

---

# **BPM을 위한 어플리케이션 아키텍처**

---

- 
- Application Architecture
  - Data Architecture
  - Technical Architecture
  - 시스템 설계 구축 검증
  - DRA 단계별 확인 사항

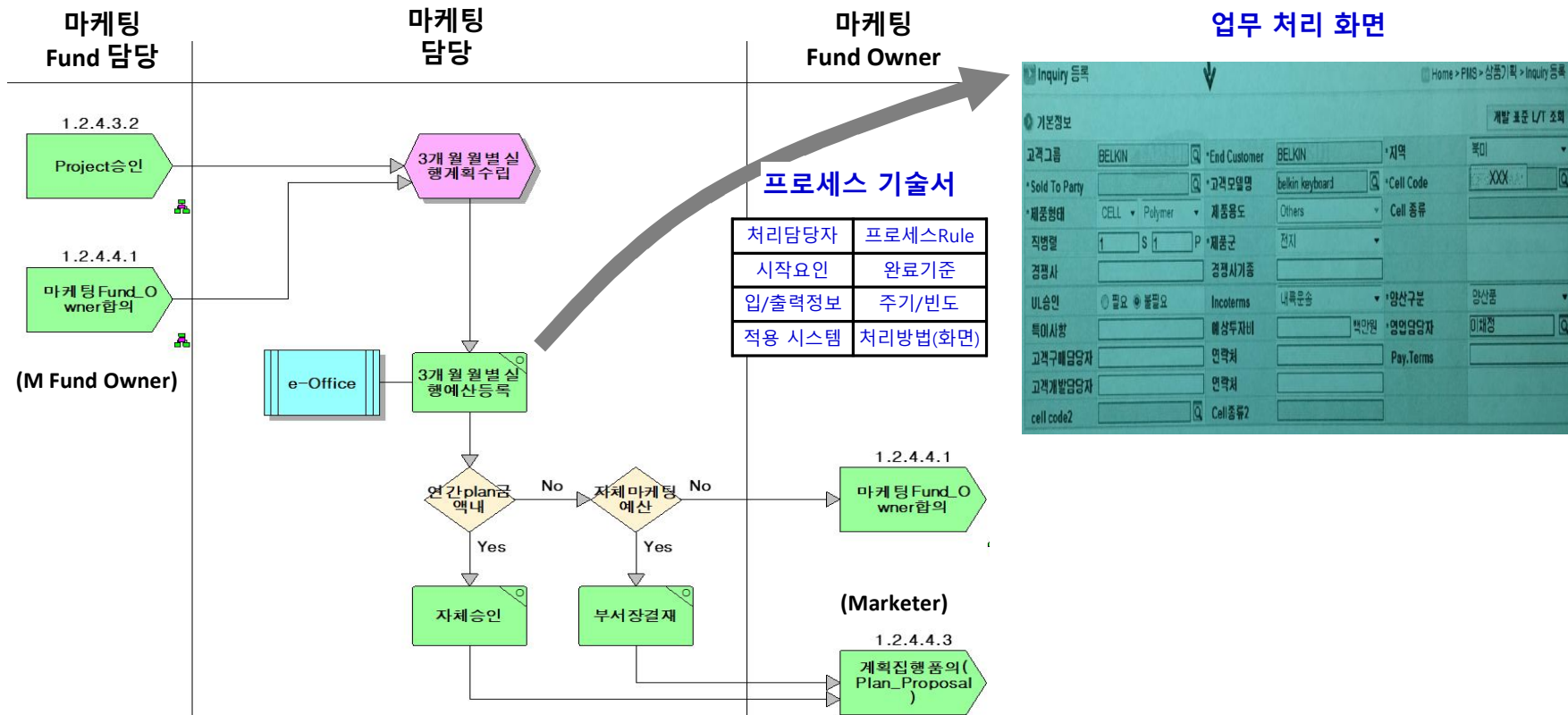
# Application Architecture

## ■ Application Architecture (AA)

Application Architecture(AA)란 프로세스를 구성하는 단위 업무를 처리하기 위해 시스템 내에 화면을 설계하는 것으로 Process에 종속되어 일체화 관리가 필수적임

→ 프로세스가 변경되면 그에 따라 업무처리 화면도 동시에 변경

→ 운영 시스템에서 Process와 Application은 한 몸 (Activity/Task : 업무처리화면 = 1 : 1<n)

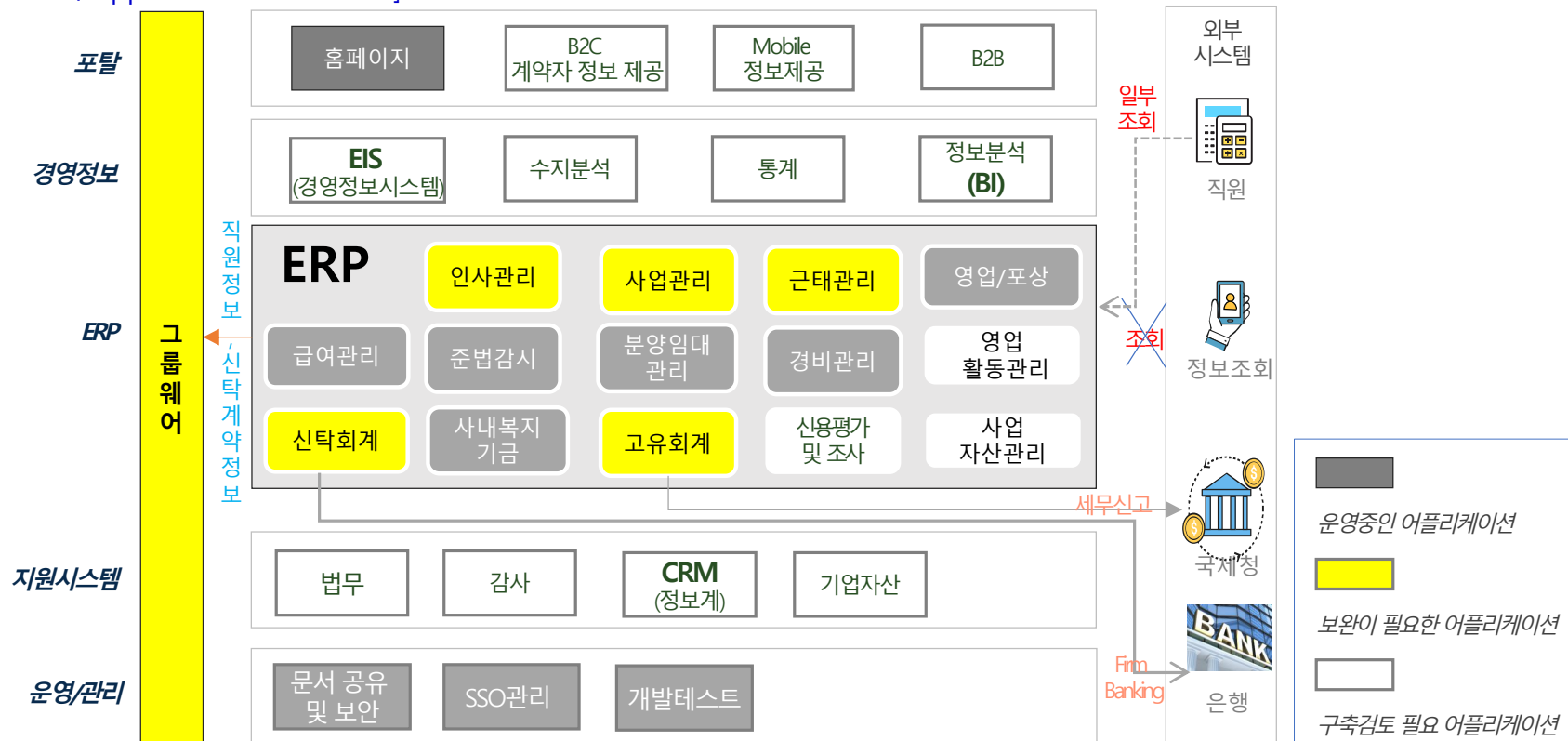


# Application Architecture

## ■ 시스템의 종류

- 운영계 (Operational) : 업무 처리를 위한 시스템 (Transactional 이라고도 함)
  - ERP, CRM, SRM, PLM, ...
- 분석계 (Analytical) : DB, DW, DL 등에 수집된 Data를 분석하는 시스템
  - Standalone : Minitab, OLAP : Spotfire, Server : SAS,... SaaS : Alteryx, Heart count, ...
- 기능계 (Functional) : 타 시스템과 연계 없이 단위 기능 수행 (SaaS)

[금융업의 Application Architecture]



# Application Architecture

## ■ 기능 요구사항 정의

업무 프로세스 분석 과정에서 파악된 이해 당사자들의 요구사항을 분석, 정리하고  
업무 처리 화면(UI)에서 기능 구현이 가능한 지 검토

→ 기능 요구사항 정의서 를 작성 (프로세스 정의서의 요구사항과 연동)

프로세스 명	프로세스 명 : 프로세스 번호와 이름 (계층 구조에 따른 번호)					
구분	항목	상세내용	요구 부서	중요도	검토 의견	수용 여부
		- 구분 : 화면이 구현되는 Activity				
		- 항목 : Activity 또는 Activity 내에서 이루어지는 세부 Task ( ex) 입력, 등록, 전달, I/F, 처리, 통보 등)				
		- 상세 내용: 항목에 대해 요청하는 내용 또는 작업 편이를 위한 요청 사항 등을 기술				
		- 요구 부서 : 업무처리 화면을 사용 하는 부서 (업무 프로세스를 운영하는 기능부서) 또는 업무 프로세스 운영과 관련하여 시스템 기능을 요구 한 이해 당사자				
		- 중요도 : 요구사항에 대해 기능적으로 반드시 필요한 지, 사용자 편의성, 유용성 등을 고려하여 상/중/하 Level로 구분				
		- 검토 의견 : 요구사항에 대해 설계에 반영 할 수 있는 지 여부를 판단				
		- 수용 여부 : 어플리케이션 설계에 반영 여부 판단 (수용/기각/제외 로 구분)				

# Application Architecture

## ■ 기능 요구사항 정의

→ 기능 요구사항 정의서 작성 (예)

프로세스 명	1.2.2 생산 관리					
구분	항목	상세내용	요구 부서	중요도	검토 의견	수용 여부
작업지시	작업지시	- ERP에서 해당작업지시에 주/야 실적, 불량정보를 인터페이스 받을 수 있어야 함.	생산 부서	상	- 작업자정보는 ERP에서 등록하고, 실적/불량정보는 MES에서 ERP로 전송	수용
작업지시	작업지시	- 오더 TYPE이 구분되면 좋을것 같다.	생산 부서	상	- ERP도 구분이 필요함.	수용
생산실행	재작업 관리	- 공정 별 재 작업 카운트를 작업자가 단말기를 이용하여 입력하여 재 작업 카운트를 관리	생산 부서	상	- 작업자가 재 작업 수량을 입력할 수 있는 기능	수용
생산실행	실적처리	- 고온전지CELL별 실적 처리시, 여러 건을 스캔하여 처리할 수 있도록 요청	생산 부서	상	- 여러 건을 동시에 스캔 하기엔 어려운 상황으로 별도 방법으로 처리	보류
생산실행	LOT관리	- 생산LOT ID 체계 및 채번 규칙 관리 필요	자재 부서	상		수용

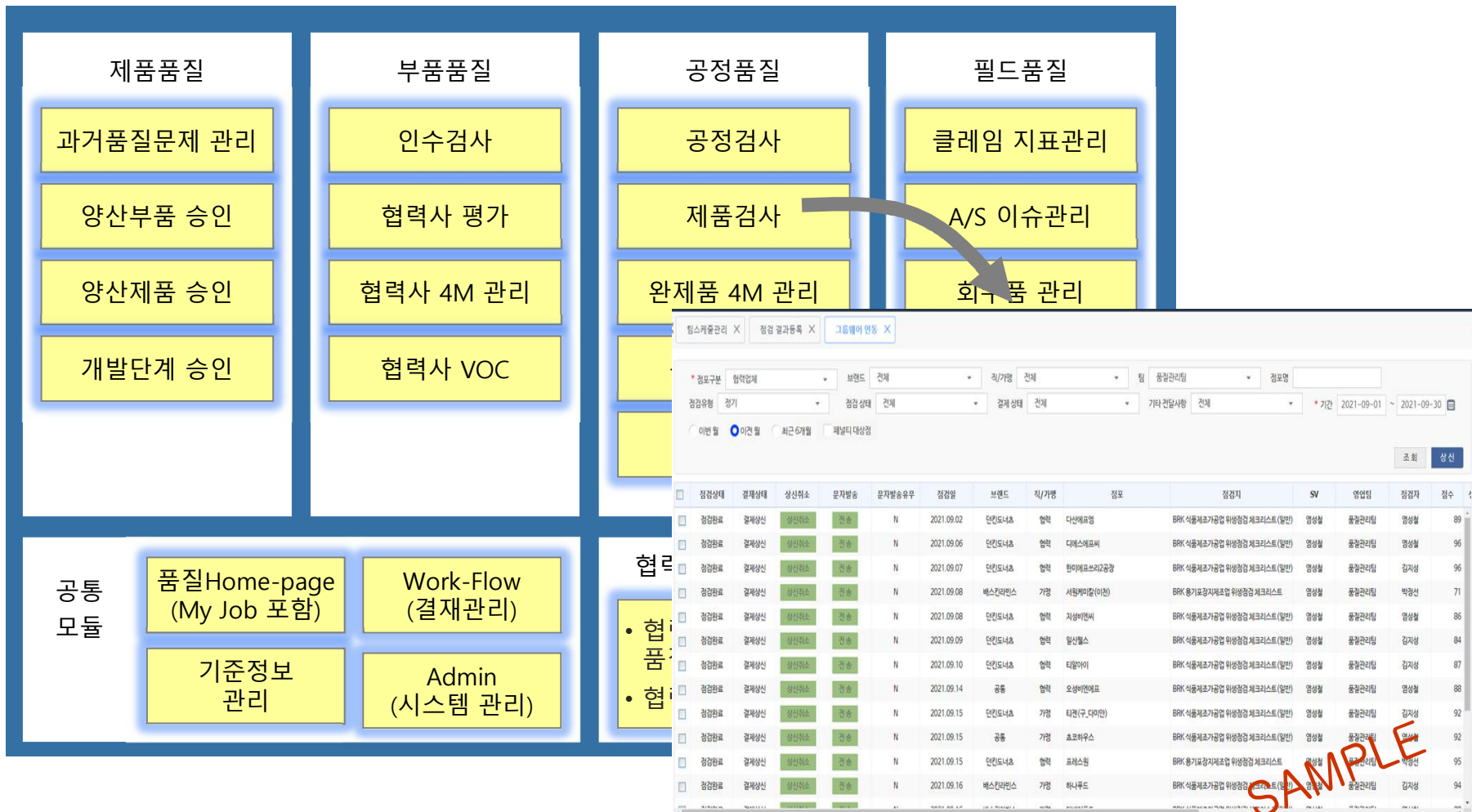
SAMPLE

# Application Architecture

## ■ 어플리케이션 구조 및 화면 설계

화면 설계(UI) 대상의 단위 업무를 Block 구조로 체계화 하고, 각각의 화면을 설계

[품질관리 시스템]



# Application Architecture

## ■ 화면 기능 설명

설계된 화면 (UI) 별로 처리하는 업무 기능을 설명하고, 구성된 항목에 대한 작업 내용을 설명

[자재/제품 설정 상세 팝업]

자재/제품 설정 상세

자재/제품 코드 \* 1000000001      자재/제품명

1 상세 정보    2 그룹    3 CMF    4 플로우 부여    5 공급사 부여    6 고객사 부여

일반

자재/제품 설명 페루산 동정광      자재/제품유형 \* 원자재

자재/제품 그룹 ANODE      단위 \* TON

자재차감유형 Backflush      LOT 수량 11

제품크기

총중량 19.000      순중량 0

단위당 부피 0      가로 치수 0

세로 치수 0      높이 0

포장

소포장수량 0      포장 수량 5

취소    저장

### ■ 기능 설명

1. 자재/제품 상세 정보를 입력하는 팝업 화면으로 상세 정보, 그룹, CMF, 플로우 부여, 공급사 부여, 고객사 부여 탭으로 구성
2. 제품의 상세 SPEC와 더불어 생산 플로우, 공급사, 고객사를 설정

### ■ 작업 순서

→ 표시된 번호 순으로  
작업 내용 설명 (추가)

- 1.
- 2.
- 3.
- 4.

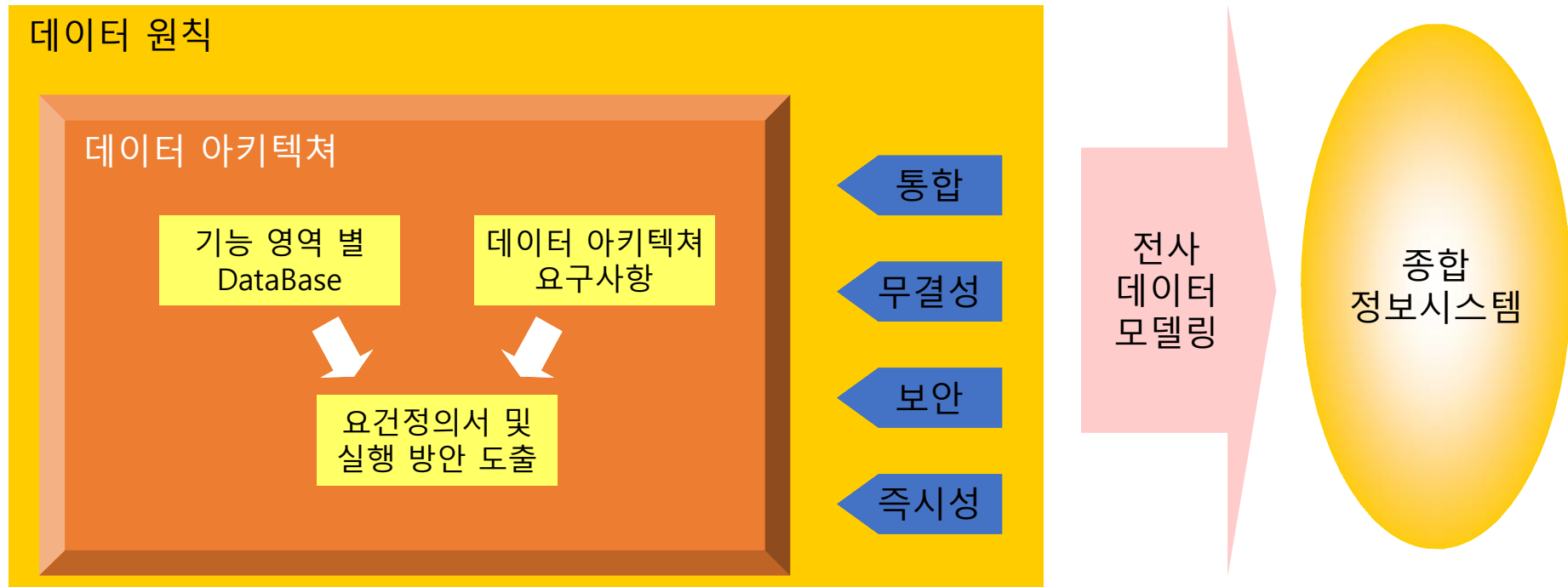


# Data Architecture

## ■ Data Architecture (DA)

업무의 기능 영역 별 데이터베이스의 개발과 적용을 위한 Application Architecture를 지원하는 데 있어, 데이터의 분산과 관리 접근법을 포함하는 포괄적인 Framework

→ 안정된 데이터 모델의 구축에 활용, 종합시스템 구현을 위한 구체적인 데이터 요건을 정의



# Data Architecture

## ■ Data의 특성

Transaction의 결과로 수집된 Data의 가치는 낮다

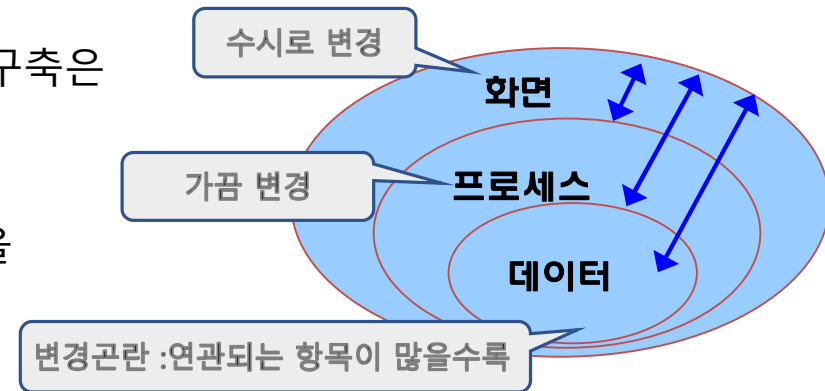
→ 업무 수행(Transaction)의 결과로 발생, 어떤 Data가 필요한지 미리 정해두지 않으면 활용도 저하

☞ Top down으로 필요한 Data를 정하는 방법 : E2E 최적화 Approach

## ■ Data Architecture의 중요성

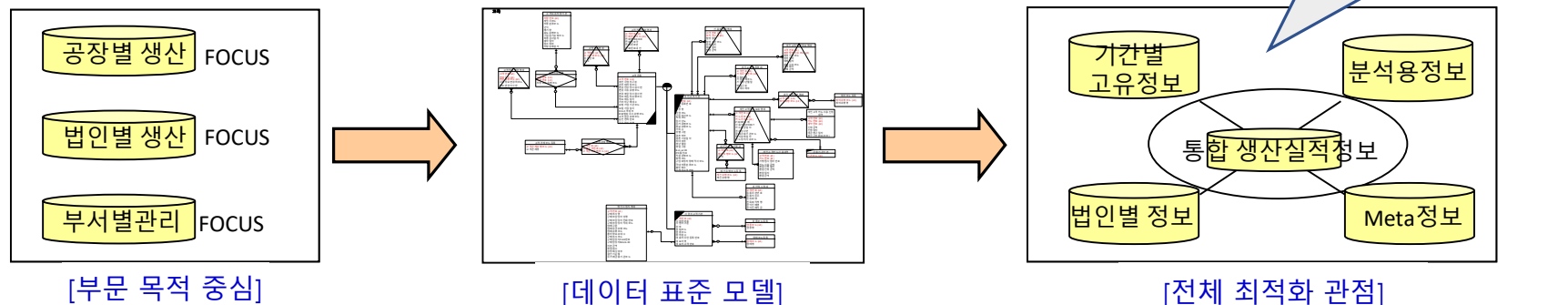
현업 요구사항이 확정되지 않은 상태에서 시스템 구축은 DB구축과 프로그램 개발을 동시에 진행하게 됨

- 수백, 수천의 프로그램이 이 데이터를 이용
- 데이터의 구조 변경은 이 수천 프로그램의 변경을 초래하여 다른 테이블과의 정합성이 깨짐
- 한번 구축된 Data Base는 변경하기 어려움



## ■ 전사 통합 데이터 표준 수립

부문 목적 중심의 데이터에서 전체 최적화 관점의 데이터통합으로 전환



# Data Architecture

---

## ■ Data 요건 및 구현 전략

### - Data 요건

- 통합 DB 구축
  - 자원의 중복 관리, 입력오류, 데이터의 지속적인 관리 부족 등에 의한 결함 요건을 제거하고 업무전반에 걸친 일관된 데이터의 흐름을 유지하기 위해서 통합 DB 구축
  - Groupware(Notes)와의 연계 및 기존 관리 DB와의 연계
  - 관계형 DB로 타 기종간의 Data Connection은 실시간으로 구현
- 데이터 관리 기술
  - 데이터 접근이 원활하고 사용이 편리하면서 데이터의 보안 및 관리가 용이
- 데이터의 사용 편의성
  - 타 기종의 하드웨어에서 운영되는 타 기종의 DB를 통합하여 사용자에게 단일한 DB 이미지를 제공

### - 구현 전략

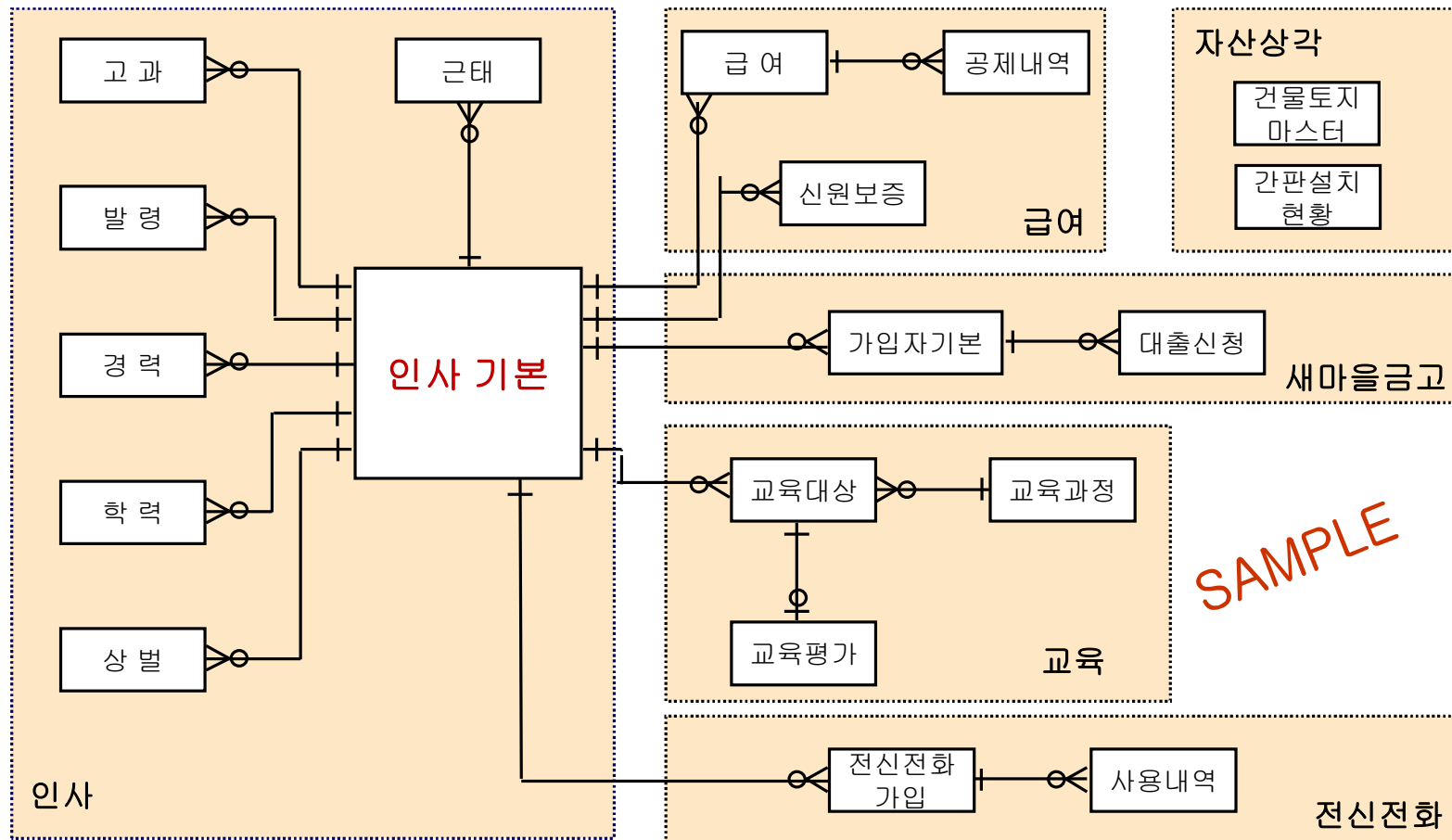
- 구현 목표
  - 개발 생산성 향상, 통합 정보 제공, 빠른 응답속도, 다양한 Ad-hoc Query 제공
- 구현 방향
  - 데이터 구조 변경의 용이성, 분산과 통합의 적절한 조화, 편리한 사용, IT 산업 표준 적용, 각종 Query Tool 활용, 다양한 정보기술과 연계성을 충족

# Data Architecture

## ■ Data Model 수립

업무 기능 영역별로 생성되는 데이터의 관련성을 도식화 하여 데이터의 연계 구조를 수립하는 것으로 Entity Model이라고 하며, ERD(Entity Relationship Diagram)로 표현  
→ 각각의 Entity에 대해 생성되는 정보의 속성을 파악하고 규정

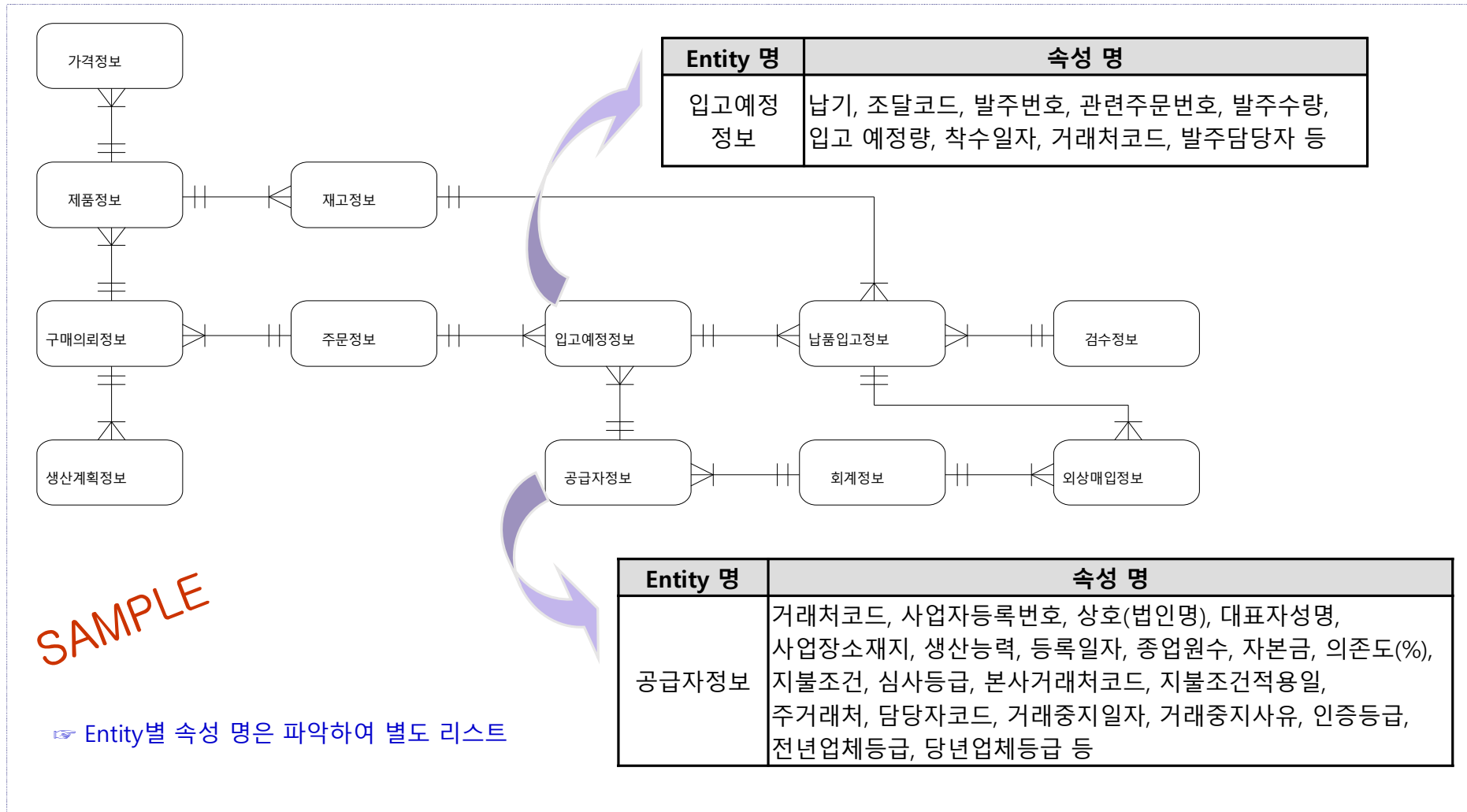
[기능별 Entity Model : 인사 관리]



# Data Architecture

## ■ Data Model 수립(예)

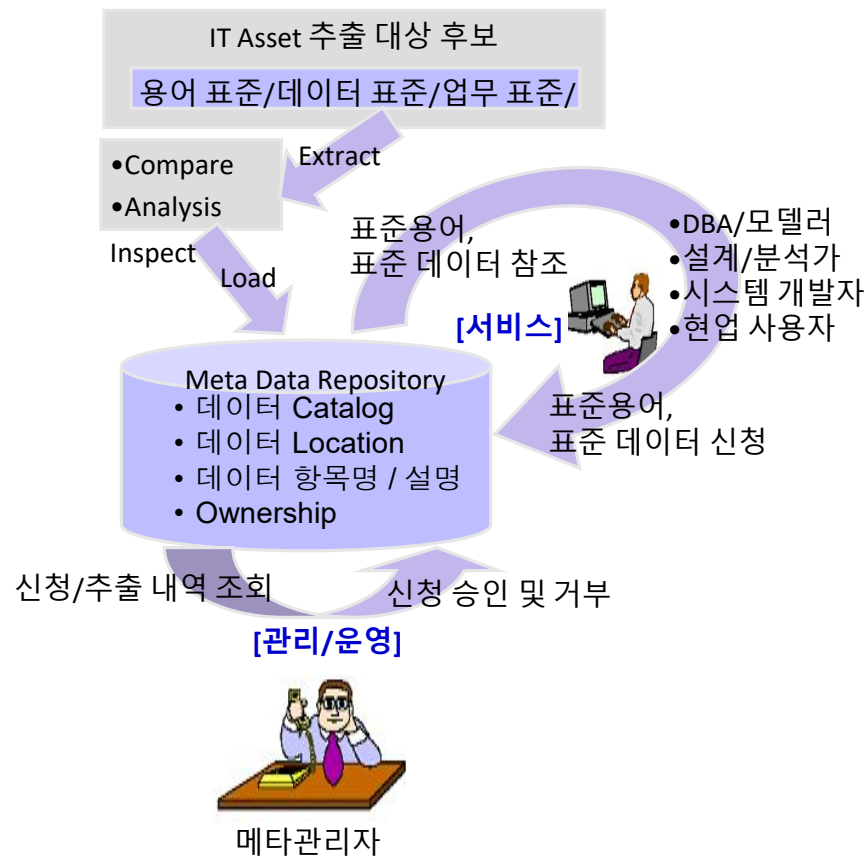
[구매업무 데이터 모델]



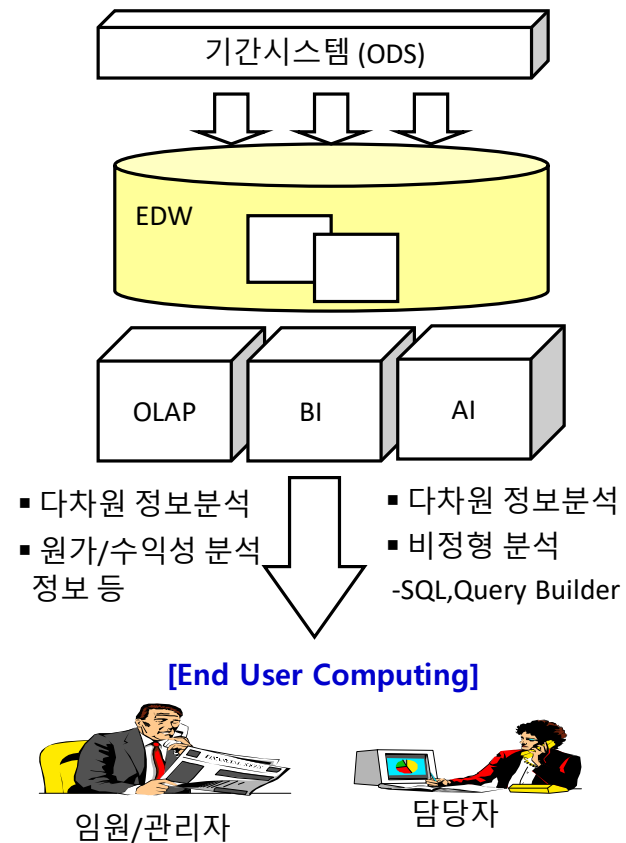
# Data Architecture

## ■ 메타 데이터 관리체계 수립 및 EUC 환경 구축

### - 메타 데이터 관리체계 수립



### - EUC 환경 구축

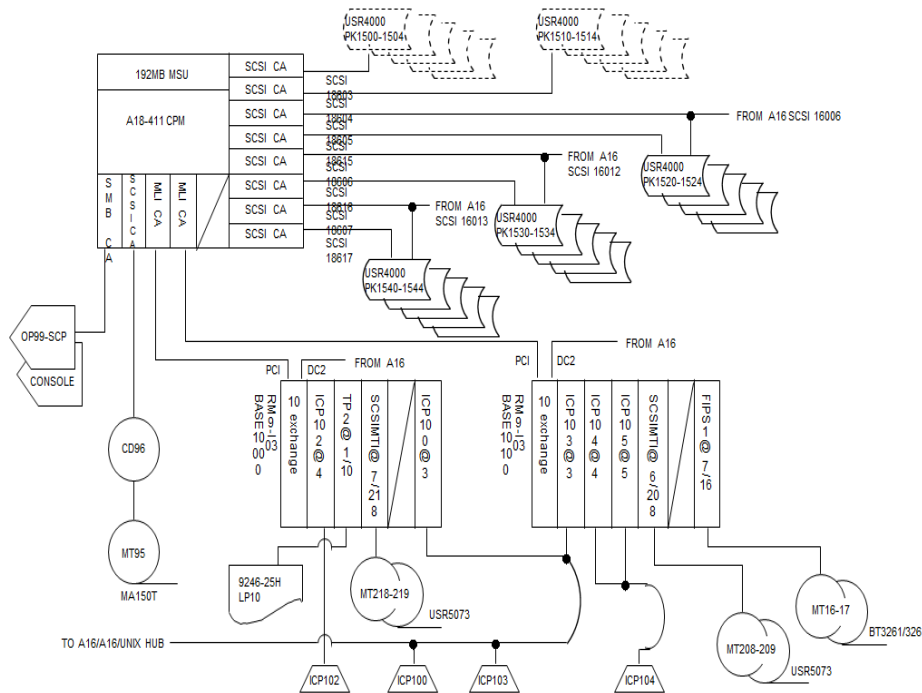


# Technical Architecture

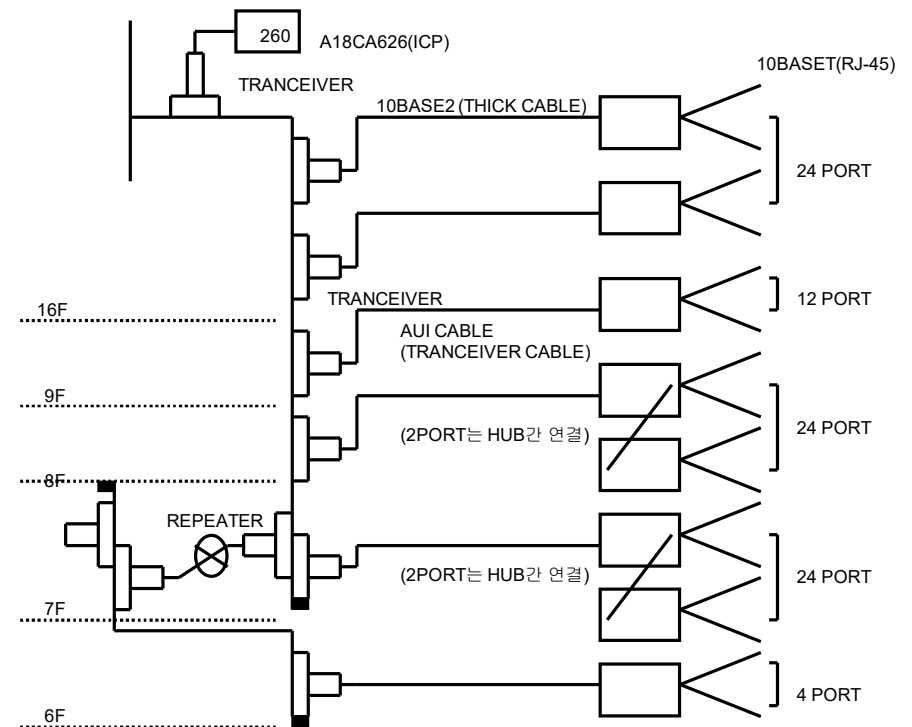
## IT Infra 구성

시스템을 통해 업무 프로세스를 원활히 수행할 수 있도록 H/W, N/W의 구조를 설계하고, 물리적 환경을 구축 (EUC 환경)

[H/W 구성도]

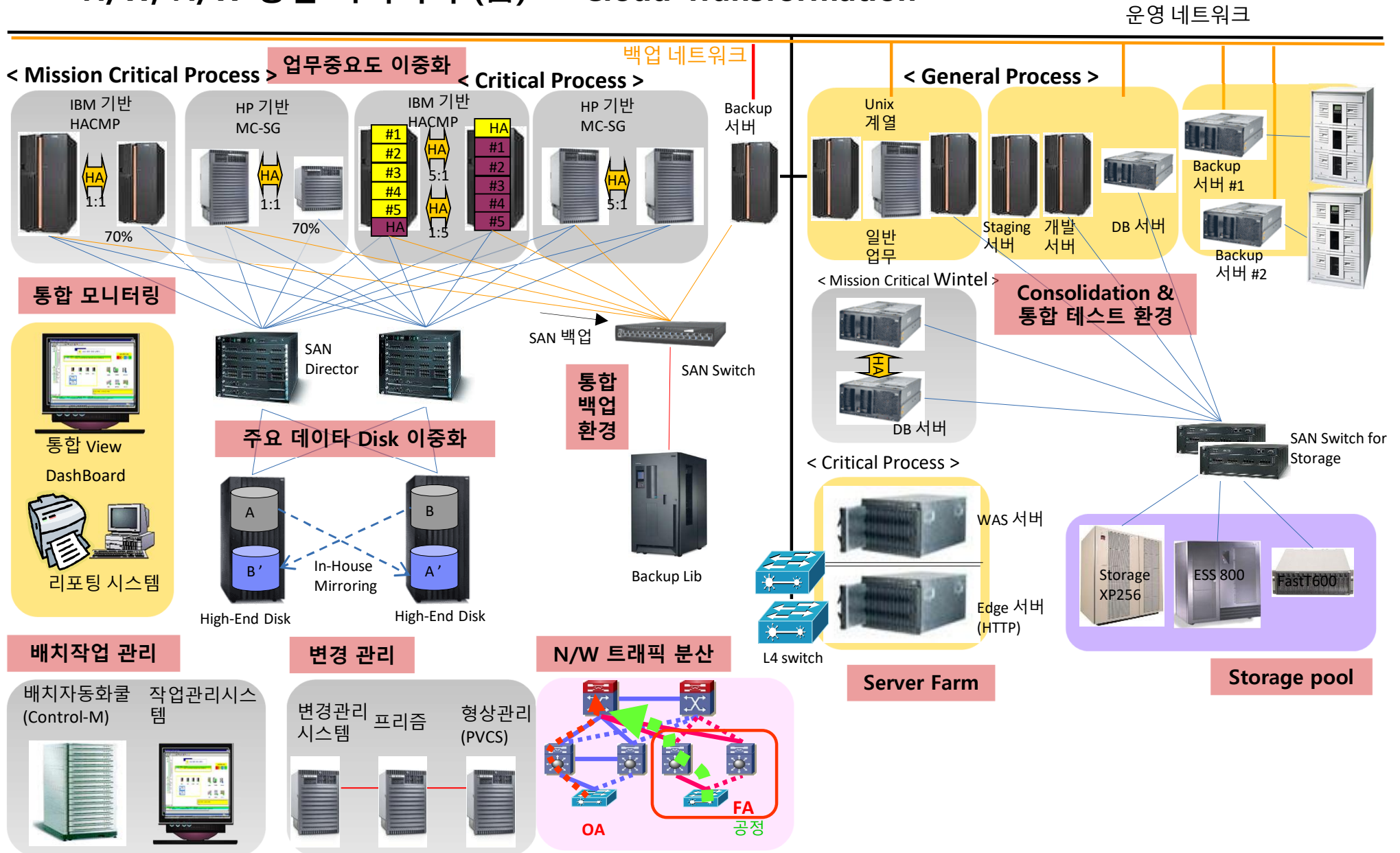


[N/W 구조]



# Technical Architecture

## ■ H/W, N/W 중심 아키텍처 (舊) → Cloud Transformation





# 시스템 설계 구축 검증

---

## ■ 시스템 설계 구축 검증

→ 업무 프로세스 및 정보 운영 시스템을 구축하는 과정이 업무 수행에 맞도록 이루어 지는지 확인하고 승인하는 일 (DRA : D<sup>x</sup> Review & Approval)

## ■ DRA의 목적

→ 시스템 개발 납기/생산성, 시스템 품질, 고객 만족 등 종합 목표를 달성  
→ 예상되는 문제점을 사전에 제거, 관리하여 시스템 활용을 극대화

## ■ DRA 단계

### - DVR (Demand Verification Review) : 사용자 요구 검증

- 시스템 개발자들이 현업의 요구사항에 따라 Prototype(화면)을 기반으로 개발 범위, 시스템 설계 및 적용 기술상의 문제점, 타 시스템과의 Data I/F항목, 전체 일정 등을 검토하여 시스템 개발 의뢰 사항을 구체화(기능 요구사항 정의서)

### - DVA (Design Validation Approval) : 설계 승인

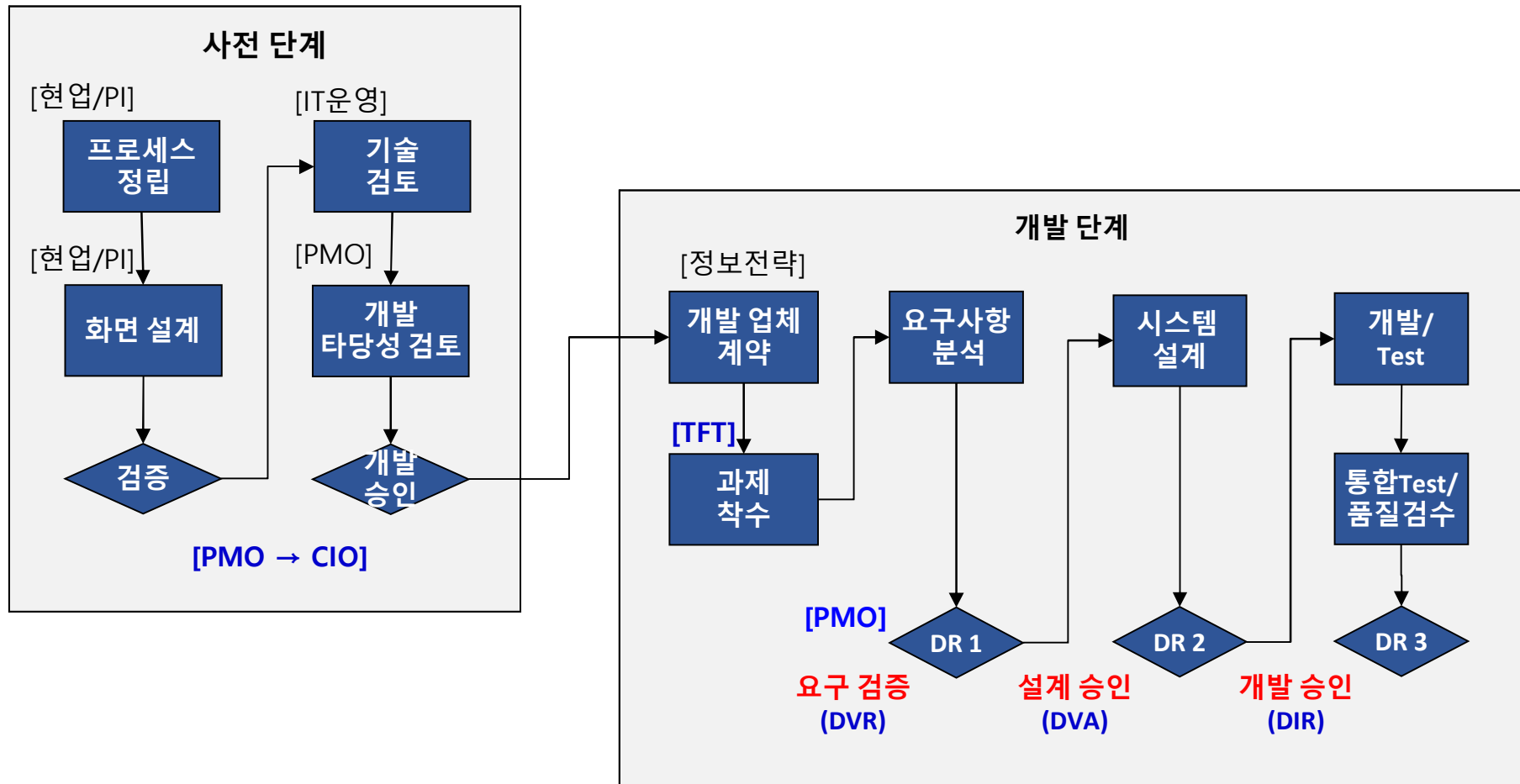
- 시스템 구조(AA/DA/TA)를 설계한 후 그 결과를 상호 조합하여 문제점을 점검하고, 설계 시뮬레이션 결과가 고객 요구를 충족할 수 있는지 확인하여 개발 착수를 승인

### - DIR (Development Implementation Review) : 개발 검증

- 개발 완료된 시스템의 단위/통합/부하 Test 및 품질 검수 결과를 검증하고, 운영으로 이관하기 위한 절차가 적절히 수행되었는지를 검증

# 시스템 설계 구축 검증

## ■ 시스템 설계 구축 검증 프로세스



PMO : Project Management Officer, CIO : Chief Information Officer, TFT : Task Force Team

# DRA 단계별 확인 사항

## ■ DVR (Demand Verification Review) : 사용자 요구 검증

점검 항목	점검 내용	근거자료/ 산출물
1. 사용자 요구사항의 명확한 정의	- 사용자 요구사항이 프로젝트 리더/오너에 의해 승인되었는가? → 총 00건의 요구사항 중 확정 00건, 기각 0, 검토 중(미결) 0건	사전진단서, 현업 인터뷰 결과, 기능 요구사항 정의서
2. 시스템의 개발범위 정의 및 확정	- 개발범위에 대해 프로젝트 리더 및 오너에 의해 승인되었는가? → 개발 내용, 일정, 투입 공수(인력,기간)등의 적정 여부 확인	개발 계획서 (프로그램 목록)
3. To-Be Process 확정 여부	- To-Be Process는 As-Is 대비 변경내용이 명확하게 반영되었나? - To-Be Process는 관련 부서 간에 합의하에 표준화되었나?	To-Be Process 설계서 (Process Map, 화면UI)
4. 타 시스템과의 I/F 항목 정의	- 타 시스템과의 I/F 현황이 명확히 작성되었는가?	I/F 설계서
5. 프로세스 분석단계의 변경사항에 대한 이력 정리	- 프로세스 분석단계에서 주요 변경사항에 대해 이력을 정리, 개발 계획 절차에 따라 승인을 받았는가?	회의록/결재문서
6. DVR 단계의 필수 산출물 점검	- 필수 산출물 작성 여부 → 개발 계획서, 요구사항 정의서, To-Be Process설계서, DVR 보고서	필수 산출물 00개

☞ 상기 점검 항목(체크리스트)에 준하여 점검 내용 확인을 위한 체크시트를 작성, 단계별로 확인

# DRA 단계별 확인 사항

## ■ DVA (Design Validation Approval) : 설계 승인

점검 항목	점검 내용	근거자료/ 산출물
1. 사용자 요구사항의 설계 반영 여부	- 요구사항 추적 표, 화면 Prototype → DVR 반영사항 중 추가, 기각 건 ( %)	기능 요구사항 정의서, 화면 Prototype
2. 시스템 설계 표준 설정 여부	- To-Be Architecture를 설정하였고, 그에 따라 설계 하였나? - 정보 관리 규제 요건의 충족 방법/여부 확인	관리 규제 관련 법, 모의 해킹 등의 근거 및 회신
3. Data Model의 표준 준수도 확인	- ERD(Entity Relationship Diagram) 작성 여부 - 메타 Data 기준 준수 여부 및 문제점	ERD, 프로그램 및 Table 목록/정의서
4. 권한 및 로그관리 체계	- 권한 및 로그관리 등이 관리 가능한 구조로 설계되었나?	
5. IT 표준의 설계 반영 여부	- 00개의 IT 표준이 설정되었고, 설계에 반영된 비율은 얼마인가?	IT 표준 체크시트
6. 이전 단계의 결함 조치 및 향후 일정의 적합성	- DVR 단계의 결함, 지적 사항은 모두 조치되었나? - 향후 개발 시 투입 공수 및 일정에 문제는 없는가?	
7. DVA 단계의 필수 산출물 점검	- 필수 산출물 작성 여부 → Page Layout(화면 장표), ERD, 프로그램 목록, Data/Code 정의서, Table 목록, Table 정의서, 프로그램 사양서, I/F 정의서	필수 산출물 00개

☞ 상기 점검 항목(체크리스트)에 준하여 점검 내용 확인을 위한 체크시트를 작성, 단계별로 확인

# DRA 단계별 확인 사항

## ■ DIR (Development Implementation Review) : 개발 검증

점검 항목	점검 내용	근거자료/ 산출물
1. Test 시나리오의 적합성 검증 및 Test 결과	- Test 시나리오는 전체 업무를 빠짐없이 반영하였나? - 업무 담당자가 통합Test에 참여하여 완료되었는가?	단위/통합/시스템 Test Scenario 및 결과
2. 품질 검수 결함조치 확인	- 통합Test와 품질 검수의 결함은 모두 조치되었는가?	품질 검수 결과서
3. 현업 요구사항 반영 및 이력 관리	- 총 00건의 요구사항 중 몇 건이 개발에 반영되었나? → 반영 00건, 추가 00건, 기각 00건 ( %)	요구사항 추적 표 (통합Test 요구 포함)
4. Data 이관 여부 및 장애 대응 프로세스 점검	- Data이관 계획이 수립되었고 정확하게 실시되었나? - 장애 발생에 대한 대응 프로세스는 수립되었는가? → 조치방법, R&R, 담당자, 보고절차 등	이관 계획서, 장애 대응 프로세스
5. 교육 실시 여부	- 매뉴얼은 작성되었고, 운영/사용자에 대한 교육이 완료되었는가? (총 교육인원 : 00명)	교육 결과, 운영/사용자 매뉴얼
6. 이전 단계의 결함 조치	- DVA 단계의 결함, 지적 사항은 모두 조치되었나?	DVA 결과 보고서, 회의록, 결재문서
7. DIR 단계의 필수 산출물 점검	- 필수 산출물 작성 여부 → 단위/통합/시스템Test 결과서, 이행 계획서, 교육 계획/결과서, 운영/사용자 매뉴얼	DIR 단계 보고서, 개발완료 보고서 등

☞ 상기 점검 항목(체크리스트)에 준하여 점검 내용 확인을 위한 체크시트를 작성, 단계별로 확인