K-DT BigData#2 (DB-SQL)

[K-DT 강사 아카데미] Big Data Track #2. Database와 SQL

[K-DT#02_230615] Database와 SQL

(1) 데이터 모델링(Data Modeling, DM)

- DM 개요
  - 데이터베이스(DB): 공용(shared)-통합(integrated)-저장(stored)-운영(operational)되는 데이터의 집합
  - DM 절차: 요구사항 → 논리 모델 → 물리 모델 → DB 구조 (프로세스 모델링: 요구사항→분석→설계→소스코드)
  - DM 정의: 데이터 모델(논리/물리) 생성 과정, DB 분석/설계, 데이터 요구를 분석 및 정의
  - 데이터 모델(데이터 구조): 데이터가 표현되는 방식을 기록한 추상 모델, 데이터 요소와 관계를 정의
  - 목적: DB 최적 설계, 일정 품질의 데이터 모델 생성, 모두가 이해 가능한 설계 결과물, 시스템 이해 증가
- DM 종류
  - 논리(Logical) DM: 모델링 거의 완료(모든 엔티티 및 속성-관계 표현), 물리적 스키마 설계 전, DM의 핵심 단계
  - 물리(Physical) DM: 특정 DBMS에 따라 DB로 저장(엔티티→테이블, 속성→칼럼), 인덱스/파티셔닝/저장공간 고려
- ER 모델
  - 개요: 피터 첸(1976), 표준적 방법, 개념/논리 모델링, ERD로 표현, 세 가지 요소(엔티티/관계/속성)
  - 정보공학(IE) 표기법: 제임스 마틴, 까마귀발(Crow's foot) 모델, 키/일반 속성을 다른 영역에 표기, 한정적 관계 표시
  - 바커 표기법: Caci-바커-오라클, 속성(* 필수 o 옵션), 두 엔티티 사이의 관계는 양방향으로 구체적 표현
  - 차이점: 옵션-필수 표시 (바커는 점선/실선, IE는 I/O로 구분) 및 위치가 반대
- 모델링 성과
  - 성공 요소: 현업과 공동 작업, 체계적 방법(데이터 중심, 데이터 무결성, 정규화), 담당자들 간 지식 공유

- 품질 저하 원인: 업무 혹은 자료 간 관계 분석 부족, 정규화 등 이론 부족, 이론적으로만 분석, 미검토

(2) 논리 모델링(Logical Modeling)

- 엔티티(Entity)
  - 정의: 사람/사물/장소/개념/사건 등의 실체, 궁극적 관리 대상, 명사형으로 표기
  - 엔티티 표기: 명사형, 둥근 사각형 기호, Key/Non-Key 속성 구분 표기
  - 엔티티 분류
    - 키 엔티티: 부모 없이 처음부터 존재 → 사원, 부서, 고객, 상품, 자재
    - 메인 엔티티: 업무의 중심(다른 엔티티로부터 생성) → 보험계약, 사고, 구매의뢰, 공사, 주문
    - 액션 엔티티: 반드시 다른 엔티티 존재 하에서 생성 → 상태이력, 차량수리내역, 상세주문내역
- 관계(Relationship)
  - 정의: 2개의 엔티티 간 존재하는 상호 연관성 (제3의 관점은 배제)
  - 관계 Cardinality: 두 엔티티 사이의 관계에 대한 대응 수 → 1:1, 1: N (N:1), M:N
    ※ 선택(Optional)-필수(Mandatory) 관계
  - 관계 분류: 일반적 관계, 순환관계, 병렬관계, 상호배타적 관계
- 속성(Attribute)
  - 정의: 더 이상 분할되지 않은 최소의 데이터 보관 단위, 반드시 식별자에 종속(제2정규형)
  - 분류#1: 식별자(identifier / primary key) - 설명자/비식별자(descriptor)
  - 분류#2
    - 기초 속성(basic attribute): 관리-유지되는 기본 속성 → 주문일자, 납기일자, 수량, 단가
    - 설계 속성(designed attribute): 존재하지 않지만, 설계자가 필요에 따라 생성 → 주문번호, 고객번호, 일련번호, 품목 코드
    - 추출 속성(derived attribute): 다른 속성의 가공으로 만들어진 중복적 속성 → 주문총액, 금액
  - 속성 정의 4단계: ① 최소 단위로 분할, ② 하나의 값인지 검증, ③ 추출 속성인지 검토, ④ 관리 수준을 검토
  - 속성 검증: 단일값 속성(solitary attribute) 제거, 다른 의미의 여러 속성 통합 금지, 속성-식별자 종속 관계 검토, 코드 속성은 의미에 따라 상호 독립성 유지
- 식별자

    - 정의: 엔티티의 인스턴스들 사이의 유일성을 보장하는 속성, 모든 엔티티에 존재, 식별자 구성 속성은 Not NULL
    - 선정 기준: 업무 활용도 높은 것, 유일성 보장 최소 집합, 반드시 존재(Not NULL)
    - 인조 식별자 생성 고려하는 경우: 이미 사용중, 편의나 효율성, 후보 식별자의 값이 불규칙 존재
    - 식별자 도출: 속성 중에서 검토, 관계를 고려, 식별자 검증
- 정규화
    - 개요: 불필요한 데이터의 중복 제거 및 데이터 일관성 유지 ※과도한 정규화 시 응답속도 지연 문제 발생 가능
    - 1차 정규화: 모든 속성은 하나의 값만, 각 속성의 모든 값은 동일 형식, 각 속성은 유일한 이름 가짐
    - 2차 정규화: (1차 정규화) + 식별자가 아닌 모든 속성은 식별자에 완전히 종속
    - 3차 정규화: (2차 정규화) + 이행 종속이 없어야 함
    - BCNF 정규형: 모든 결정자는 키가 되어야 함
- 데이터 표준화
    - 개요: 시스템 별로 산재되어 있는 데이터 정보 요소에 대한 명칭/정의/형식/규칙 원칙 수립하여 전사적/조직적으로 적용하는 것
    - 기대 효과: 정확한 데이터 사용으로 인한 올바른 의사 결정, 의사소통 명확, 데이터 품질 향상, 관리 비용 감소
    - 구성 요소: 공통원칙, 표준용어, 표준코드, 표준 도메인
- 도메인
    - 개요: 엔티티 속성이 가질 수 있는 데이터의 집합
    - 도메인 추출: 모든 속성 추출 → 공통 접미어 묶어 이름 부여 → 도메인 별 데이터 타입(문자/숫자/날짜)과 길이 지정 → 각 엔티티 속성에 도메인 할당
    - 도메인 정의: 공통되고 일정한 규칙에 따라 속성에 대한 규칙을 정의하는 방법 → 도메인 정의서 작성
- 모델 검증
    - 구조적 타당성(structural validity), 단순성(Simplicity), 비중복성(Non-Redundancy), 공유성(Commonality)
    - 무결성(Integrity), 완전성(Completeness), 자명성(Self-Explanation), 확장성(Extensibility), 정규성(Normality)

(3) 물리 모델링(Physical Modeling)

- 개요 및 절차

    - 개요: 논리 모델을 특정 DB에 맞도록 물리적인 스키마로 설계하는 과정(엔티티→테이블, 속성→칼럼)
    - 절차: 모델 최적화(반정규화, 인덱스 설계, 파티션 고려), 무결성 규칙 설계, 논리-물리 모델 변환, DDL 스크립트 생성-실행하여 테이블 구조 생성
- 반정규화
    - 정의: 정규화 된 모델을 정규화를 위반하는 구조로 재조정하는 작업
      → 성능 등으로 인하여 실시 (다른 튜닝으로 해결되지 않는 경우)
    - 속성 중복
        - 접근 경로의 단축을 위하여 직접적인 관계가 없는 테이블의 외부 키를 복사
        - 조합된 계산 결과를 많이 적용하는 경우, 다중 테이블의 접근을 줄이고자
    - 테이블 중복: 정규화로 인하여 쿼리문의 속도가 느려질 경우 (집계, 진행, 특정 부분 테이블)
    - 테이블 통합: 테이블 구조가 유사한 경우 통합 고려 가능
    - 테이블 수평 분할: 테이블 내부 Row 개수가 매우 다량이라 쿼리 성능이 느려질 경우
    - 테이블 수직 분할: 조회/갱신 구분, 특정 칼럼의 특성(자주 조회, 아주 큰 크기, 보안 적용 필요)
- 인덱스: 조회 성능을 높이기 위함 → 10~15% 이내의 테이블 자료 접근 시 효율적
    - 인덱스 적용 대상 테이블: 중대형 규모, 10~15% 이내의 데이터 요구, 무작위 접근이 빈번한 테이블 등
    - 인덱스 컬럼 선정 기준: 분포도(Cardinality) 좋고, 수정이 빈번하지 않고, 외부키/조인의 연결, 정렬기준 등
    - 인덱스 선정 시기 및 절차
        - 시기: 설계 단계(기본 인덱스 선정), 개발 단계(최적 인덱스 선정)
        - 절차: 접근 경로 조사 → 인덱스 칼럼 선정 및 분포도 조사 →CAP (Critical Access Path) 및 우선순위 결정 → 인덱스 컬럼의 조합/순서 결정 → 시험 생성 및 테스트 → 반영 → 적용
- 논리-물리 변환
    - 원칙: ① 엔티티→테이블, ② UID→PK, ③일반 속성→칼럼, ④ 일반 관계→칼럼, ⑤ 배타적 관계 변환 형태 결정, ⑥ 제약조건 및 파라미터 결정
    - 관계의 변환: ① 1:N은 1의 PK가 N의 FK로, ② 1:1은 NULL이 생성되지 않는 방향으로 변환, ③ M:N은 양쪽의 PK를 합쳐서 복합 속성이 생성되도록
    - 수퍼서브 타입: ① 수퍼타입 엔티티로 통합, ② 서브타입 엔티티 기준으로 통합, ③ 수퍼/서브타입을 각각 분리
    - ARC 타입: 서로 배타적인 관계 → 예: 개인/법인/단체

    - 계층 관계(회사-사업부-팀-파트): ① 계층별로 엔티티 구성 vs. ② 1개의 순환 엔티티 구성
    - 다대다(M:N) 관계: 다대다 순환 관계는 1대다 2개로(1:M 및 1:N) 변환
- 무결성
    - 개요: 데이터의 일관성과 정확성을 보장하는 것 → 엔티티/참조/도메인 무결성
    - 엔티티 무결성: 기본키는 유일성을 보장해 주는 최소한의 집합이며, 그 속성은 Not NULL
    - 참조 무결성(Referential Integrity): 실체 간의 관계에서 파생되는 데이터 무결성
    - 도메인 무결성: 제약조건 추가 → Data Type, Length, Default Value, Constraints, Uniqueness, Null Support 등 ※ Check 제약조건을 둘 수 있음

(End of Document)