**논리 데이터 모델링**

**제 1절 논리 데이터 모델링 이해**

1. **논리 데이터 모델 정의**

- 논리적 데이터 모델링이란 데이터베이스 설계 프로세스의 input

- 비즈니스 정보의 구조와 규칙을 명확하게 표현하는 기법

- 논리적 모델은 데이터 모델링이 최종적으로 완료된 상태

- 누가 어떻게 데이터에 엑세스하는 가의 전산화와는 독립적

- 데이터 모델링 과정에서 가장 핵심이 되는 부분

- 시스템 구축을 위해서 가장 먼저 시작해야할 기초적 업무 조사를 하는 초기 단계부터 인간이 결정해야할 대부분의 사항을 모두 정의하는 시스템 설계의 전 과정을 지원하는 과정의 도구

**2. 논리 데이터 모델링 목적 및 효과**

- 해당 비즈니스에 대한 데이터 관점에서의 명확한 이해

- 전사적인 통합 데이터 체계 확립

- 데이터의 일관성 및 정확성 유지를 위한 규칙 도출

- 사용자와의 명확한 의사 소통을 위한 수단으로 활용

**3. 논리 데이터 모델링 필수 성공요소**

- 업무에 능통한 현업 사용자와 함계, 절차 보다는 데이터에 초점을 두고, 데이터 구조와 무결성을 함께 고려하며 모델링을 진행.

- 개념화와 정규화 기법을 적용하며 가능하면 다이어그램을 이용하여 업무를 표현하고 데이터 사전을 구축하라.

**제 2절 속성 정의**

**1. 속성의 개념 : 가공되지 않은 것, 원천을 의미**

\* 속성의 특징

- 속성도 일종의 집합이며 릴레이션십도 속성이다.

- 속성들 간은 서로 독립적이다.

**2. 속성 후보 도출**

1) 속성 후보 수집처

- 구 시스템의 문서 자료, 현업 장표 / 보고서, 사용자와 협의, DFD, 전문 서적 및 자료, 다른 시스템 자료.

2) 속성 후보 선정 원칙

- 원시 속성으로 보이는 후보는 버리지 않는다. (추출속성이 아니며 재현할 수 없는 속성)

- 소그룹 별로 후보군을 만들고 가장 근접한 엔터티에 할당

3) 속성의 기본 구성요소

- 속성명, 도메인, 선택성 (not null . null)

- 속성명 : 의미를 명확히 표현하는 명사구. 일반적으로 사용하는 용어. 엔터티명 사용 x. 필요시 표준 약어를 제정하며 단 하나의 실체에만 속하도록 할 것.

- 도메인 : 데이터 타입/ 길이 / 허용 값 / 디폴트 값 및 디폴트 알고리즘

- 선택성 : 필요조건 / 금지조건 / 무관계 조건

**3. 속성 검증 및 확정**

- 최소단위 까지 분할 : 집합 개념의 속성은 단순 개념으로 분할하며 분할 및 통합의 기준은 업무의 요구사항에 따른다.

- 하나의 값만 가지는지 : 반복되는 속성은 잘못된 속성이므로 1차 정규화의 대상이 된다.

- 추출속성인지 검증한다.

- 보다 상세하게 관리할 것인지

**4. 추출속성 규칙**

- 초기에는 중복으로 판단하고 안하는걸 권고했으나 현재는 반드시 기술할 것을 권장

- 도출속성이란 어떤 추가 계산 작업을 수행함으로써 창출되는 속성이다,

- 도출 속성은 경영층이 진실로 원하고 필요로 하는 정보이며 사용자들의 데이터 요구사항을 나타내며 도출속성을 포함시키는 것은 그들의 물리적 구현에 대한 어떤 것을 내포하지 않으며 향후 과정으로써 결코 기본키 역할을 맡아서는 안된다.

**5. 속성 정의 시 유의사항**

- 의미가 명확한 속성 명칭 부여

- 유일한 복합명사 사용

- 단수형으로 속성명을 사용

- 표준단어 제정

- 작의적인 전용 금지

**제 3절 엔터티 상세화**

- 식별자 : 엔터티 내의 모든 인스턴스를 유일하게 구분하기 위해 쓰임

- 본질 식별자 : 키 엔터티에서는 엔터티 모호성을 해소하기 위해, 행위 엔터티에서는 부모가 누구인지 확인하기 위해서 쓰이며 절대종속 (태어나게 하는 관계) / 상대종속 (없어도 됌)으로 나뉜다. 본질식별자 구분 방법은 육하원칙을 이용한다.

- 후보 식별자 : 인스턴스를 유일하게 식별할 수 있어야 하며 나머지 속성들을 직접 식별 할 수 있으며 NULL이 될 수 없다. 개념적으로 유힐해야 하며 자주 변경되지 않는 것.

- 대체식별자 : 사원 엔터티의 주민번호

- 인조식별자 : 최대한 범용적인 값으로, 유일한 값을 만드며, 하나의 인조 식별자 속성으로 대체할 수 없는 형태를 주의하며 편의성, 단순성 확보를 위해 사용할 수 있다. 의미의 체계화를 위해서 사용할 수도 있고, 내부적으로(오라클 구조) 사용할 수도 있다.

**1. 식별자 확정**

1) UID BAR의 두가지 의미 : 식별자, 정보로서의 역할

2) 식별자 확정 절차 : 하향식 방식.

- 키 엔터티 식별자 확정 : 본질 식별자 결정 단계에서 미리 결정

- 메인 엔터티 식별자 확정 : 식별자 속성의 개수를 적게하며 인조식별자 생성하기도 함

- 하위 엔터티 식별자 확정 : 본질 식별자 지정

**2. 정규화**

- 데이터의 일관성과 무결성을 위해 행하는 것

1) 정규화의 장점

- 중복 값이 줄어드며 NULL 값도 줄어드며 복잡한 코드로 보완할 필요가 없으며 새로운 요구사항의 발견 과정을 도우며 업무 규칙의 정밀한 포착, 안정성을 최대화 한다,

2) 정규화 단계

- 1차 정규형 : 모든 속성은 반드시 하나의 값을 가져야 한다. (자식 생성)

- 2차 정규형 : 모든 일반 속성들은 전체 식별자에 종속되어야 한다, (부모 생성)

- 3차 정규형 : 일반 속성들끼리 종속이 있으면 안된다. (참조관계 생성)

- BCNF 정규형 : 모든 결정자는 후보키여야 한다.