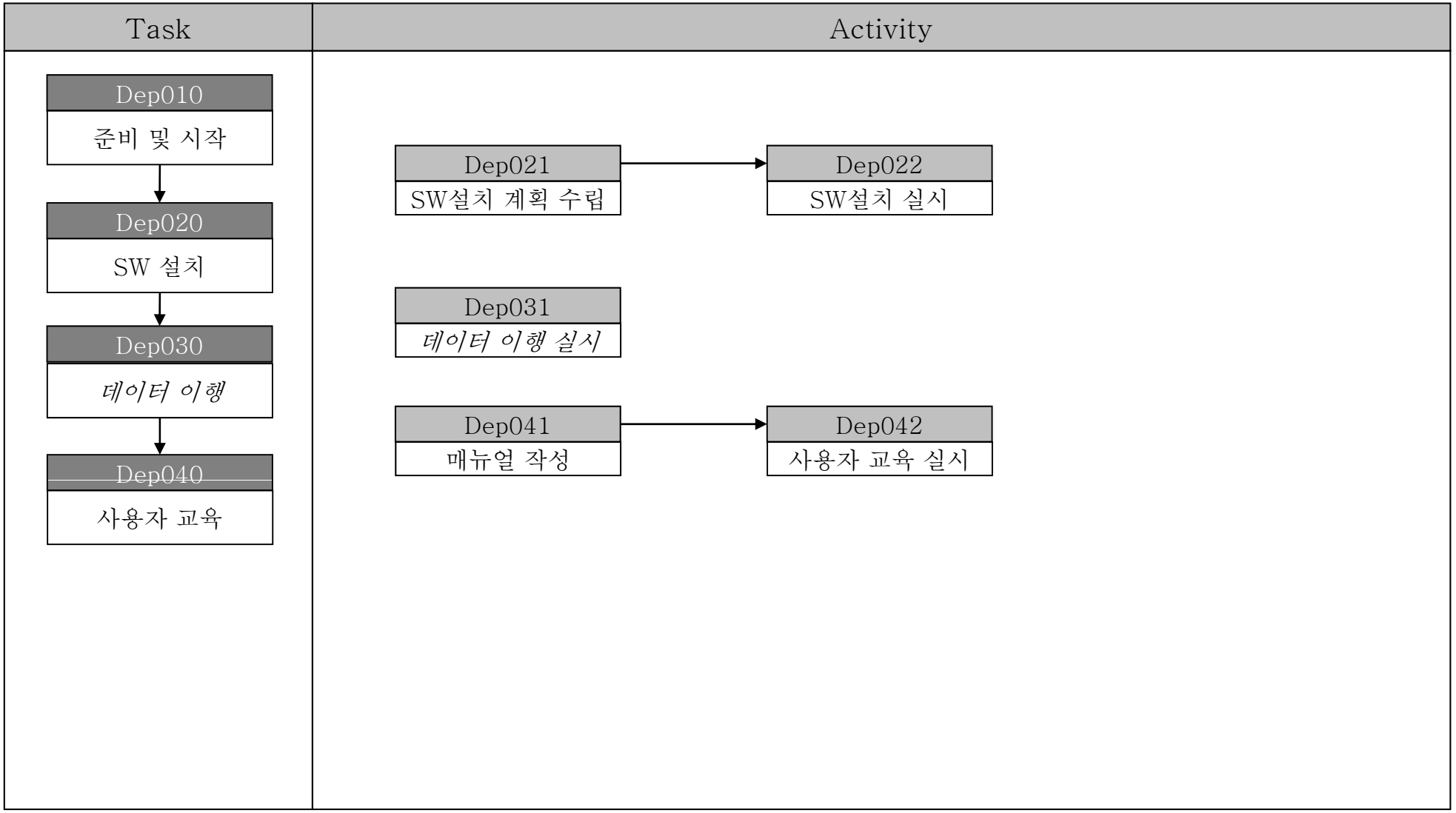
❑ Test000:테스트 단계의 개발 절차(Task→Activity)



❑ Test000:테스트 단계의 산출물 구성

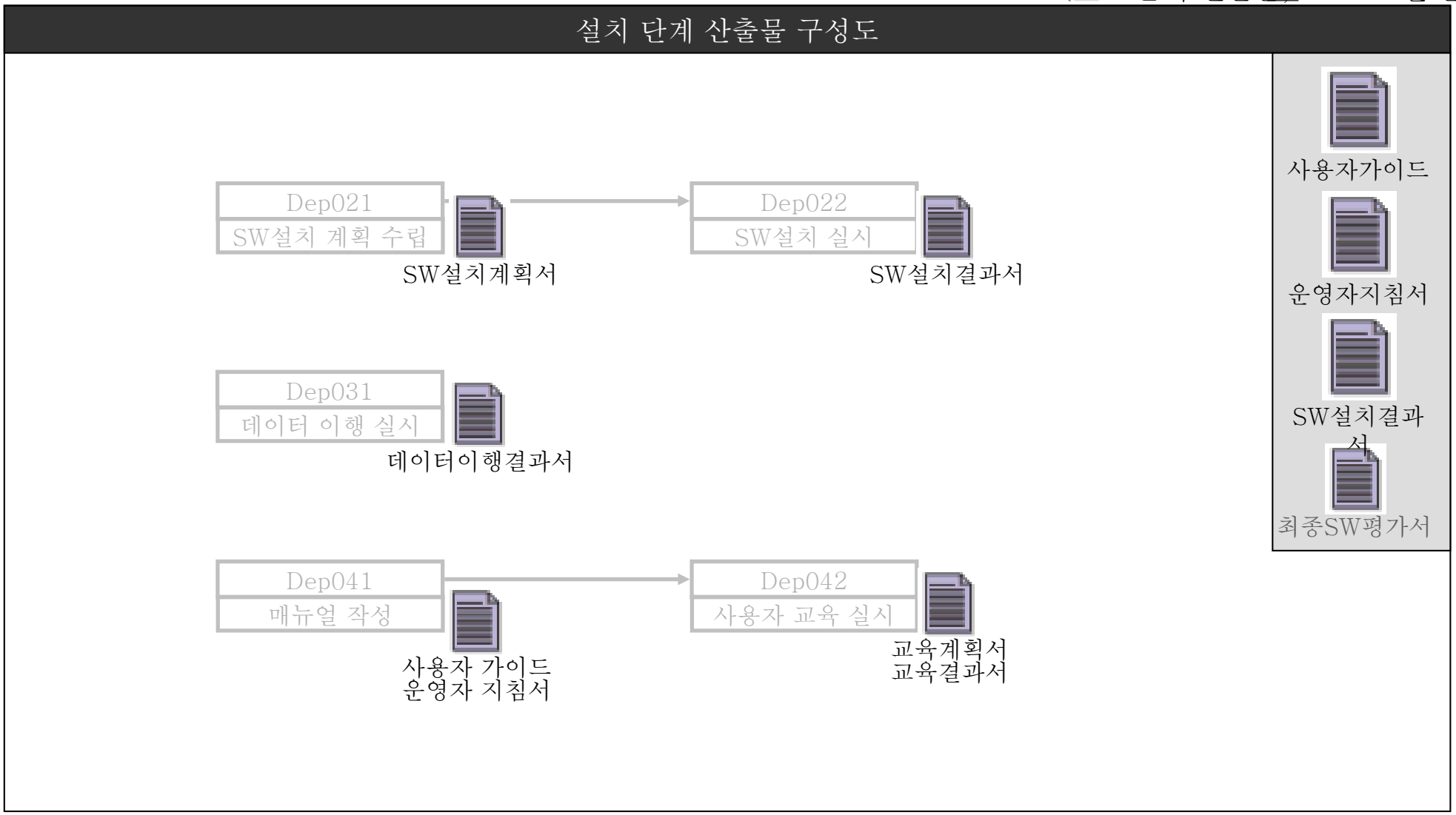
(- 문서 산출물 - 모델 산출물

❑ Dep000:설치 단계의 개발 절차(Task→Activity)



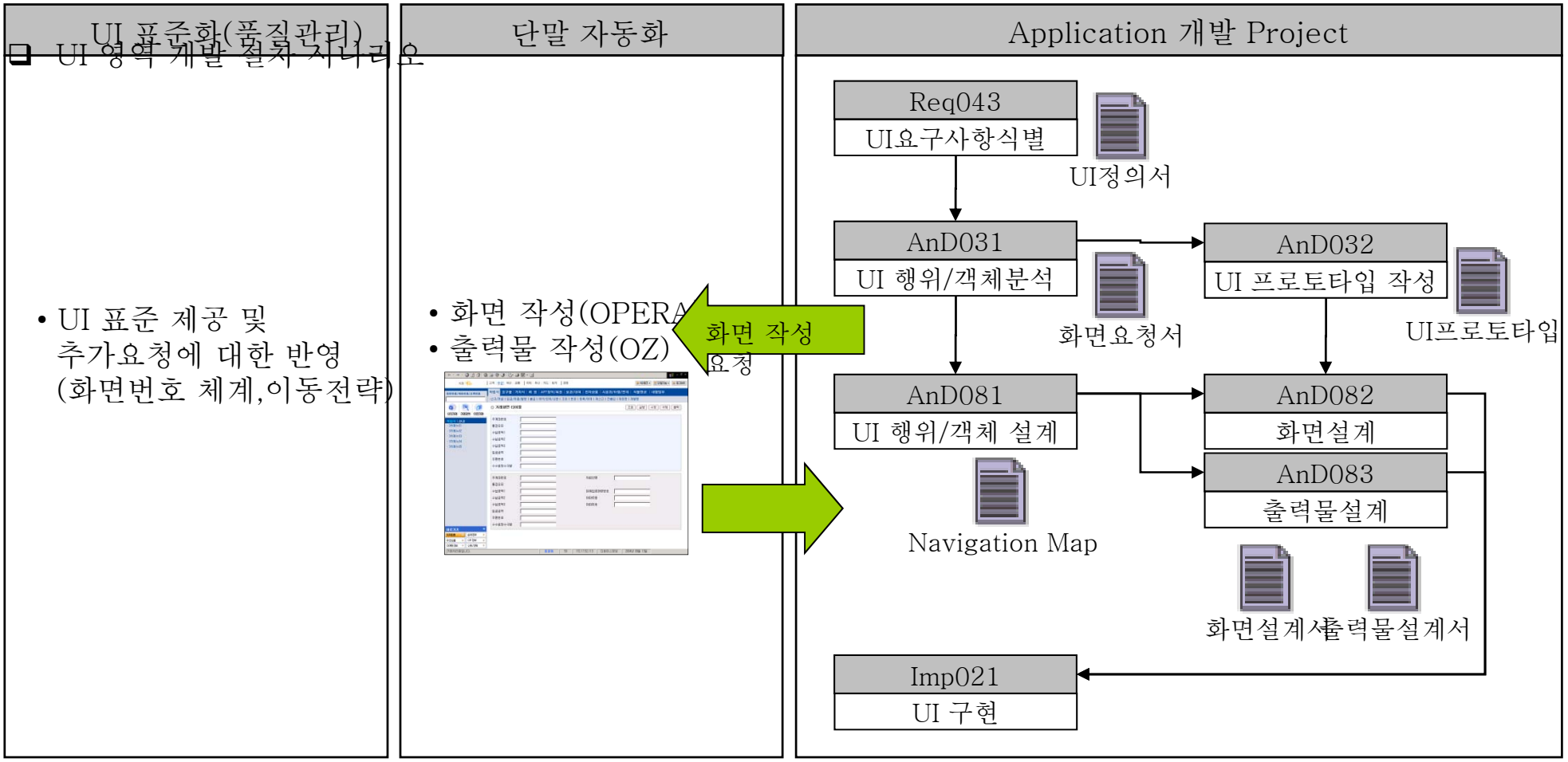
❑ Dep000:설치 단계의 산출물 구성

(- 문서 산출물, - 모델 산출물

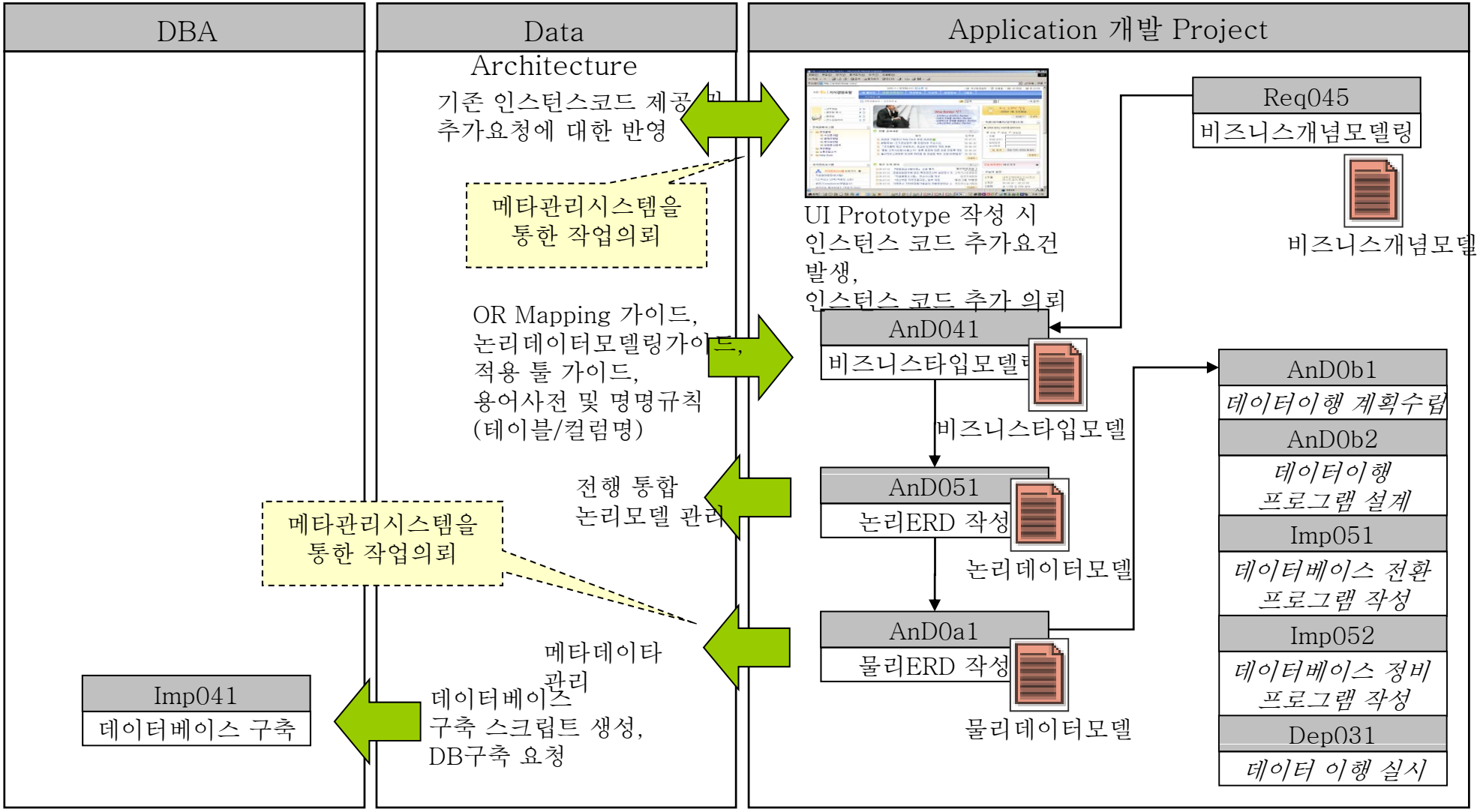
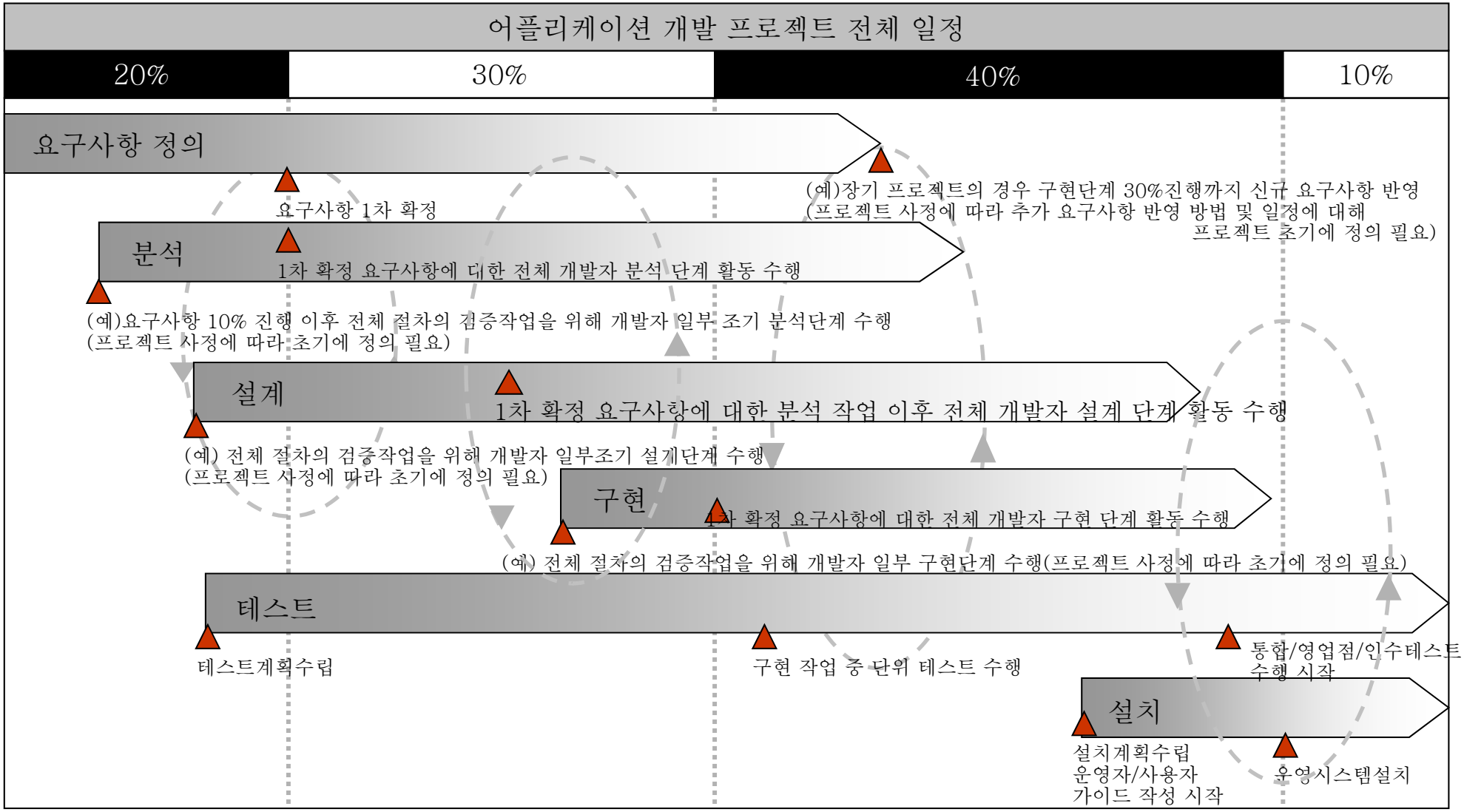


개발 방법론의 절차는 소프트웨어를 구성하는 영역을 UI, Component, Database 로 구분하여 각 단계의 목적에 따라 상세화 됩니다.
개발 방법론을 어플리케이션 개발 프로젝트에 적용할 경우, UI 영역과 Database영역은 타팀과의 협업을 통해 완성되게 됩니다.
(타팀과의 협업 내용은 개발 방법론 1차 완성 당시(2005.07)의 팀별 역할을 기준으로 작성 되었습니다.)

❑ UI 영역 개발 절차 시나리오



❑ Database 영역 개발 절차 시나리오



어플리케이션 개발을 위해 To-Be 개발 방법론을 적용할 경우, 다음과 같이 계획을 수립합니다.