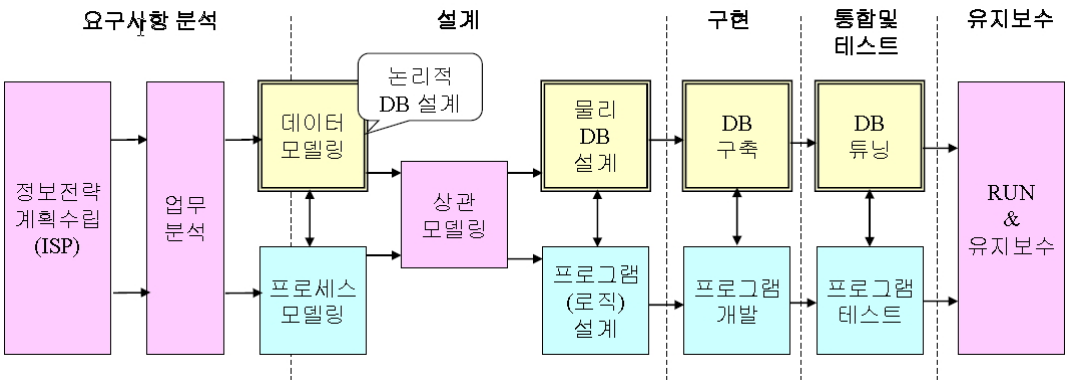
IT&Value Ltd.

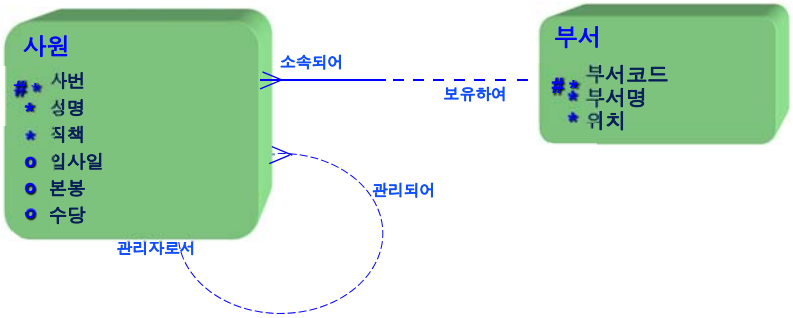
정보시스템 구축과 DB 설계



IT&Value Ltd.

데이터 모델링이란?

- 현실 세계를 데이터 관점에서 파악하여 그 관계를 정의하고 형상화하는 것



Entity-Relationship Data Model은 필요한 정보와 업무의 기능을 충분히 제공할 수 있도록 설계되어야 한다.

IT&Value Ltd.

설계 방법

Function Modeling 후 Data Modeling

- 현실적으로 가장 많이 사용되는 형태이고, 접근하기 쉽다.
- 기능이 변하면 데이터 모델도 변경된다.

Function Modeling과 Data Modeling 동시 진행

- 대형 프로젝트에서 다수의 개발팀이 업무별로 각각 별도로 진행
- 다수의 전문가 필요하며, 그렇지 않으면 서로 어긋날 수 있다.

Data Modeling 후 Function Modeling

- 전문 모델러 확보 시 가능하며, 객관적, 구체적, 데이터의 흐름이 없이 설계 가능
- 기능이 변경돼도 데이터 모델은 잘 변하지 않는다. (체계적 개발 가능)

IT&Value Ltd.

Logical Modeling

- 현실세계를 데이터 관점에서 파악하여 ERD로 표현하는 단계
  - ENTITY: 관리할 대상이 되는 것(실체)
  - RELATIONSHIP: ENTITY 간의 대응관계
  - ATTRIBUTE: 관리할 정보의 구체적 항목
- 효과적인 정보의 문서화나 수집이 가능
- Logical Modeling 시 요구사항을 충분히 수집하지 않으면 다음 단계의 요구사항 변경에 따른 많은 비용이 발생
- Logical Modeling은 H/W나 S/W에 독립적 (HDB, NDB, RDB)

IT&Value Ltd.

Entity

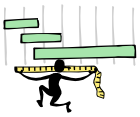
- 우리가 관리하고자 하는 … (주어가 분명해야 함)
- 영속적으로 존재하는 것
- 객관적인 집합 (추상적이어서는 안됨)
- 속성(Attribute)과 행(Instance)가 있어야 함

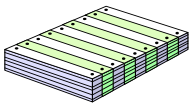
분류 항목	구체적인 예
사람	사원, 계약자, 이용자
물건	재료, 상품, 시설, 지점
사건	계약, 작업, 사고
장소	구획, 지역, 하천, 항만

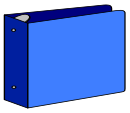
IT&Value Ltd.

Entity 수집

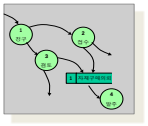
- Entity가 될 가능성이 있는 모든 대상을 수집
- 너무 깊게 생각하지 말고 Entity 자격 유무(有無)로만 단순히 판단
- 비슷한 동의어가 있더라도 함부로 버리지 말 것
- 개념이 모호한 대상은 업무 전문가와 Interview하여 1차로 개념만 파악
- 프로세스(처리과정)에 너무 연연하지 말 것
- 예외 경우(Exception Case)에 너무 집착하지 말 것
- 단어 하나 하나에 집중하여 판단할 것 (상식적이라고 쉽게 생각하지 말라)
- 일단 Entity 대상으로 선정하였다면 그 핵심적인 특징을 파악해 둘 것
- 사용자에게 의존하지 말고 자신의 이해를 바탕으로 할 것
- 구현할 시스템 업무의 본질을 항상 염두에 둘 것

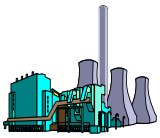

향후 전략


업무 장표


문서


인터뷰


DFD


현장 조사

IT&Value Ltd.

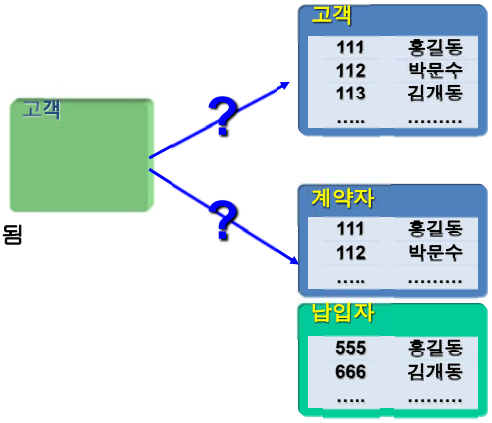
Entity 분류

구 분	정의	예제
Key Entity	• 태초부터 창조된 실체 • 잘 변하지 않는 것 • 데이터를 발생시키는 주체 • 자신의 부모를 갖지 않음	부서, 사원, 거래처, 자재, 고객
Main Entity	• 부모로부터 태어난 실체지만 업무의 중심이 되는 실체 • 많은 자손을 가짐 • 데이터를 발생시키는 주체	카드, 공사, 계약
Action Entity	• 실제 발생하는 업무 • 자주 변경되고 지속적으로 증가 • 반드시 부모를 가짐 • 많은 부모를 가짐	카드이용, 공사내역, 계약변경

Entity 선정

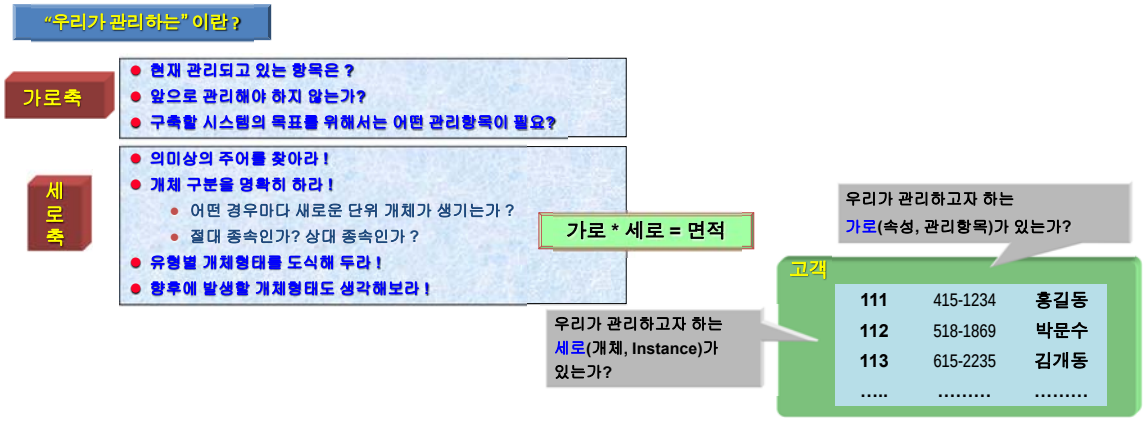
IT&Value Ltd.

- 매우 일반적인 용어로 정의하더라도 오류는 발생할 수 있음
- 반드시 어떠한 요소들이 포함되는지를 명확하게 정의
- Entity가 명확하지 않으면 추후 많은 혼란 발생
  - 관계를 맺고 있는 다른 Entity와 관계형태의 혼란
  - 배타적 논리합 관계 다수 발생
  - 부모를 가지지 않는 데이터 발생
  - Entity간의 부정확한 관계, 매우 복잡한 관계로 나타나게 됨
  - 추후 애플리케이션에서 복잡한 IF 처리를 해야 함



ENTITY의 자격 검증

IT&Value Ltd.

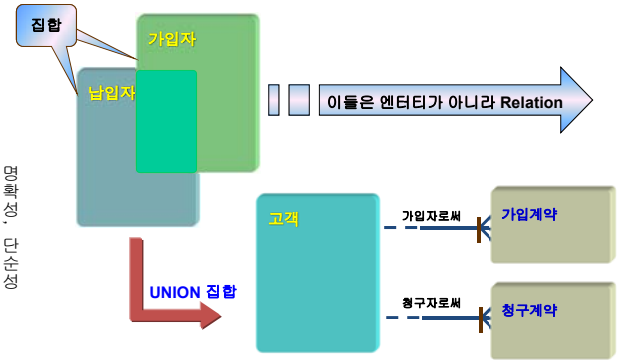


ENTITY의 자격 검증

IT&Value Ltd.

가로 * 세로 = 면적 = 집합

- 그렇다면 집합은 모두 다 엔터티인가?

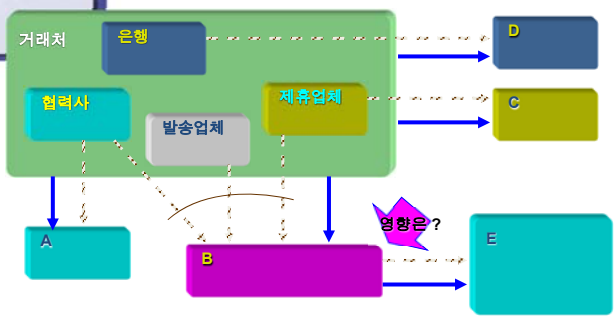


- 홍길동이가 가입자도 되고 청구자도 된다고 홍길동이는 두 명인가?

KEY ENTITY의 단순화

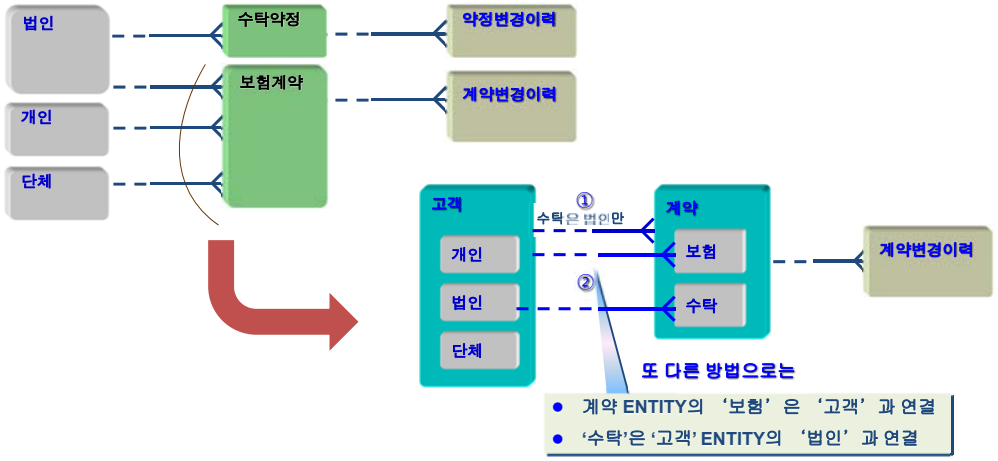
IT&Value Ltd.

- 부모들이 분산되어 있으면 자손들은 매우 복잡해 진다.
- 상위 Entity는 하위 Entity에 지대한 영향을 미친다.
- 상위로 갈수록 많은 하위에 영향을 미친다.
- 우리는 인수분해, 통분을 해야 한다. (SUBTYPE 지정)
  - ◆어떤 공통인수로 인수분해 할 것인가?
  - ◆어디까지 통분해야 하는가?
- 결정의 키 포인트는 ?



IT&Value Ltd.

KEY ENTITY의 단순화



IT&Value Ltd.

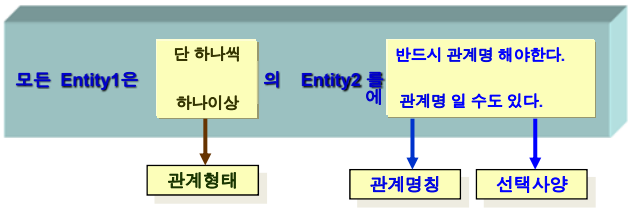
Entity 작성

- ENTITY NAME은 단수형이고 유일하게 부여하고, 이름만으로도 의미 전달이 되도록 정의
- 보충 설명이 필요한 경우나 지금까지 통상 사용하던 단어가 있다면 동의어를 ()사용하여 추가
- ENTITY 명을 ATTRIBUTE 명과 같게 사용하지 말 것
- 모든 ENTITY는 다수의 사용형태(INSTANCE)를 가져야 함
- 모든 ENTITY의 사용형태는 ATTRIBUTE에 특정한 값을 가져야 함
- 모든 INSTANCE는 같은 ENTITY 내에서 반드시 다른 INSTANCE와 구별가능한 식별자(UID)를 가져야 함
- 식별자는 현존하지 않더라도 개념적으로는 존재해야 함
- 만약, 유일한 식별자가 없다면 ENTITY가 아님
- ENTITY NAME은 대문자, 크게 표시
- ATTRIBUTE NAME은 소문자, 작게 표시
- ENTITY의 UID가 되는 ATTRIBUTE에는 #*을 표시

IT&Value Ltd.

Relationship

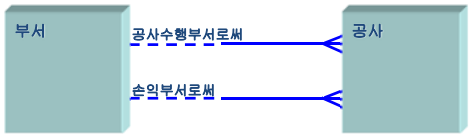
- 두 개의 Entity나 그 자신과의 특정관계를 양방향으로 표현
- 현재의 관계나 장래 유용한 관계만 한정적으로 표시
- 관계명은 구체적이어야 한다. (속하여, 참조하여 등은 피할 것)
- 주는 쪽(One) : 특정 집합만 관계를 가질 때는 해당 집합을 표현하는 것이 좋음
- 받는 쪽(One or Many) : 최대한 관계 내용을 구체적으로 표현할 것
- 보편타당성을 유지 (특별한 설명이 없더라도 이해할 수 있는 일반적이고 객관적인 용어 사용)



IT&Value Ltd.

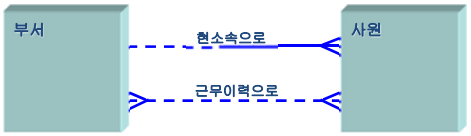
Relationship

역할위주로 정의하는 경우



- 주는 쪽(One)의 특정 집합이 서브타입으로 지정되어 있지 않은 경우 관계를 가지는 주체들을 구체적으로 명시할 때 사용
- 이러한 관계정의가 명확히 되어 있지 않으면 관계가 모호해짐

내용 위주로 정의하는 경우

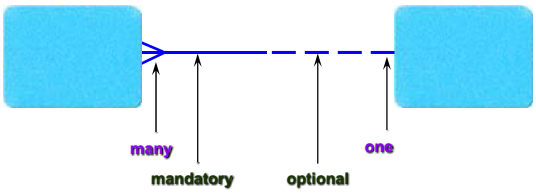


- 관계를 가지는 집합은 동일하나 관계를 맺는 내용이 다를 때 사용
- 주는 쪽의 집합이 서브타입이나 구분값을 가지지 않는 경우에 사용
- 즉, 관계를 맺는 집합이 명확하지 않을 때 적용함

IT&Value Ltd.

Relationship의 표현

1. 두 ENTITY 사이에 선을 그린다.
2. 관계명칭을 기록한다.
3. 선택사양(Optionality)를 표시한다.
   - ≻— : 하나이상(one or more)
   - —— : 단하나씩(one and only one)
4. 관계형태(Degree)를 표시한다.
   - - - - : Optional(maybe)
   - —— : Mandatory(must be)



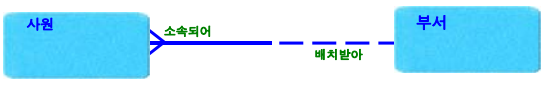
IT&Value Ltd.

관계형태(Relationship type)

MANY to ONE (M : 1) 관계 :
MANY to MANY (M : M) 관계 :
ONE to ONE (1 : 1) 관계 :

M : 1 관계

- 한쪽 방향은 하나이상(one or more)
- 다른 방향은 단 하나씩(one and only one)
- 가장 일반적인 형태
- 보통 Must be 와 May be 로 지정되나 드물게는 양방향 Must be 로도 지정된다



각 사원은 단하나씩의 부서에 반드시 소속되어야 한다.
각 부서는 하나이상의 사원을 배치받을 수도 있다.

IT&Value Ltd.

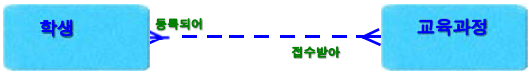
관계형태(Relationship type)

M : M 관계

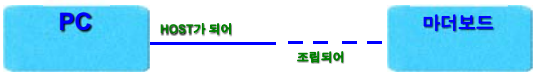
- 양쪽 방향 모두 하나이상(one or more)
- 자주 발생되는 형태
- 상세개념모델(Advanced Conceptual Data Modeling) 단계에서 분할된다

1 : 1 관계

- 양쪽 방향 모두 단 하나씩(one and only one)
- 드물게 발생되는 형태
- 양방향 모두 반드시(must be) 가 되는 경우는 아주 드물다
- 1 : 1 관계는 실제로는 동일한 ENTITY 일 경우가 많다
- 1 : 1 관계가 많이 나타난다면 Entity 가 명확하게 정의되지 않았음을 의미함



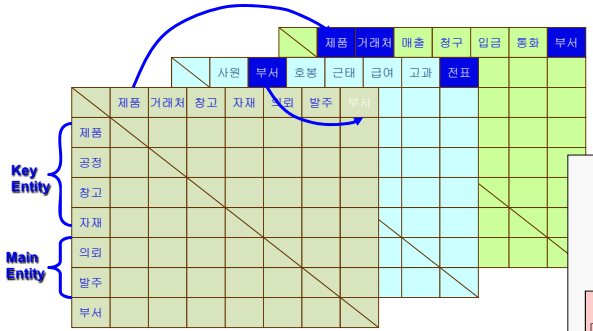
각 학생은 하나이상의 교육과정에 등록되어 질 수도 있다
각 교육과정은 하나이상의 학생을 접수 받을 수도 있다.



각 PC는 단 하나씩의 마더보드에 반드시 HOST가 되어야 한다
각 마더보드는 단 하나씩의 PC에게 조립되어 질 수도 있다

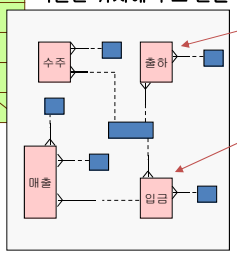
IT&Value Ltd.

MATRIX를 이용하여 ERD 작성



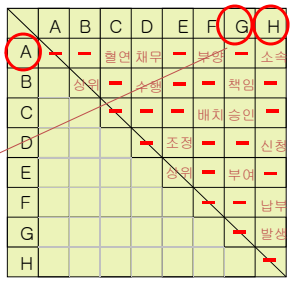
- 그룹별로 분류
- 가능한 한 그룹이 20~30개 이상이 되지 않도록 할 것
- 통상 서브시스템 단위로 분류함
- 매우 큰 시스템에서는 주요 업무별로 분류
- 분류된 그룹별로 타 그룹과 연관이 있는 Entity를 보충

부모가 되는 KEY 엔터티를
적절한 위치에 두고 연결



Relation을 연결하면서 1자
입장에서 관계를 검증하여
확실하게 정의

활동이 많은 엔터티를 업무의 흐름에
맞추어 적절히 배치 (주요 MAIN 엔터티)



IT&Value Ltd.

Relationship

관계형태 결정

- 하나이상 발생되는 경우를 찾아보고 한가지라도 있으면 ONE or MORE (1 : M)
- 매우 적은 가능성때문에 M : M관계가 되는 경우는 1 : M관계로 하기 위한 업무규칙의 재정의를 검토
- 경우에 따라 발생횟수를 조사
- 기존 시스템의 DATA 참조
- 양쪽 방향 모두 조사

선택사양 결정

- 일반적이고 상식적인 선에서 먼저 판단
- 항상 그 관계를 만족해야만 하는지 파악
- 관계가 만족되지 않는 경우를 찾아보고 하나라도 만족되지 않는 경우가 있으면 MAY BE
- 양쪽 방향 모두 조사
- 관계형태에 따른 유형과 대비해 볼 것

관계의 검증

- 업무의 일반적인 규칙과 대비
- 하위 Entity 추가시 상위 Entity 관계 재조명
- 기존 시스템과 비교.분석

IT&Value Ltd.

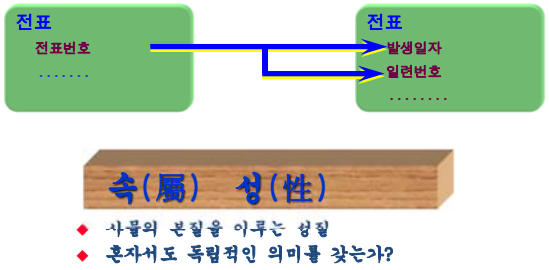
Attribute

- Entity 내에서 관리하고자 하는 정보들의 항목
  - 자격을 부여
  - 식별자
  - 분류를 위해
  - 양의 계수화
  - 상태, 추이의 관리
- Attribute 명칭은
  - 의미가 명확하고 내용을 함축성 있게
  - 길어도 좋으나 의미에 충실
  - Entity 명을 사용하지 말 것
  - 가능한 복합명서 (일자->판매일자)
  - 단 하나의 Entity 에만 속하도록

ATTRIBUTE 지정원칙 (4단계)

IT&Value Ltd.

1단계 : 최소단위까지 분할

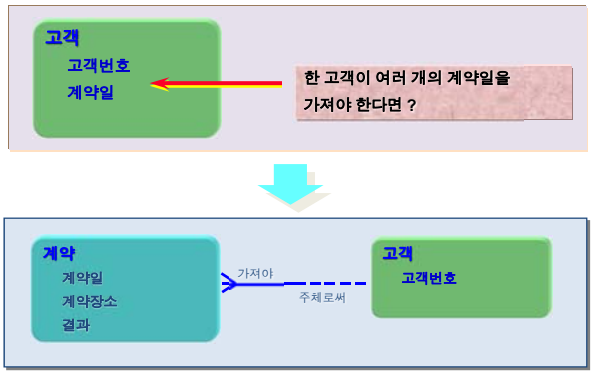


- 집합개념의 ATTRIBUTE는 단순개념으로 분할
- 가능한 최소 단위까지 분할한 후 관리 필요에 따라 통합
- 일자,시간,성명,주민등록번호,우편번호 등은 일반적으로 분할하지 않는 것이 좋다
- 주소와 같은 것은 그냥 두었다가 설계단계에서 필요 시 분할하기도 한다
- 분할 및 통합의 기준은 업무의 요구사항에 따른다

ATTRIBUTE 지정원칙 (4단계)

IT&Value Ltd.

2단계 : 하나의 값만 가지는가 ?



- **여러 값을 가지거나 반복되는 ATTRIBUTE 는 잘못됨**
- **반복되는 경우 새로운 ENTITY로 분할**
- **속성 의미에 대한 정의에 따라 하나일 수도, 여러 개 일 수도 있음**
- **배타적 관계는 가능한 하나의 속성으로 통합할 것**

IT&Value Ltd.

ATTRIBUTE 지정원칙 (4단계)

3단계 : 추출값(derived data) 검증

일반적인 예

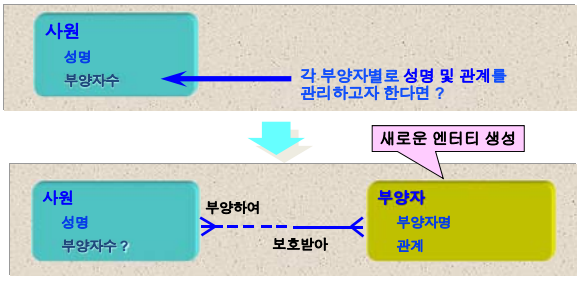
- 개수(COUNT) : 특정범위의 개수, ...
- 합계(TOTAL) : 특정 기간의 총 매출액, ...
- 최대 / 최소 / 평균(MAX / MIN / AVG) : 통계정보
- 기타 계산 : 급여의 10%, 금액(=단가*수량)

- 이력관리를 어떻게 하느냐에 따라 추출 값이 아닐 수도 있음
- 판단이 애매한 경우도 많이 있음
- E-R MODEL 내에는 추출 값을 포함시키지 말 것
- 추후 DATABASE DESIGN 시에 검토하라 !
- 추출 값은 낭비(redundance)
- 추출 값은 데이터의 일관성을 저해
- 추출 값의 기본이 되는 ATTRIBUTE가 변경되면 같이 변경시킨다.
- 꼭 필요한 컬럼은 별도로 정리해 두거나 특별한 표시를 해 둘 것

IT&Value Ltd.

ATTRIBUTE 지정원칙 (4단계)

4단계 : 관리수준 상세화 검토



- ATTRIBUTE가 자신 소유의 ATTRIBUTE를 가지면 ENTITY
- 현재에 만족하지 말고 미래의 관리수준을 감안하라!
- 이 부분을 간과하면 개발 및 테스트, 운영 시에 많은 보완이 발생
- 한번 더 깊이 생각하지 않으면 우리 눈에 보이지 않는다.
- 모델링 시에 좀더 깊이 파 헤친 것이 시간 및 비용 절약의 원천

IT&Value Ltd.

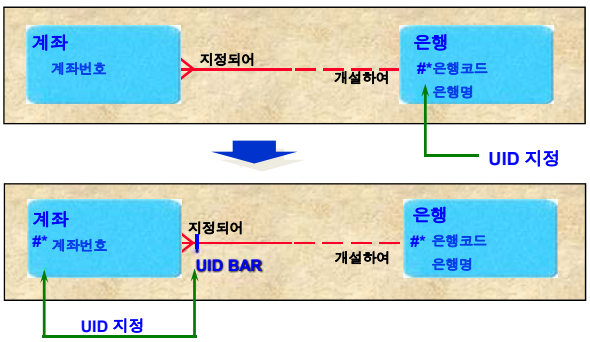
UID(식별자) 지정

- UID: Unique Identifier
- 하나, 혹은 하나 이상의 Attribute로 구성
- 모든 Entity는 반드시 UID를 갖는다.
- UID를 갖지 못하면 Entity가 아니다.
- UID를 구성하는 모든 Attribute는 반드시 존재해야 한다.
- 전략적인 판단이 요구됨

자신의 ATTRIBUTE 가 아니면서 RELATION을 위해 존재하게 되는 ATTRIBUTE를 **자신의 ATTRIBUTE로 표시해서는 안된다**

IT&Value Ltd.

RELATIONSHIP을 통한 UID 지정



UID BAR는 UID에 RELATIONSHIP 이 포함되어 있다는 의미
(즉, 계좌 ENTITY UID = 은행코드 + 계좌번호)

UID는 Mandatory 여야 하므로 UID BAR는 항상
Mandatory + one and only one(——)관계여야 함

IT&Value Ltd.

UID 지정 절차

ATTRIBUTE 검토

- ENTITY를 식별할 Mandatory Attribute 가 있는가 ?
- 과연 식별자인가 ?
- 어떤 결합이 식별자가 될 수 있는가 ?
- 결합된 Attribute 중에도 없으면 인공 Attribute 고려

Relationship 고려

- Relationship 이 누락되지는 않았는가 ?
- ENTITY를 식별할 Relatioship 이 있는가 ?
- Relatioship = Mandatory + One and only One ?

UID 검증

- Sample DATA를 작성해 보라.
- UID를 구성하는 Attribute 와 Relationship 이 MANDATORY 인가 ?

IT&Value Ltd.

정규화 (Noramlization)

- 어떤 대상을 일정한 규칙이나 기준에 따르는 '정규적인' 상태로 바꾸거나, 비정상적인 대상을 정상적으로 되돌리는 과정
- 데이터를 일정한 규칙에 따라 변형하여 이용하기 쉽게 만드는 일

1정규화
- Repeating group 제거

2정규화
- 복합UID에 대한 부분 종속 제거

3정규화
- Non-UID에 대한 종속 제거

IT&Value Ltd.

정규화 예제- 정규화되지 않은 상태

주문번호	주문날짜	고객번호	고객이름	주문상태	상품번호	상품범주	구매수량	상품가격
100	18/05/21	1500	이지우	배송중	2002	전자제품	1	150000
100	18/05/21	1500	이지우	배송중	4092	주방용품	1	24000
100	18/05/21	1500	이지우	배송중	3923	가구	1	350000
101	18/05/21	2100	김보미	결제대기	2002	전자제품	1	150000
102	18/05/21	3500	김서하	구매확정	2002	전자제품	1	150000
102	18/05/21	3500	김서하	구매확정	4092	주방용품	1	24000
102	18/05/21	3500	김서하	구매확정	5100	서적	2	30000

IT&Value Ltd.

정규화 예제- 1정규화 (Repeating group 제거)

주문일반

주문번호	주문날짜	고객번호	고객이름	주문상태
100	18/05/21	1500	이지우	배송중
100	18/05/21	1500	이지우	배송중
100	18/05/21	1500	이지우	배송중
101	18/05/21	2100	김보미	결제대기
102	18/05/21	3500	김서하	구매확정
102	18/05/21	3500	김서하	구매확정
102	18/05/21	3500	김서하	구매확정

주문상세

주문번호	상품번호	상품범주	구매수량	상품가격
100	2002	전자제품	1	150000
100	4092	주방용품	1	24000
100	3923	가구	1	350000
101	2002	전자제품	1	150000
102	2002	전자제품	1	150000
102	4092	주방용품	1	24000
102	5100	서적	2	30000

IT&Value Ltd.

정규화 예제- 1정규화

주문일반

주문번호	주문날짜	고객번호	고객이름	주문상태
100	18/05/21	1500	이지우	배송중
101	18/05/21	2100	김보미	결제대기
102	18/05/21	3500	김서하	구매확정

제1정규형

주문상세

주문번호	상품번호	상품범주	구매수량	상품가격
100	2002	전자제품	1	150000
100	4092	주방용품	1	24000
100	3923	가구	1	350000
101	2002	전자제품	1	150000
102	2002	전자제품	1	150000
102	4092	주방용품	1	24000
102	5100	서적	2	30000

제1정규형

IT&Value Ltd.

정규화 예제- 2정규화 (복합UID에 대한 부분 종속 제거)

주문일반

주문번호	주문날짜	고객번호	고객이름	주문상태
100	18/05/21	1500	이지우	배송중
101	18/05/21	2100	김보미	결제대기
102	18/05/21	3500	김서하	구매확정

제1정규형, 제2정규형

주문상세

주문번호	상품번호	구매수량
100	2002	1
100	4092	1
100	3923	1
101	2002	1
102	2002	1
102	4092	1
102	5100	2

제1정규형, 제2정규형

상품

상품번호	상품범주	상품가격
2002	전자제품	150000
4092	주방용품	24000
3923	가구	350000
2002	전자제품	150000
2002	전자제품	150000
4092	주방용품	24000
5100	서적	30000

정규화 예제- 2정규화 (복합UID에 대한 부분 종속 제거)

IT&Value Ltd.

주문일반

주문번호	주문날짜	고객번호	고객이름	주문상태
100	18/05/21	1500	이지우	배송중
101	18/05/21	2100	김보미	결제대기
102	18/05/21	3500	김서하	구매확정

제1정규형, 제2정규형

주문상세

주문번호	상품번호	구매수량
100	2002	1
100	4092	1
100	3923	1
101	2002	1
102	2002	1
102	4092	1
102	5100	2

제1정규형, 제2정규형

상품

상품번호	상품범주	상품가격
2002	전자제품	150000
4092	주방용품	24000
3923	가구	350000
5100	서적	30000

제1정규형, 제2정규형

정규화 예제- 3정규화 (Non-UID에 대한 종속 제거)

IT&Value Ltd.

주문일반

주문번호	주문날짜	고객번호	주문상태
100	18/05/21	1500	배송중
101	18/05/21	2100	결제대기
102	18/05/21	3500	구매확정

고객

고객번호	고객이름
1500	이지우
2100	김보미
3500	김서하

주문상세

주문번호	상품번호	구매수량
100	2002	1
100	4092	1
100	3923	1
101	2002	1
102	2002	1
102	4092	1
102	5100	2

상품

상품번호	상품범주	상품가격
2002	전자제품	150000
4092	주방용품	24000
3923	가구	350000
5100	서적	30000

제1정규형, 제2정규형, 제3정규형

IT&Value Ltd.

연습문제

나는 관리기법으로 운영되는 강사지도 강좌를 제공하는 교육기관의 관리자이다. 우리는 많은 강좌를 가르치고 있으며 각 강좌는 CODE, 강좌명, 수업료를 가지고 있다.

UNIX 기초와 C 프로그래밍은 우리의 인기 강좌중의 하나이다. 강좌의 수업일수는 1~4 일간으로 다양하다. 각 강사는 여러개의 강좌를 가르친다. 이몽룡과 성춘향은 우리의 최고 강사 중의 하나이다. 우리는 각 강사의 이름과 전화번호를 관리한다.

우리는 강좌를 개설하고 강사를 배정한다. 각 강좌는 단 한명의 강사에 의해 진행되며 어떤 강좌는 아직 강사가 배정되지 않을 수도 있다.

한 학생이 동시에 여러 강좌를 수강할 수 있으며 많은 학생이 그렇게 한다. 예를 들어 ABC전자의 홍길동은 우리가 제공한 모든 강좌를 수강했다. 우리는 각 학생의 성명과 전화번호를 관리하고자 하며, 때로는 학생과 강사가 그들의 전화번호를 알려 주지 않을 때도 있다.

대부분의 강좌는 여러 번 평가를 실시하고 평가에는 출석사항을 반영하며 기본점수 40점을 부여하며 우리는 평가된 종합 결과만 관리하고자 한다.