



저작자 표시 – CREATIVE COMMONS

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

해당 자료는 송즈캠퍼스 김강민 강사의 자료입니다. (www.songscampus.co.kr)
이 저작물을 복제, 배포, 전송, 할 수 있습니다.

단, 다음과 같은 조건을 따라야 합니다.

저작자 표시 : 귀하는 원저작자를 표시하여야 합니다.

비영리 : 이 저작물을 영리 목적으로 이용할 수 없습니다.

해당 내용은 유튜브에서 무료로 공개한 강의를 통하여 도움을 받을 수 있습니다.

데이터모델링

데이터모델링의 특징

-단순화(Simplification)

누구나 쉽게 이해할 수 있도록 표현한다.

-추상화(Abstraction) <모형화, 가설적>

현실세계를 간략히 표현한다.

-명확화(Clarity)

명확하게 의미가 해석되어야 하고 한 가지의 의미를 가지면 된다.

데이터모델링 단계별 특징

-- 개념 데이터 모델링

가장 높은 수준의 추상화, 포괄적이고 전사적인 모델링

-- 논리 데이터 모델링

정규화, 데이터 표준화를 시행하는 단계, 재사용성이 높다.

-- 물리 데이터 모델링

가장 구체적인 모델링이다. 데이터 개체, 속성, 제약조건 등 전반적으로 정의한다. 성능, 보안성, 가용성 을 고려하여 구축한다.

데이터모델링의 중요성

파급 효과(Leverage)

시스템 구축이 완성되어가는 시점에서 데이터 모델을 변경하게 되면 시스템 구축 프로젝트에 큰 위험요소가 된다.

간결한 표현(Conciseness)

정보 요구 사항과 한계가 정확하고 간결하게 표현되어야 하므로 데이터 모델이 필요하다.

데이터 품질(Data Quality)

데이터 구조의 문제로 인해 데이터 품질의 문제가 발생한다. 예를 들면 중복데이터의 미정의, 데이터 구조의 비즈니스 정의의 불충분, 동일한 성격의 데이터를 분리하여 데이터 불일치 등이 있다.

데이터 품질과 관련된 유의점

중복(Duplication)

데이터 모델은 같은 데이터를 사용하는 사람, 시간, 그리고 장소를 파악하는데 도움을 준다.

데이터베이스가 여러 장소에 같은 정보를 저장하는 잘못을 하지 않도록 한다.

비유연성(Inflexibility)

데이터 모델을 어떻게 설계했느냐에 따라 사소한 업무 변화에 대해서도 데이터 모델이 수시로 변경된다면 유지보수의 어려움을 가중시킬 수 있다.

비일관성(Inconsistency)

데이터의 중복이 없더라도 비일관성은 발생한다. 데이터 모델링시, 데이터와 데이터 간 상호 연관 관계에 대한 명확한 정의가 요구된다.

데이터모델링의 필수요소

- 1) 업무가 관리하고자 하는 복수의 대상(엔터티, Entity)
- 2) 대상들이 갖는 속성 (Attribute)
- 3) 대상들 간의 관계 (Relationship)

엔터티

업무에서 저장되고 관리되어야 하는 데이터들의 집합이다.

엔터티의 특징

식별 여부

엔터티(Entity)는 식별자(Identifier)에 의해서 식별이 가능하다. 엔터티는 유일한 식별자(Unique Identifier)를 필수로 한다.

인스턴스들의 집합

엔터티는 최소 2개 이상의 인스턴스가 존재해야한다.

속성

엔터티는 반드시 속성(변수)를 가지고 있어야 한다. 각각의 엔터티에는 속성을 2개 이상 갖는다. 하나의 인스턴스는 각각의 속성들에 대해 한 개의 속성 값을 갖는다.

업무

엔터티는 업무에서 관리되어야 하는 집합이다.

관계

엔터티는 다른 엔터티와 최소한 한 개 이상의 관계를 가진다. 예외로 통계를 위한 엔터티, 코드를 위한 엔터티, 시스템 처리시 내부적인 필요를 위한 엔터티는 관계가 없을 수 있다.

엔터티 생성시점에 따른 분류

기본 엔터티(Fundamental Entity)

업무에 원래 존재하는 정보이다. 다른 엔터티와의 관계에 의해 생성되지 않고 독립적으로 생성이 가능하고 엔터티의 부모역할을 하게 된다.

중심 엔터티(Main Entity)

기본 엔터티로부터 발생되고 업무에 있어서 중심적인 역할을 한다.

행위 엔터티(Active Entity)

두 개 이상의 부모 엔터티로부터 발생되고 내용이 자주 바뀌거나 데이터 양이 증가된다.

엔터티 형태에 따른 분류

유형 엔터티(Tangible Entity)

물리적 형태가 있는 엔터티이다. 형태가 있으므로 안정적, 지속적인 특징이 있다.

개념 엔터티(Conceptual Entity)

물리적 형태가 없는 엔터티이다. 개념적 정보의 엔터티이다.

사건 엔터티(Event Entity)

업무를 수행함에 따라 발생하는 엔터티이다. 정보의 발생량이 많다는 특징이 있다.

속성

업무상 분석해야 하는 대상의 성질, 특징을 의미한다. 업무상 인스턴스로 관리하고자 하는 더 이상 분리되지 않는 최소의 데이터 단위이다.

속성의 특성에 따른 분류

기본 속성(Basic Attribute)

업무로부터 추출된 모든 속성이다. 엔터티에 가장 일반적이고 많이 존재하는 속성이다.

설계 속성(Designed Attribute)

기본 속성 외, 업무를 규칙화하기 위해 새로

만들어지거나 기존 속성을 변형하여 만들어지는 속성이다.

파생 속성(Derived Attribute)

다른 속성에 영향을 받아 발생하는 속성이다. 보통 계산된 값들이 해당된다.

도메인

각 속성의 가질 수 있는 값의 범위 (데이터 타입, 크기, 제약사항 등)

식별자(Identifiers)

엔터티 내에서 인스턴스들을 구분할 수 있는 구분자이다.

식별자의 대표성 여부에 따른 분류

주 식별자

엔터티 내에서 각 인스턴스들을 구분할 수 있다. 데이터 모델링시 대표성을 지니게 된 식별자

보조 식별자

엔터티 내에서 각 인스턴스들을 구분할 수 있다. 다만, 대표성을 가지지 못하는 식별자 주 식별자들만 타 엔터티에 외부식별자를 생성하여 참조관계를 형성할 수 있다.

식별자의 스스로 생성여부에 따른 분류

내부식별자

엔터티 내부에서 스스로 만들어지는 식별자

외부식별자

타 엔터티와의 관계를 통해 타 엔터티로부터 받아오는 식별자 일반적으로 외부식별자는 물리 데이터 모델링시 외래 키(Foreign Key)로 구성된다.

식별자의 속성의 수에 따른 분류

단일 식별자

하나의 속성으로 구성된 식별자

복합 식별자

둘 이상의 속성으로 구성된 식별자

식별자의 대체 여부에 따른 분류

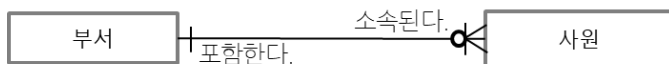
본질 식별자

업무에 의해서 만들어지는 식별자

인조 식별자

업무에 의해서 만들어지지는 않으나, 기존 식별자가 복잡한 구성을 가지고 있어 인위적으로 만들어낸 식별자

관계명



엔터티의 관점에서 관계가 시작되는 부분을 관계 시작점(The Beginning), 관계를 받는 부분을 관계 끝점(The End)라고 부른다

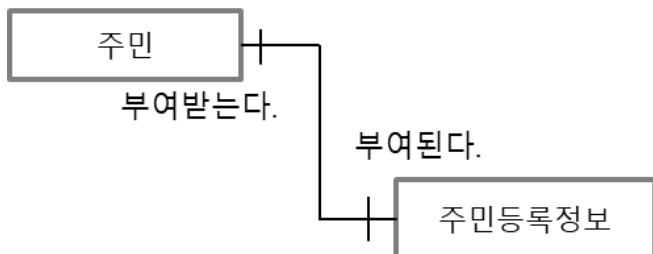
관계명 기술 규칙

애매한 동사를 피해야 한다.

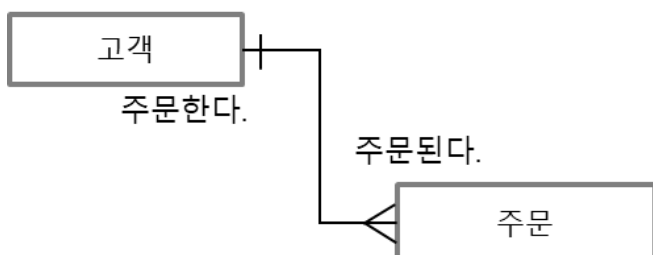
현재형으로 표현해야 한다.

관계차수 (Degree / Cardinality)

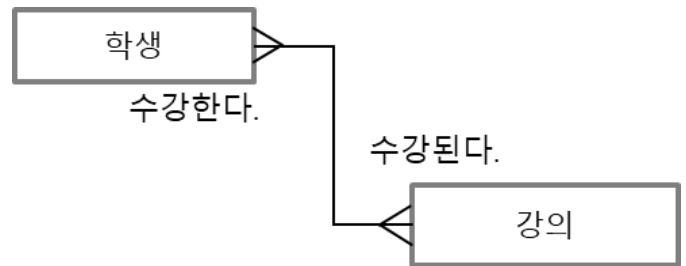
(1) 1:1 관계



(2) 1:M 관계



(3) M:M 관계



관계 정의시 점검할 사항

관계 연결에 대한 규칙이 서술되어 있는지 여부
관계 연결이 '동사(Verb)'로 이루어져있는지 여부
두개의 엔터티 사이에 연관규칙이 존재하는지 여부
엔터티 사이에 정보의 조합이 발생하는지 여부

식별자관계 / 비식별자 관계

식별자 관계(Identifying Relationship)

부모 엔터티의 주식별자(Primary Key)가 자식 엔터티의 주식별자로 상속된 경우이다.

강한관계

과도한 식별자 관계는 SQL 구문을 복잡하게 하고 오류가능성을 증가시킨다.

비식별자 관계(Non-Identifying Relationship)

부모 엔터티의 주식별자(Primary Key)가 자식 엔터티의 일반 속성으로 상속된 경우이다.

약한관계

과도한 비식별자 관계는 불필요한 조인을 유발시킨다.

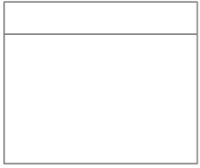
| | 식별자 관계 | 비식별자관계 |
|---------------|---|--|
| 관계의 강도 | 강한 관계 | 약한 관계 |
| 자식 주식별자 포함여부 | 자식의 주식별자에 포함된다. | 자식의 일반 속성에 포함된다. |
| 표기 방법 | 실선 표현 | 점선 표현 |
| 데이터 모델링시 고려사항 | 자식 엔터티는 부모 엔터티에 종속된다. 상속받은 주식별자의 속성을 다른 엔터티에 이전 필요하다. | 자식 주식별자를 별도로 구성한다. 상속받은 주식별자의 속성이 상속되는 것을 차단하기 위해서 고려한다. |
| 단점 | SQL 명령문이 복잡해질 수 있다. 오류 가능성이 커질 수 있다. | 과도한 조인을 유발하게 된다. |

ERD 표기법

(1) 엔티티 표기법

I/E 표기법 예

사원 정보



Barker 표기법 예

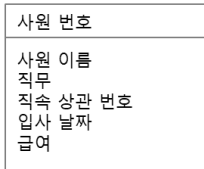
사원 정보



(2) 속성 표기법

I/E 표기법 예

사원 정보



Barker 표기법 예

사원 정보

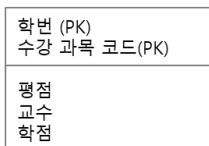
- * 사원번호
- * 사원 이름
- * 직무
- 직속 상관 번호
- 입사 날짜
- 급여

* : not null 조건을 표기한다.

(3) 식별자 표기법

I/E 표기법 예

과목 성적



Barker 표기법 예

사원 정보

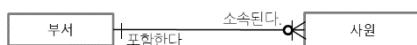
- # 학번
- # 수강 과목 코드
- 평점
- 교수
- 학점

#: 식별자 표기법

(4) 관계 표기법

- 관계명 표기법

I/E 표기법, Barker 표기법



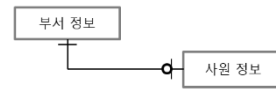
- 관계 차수 표기법

I/E 표기법, Barker 표기법

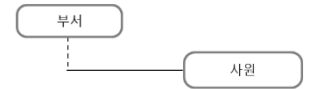


- 관계선택사양 표기법

I/E 표기법



Barker 표기법

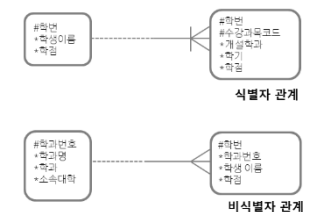


- 식별자 / 비식별자 관계 표기법

I/E 표기법



Barker 표기법



I/E 표기법에서는 관계의 선이 실선이면 식별자 관계, 점선이면 비식별자 관계를 표현한다.

바커 표기법에서는 식별자 관계에서 관계의 끝점(The End)에 UID Bar 수직선을 그어준다. 비식별자 관계에서는 UID Bar가 존재하지 않는다.

ERD 그리는 순서

ERD를 작성하는 방법은 순서가 있다. ① 엔티티를 그린다. ② 엔티티를 적절하게 배치한다. ③ 엔티티 간의 관계를 설정한다. ④ 관계명을 기술한다. ⑤ 관계의 참여도를 기술한다. ⑥ 관계의 필수 여부를 기술한다.

3단계 스키마 구조

외부 스키마 (External Schema)

데이터베이스의 **사용자 관점**이다. 데이터베이스에서 사용자에게 필요한 부분만을 보여주고, 불필요한 부분은 보여주지 않는다. 전체 데이터베이스의 논리적인 부분이므로 서브 스키마(Sub Schema)라고도 한다. 하나의 데이터베이스 시스템에는 **여러 개의 외부 스키마가 존재**한다. 일반 사용자는 SQL 명령문을 이용해 데이터를 얻을 수 있다. 프로그래머는 다른 언어를 사용하여 데이터베이스에 접근할 수 있다.

개념 스키마(Conceptual Schema)

데이터베이스의 **관리자(DBA, 데이터베이스 관리자)**

관점이다. 데이터베이스 전체가 어떻게 구성되어 있고 어떤 데이터가 저장되어 있는지, 데이터들 간의 상호 관계는 어떠한지 알려주는 단계이다. 데이터베이스의 **전체적인 논리적 구조**를 보여준다.

스키마(Schema)라고 하면 개념 스키마를 지칭하는 경우가 많다.

내부 스키마(Internal Schema)

데이터베이스의 시스템 설계자 관점이다. **데이터가 실제로 물리적 데이터베이스에 어떻게 저장되어 있는지** 알려준다.

데이터 독립성의 종류

데이터 독립성

하위 단계의 데이터 구조가 변경되더라도 상위 단계에 영향을 미치지 않는 속성

논리적 독립성

개념 스키마가 변경되어도 외부 스키마에는 영향을 미치지 않는다. 통합 논리적 구조가 변경되어도 응용프로그램에는 이상이 없다.

물리적 독립성

내부 스키마가 변경되어도 외부/개념 스키마에는 영향을 미치지 않는다. 저장장치의 구조변경은 응용프로그램과 개념스키마에 영향이 없다.

이상현상

삽입이상

| 고객(PK) | 고객명 | 상품번호(PK) | 상품명 | 가격 |
|--------|-----|----------|--------|--------|
| 001 | 유비 | 1000 | 스마트폰 | 100000 |
| 002 | 손권 | 1000 | 스마트폰 | 100000 |
| 003 | 관우 | 2000 | 노트북 | 5000 |
| 004 | 장비 | 3000 | LAN 카드 | 500000 |
| 005 | 손오공 | | | |

행 정보(Row) 삽입 시 지정되지 않은 속성 값이 NULL을 가지는 경우를 의미

갱신이상

| 고객(PK) | 고객명 | 상품번호(PK) | 상품명 | 가격 |
|--------|-----|----------|--------|--------|
| 001 | 유비 | 1000 | 스마트폰 | 200000 |
| 002 | 손권 | 1000 | 스마트폰 | 100000 |
| 003 | 관우 | 2000 | 노트북 | 5000 |
| 004 | 장비 | 3000 | LAN 카드 | 500000 |

행 정보(Row) 갱신시 일관성이 유지되지 않는 경우

삭제이상

| 고객(PK) | 고객명 | 상품번호(PK) | 상품명 | 가격 |
|--------|-----|----------|--------|--------|
| 001 | 유비 | 1000 | 스마트폰 | 100000 |
| 002 | 손권 | 1000 | 스마트폰 | 100000 |
| 003 | 관우 | 2000 | 노트북 | 5000 |
| 004 | 장비 | 3000 | LAN 카드 | 500000 |

행 정보(Row) 삭제 시 연쇄 삭제가 발생하는 현상

정규화

제1 정규화

테이블의 모든 속성은 반드시 하나의 값을 가져야 하고, 테이블의 행(Row)들은 서로 간에 식별 가능해야 한다. **테이블 속성의 원자성을 확보하고 주식별자를 설정**

제2 정규화

제 2 정규화는 주식별자가 2개 이상의 속성으로 이루어진 경우 부분 함수적 종속을 제거하는 과정

제3 정규화

엔터티의 일반 속성들 간에는 서로 종속적이지 않는다. 제3정규화는 **주식별자를 제외한 컬럼 간의 종속성인 이행 함수 종속성을 제거하는 과정**



저작자 표시 – CREATIVE COMMONS

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

해당 자료는 송즈캠퍼스 김강민 강사의 자료입니다. (www.songscampus.co.kr)

이 저작물을 복제, 배포, 전송, 할 수 있습니다.

단, 다음과 같은 조건을 따라야 합니다.

저작자 표시 : 귀하는 원저작자를 표시하여야 합니다.

비영리 : 이 저작물을 영리 목적으로 이용할 수 없습니다.

해당 내용은 유튜브에서 무료로 공개한 강의를 통하여 도움을 받을 수 있습니다.