

과목I. 2장 - 데이터 모델링과 성능

■ 성능 데이터 모델링

- 데이터베이스 성능 향상을 목적으로 설계 단계의 데이터 모델링 때부터 성능과 관련된 사항이 데이터 모델링에 반영될 수 있도록 함

ex) 어떤 트랜잭션이 해당 비즈니스 처리에 핵심적 & 사용자 업무처리에 중요 & 성능 저하 X

=> 프로젝트 초기에 운영환경에 대비한 테스트 환경을 구현하고 그곳에 트랜잭션을 발생시켜 실제 성능 테스트

데이터 모델의 구조도 변경하면서 어떠한 구조가 성능상 가장 적절한 구조인지 검토하여 성능이 좋은 모습으로 디자인

■ 성능 데이터 모델링 절차

- ① 데이터 모델링을 할 때 정규화를 정확하게 수행함
- ② 데이터베이스 용량 산정을 수행함
- ③ 데이터베이스에 발생하는 트랜잭션 유형을 파악함
- ④ 용량과 트랜잭션의 유형에 따라 반정규화를 수행함
- ⑤ 이력모델의 조정, PK/FK 조정, 슈퍼타입/서브타입 조정 등을 수행함
- ⑥ 성능 관점에서 데이터 모델을 검증함

■ 성능 데이터 모델링 고려사항

- 중복된 데이터를 제거함으로써 조회 성능을 향상시킬 수 있음
- 용량 산정은 전체적인 데이터베이스에 발생하는 트랜잭션의 유형과 양을 분석하는 자료가 됨
- 물리적인 데이터 모델링을 할 때 PK/FK의 칼럼의 순서 조정, FK 인덱스 생성 등은 성능 향상을 위한 중요한 요소
- 이력 데이터는 시간에 따라 반복적으로 발생되기 때문에 대량 데이터일 가능성이 높아 성능을 고려하여 칼럼 등을 추가하도록 설계

■ 정규화

- 이상 현상이 발생하지 않도록 엔터티를 관련있는 속성들로만 구성하기 위해 엔터티를 분해하는 과정 (무손실 분해)
- 엔터티에 함수적 종속성이 하나 존재하도록, 관련이 없는 함수 종속성은 별개의 엔터티로 표현
- 분해된 엔터티를 자연 조인하여 분해 전의 엔터티로 다시 복원할 수 있어야 함

■ 1차 정규화 (1NF)

- 엔터티에 속한 모든 속성의 도메인이 원자 값으로만 구성되어 있으면 제1정규형에 속함
- 하나의 속성이 다중 값을 포함하는 경우
- 같은 성격과 내용의 칼럼이 연속적으로 나타나는 경우

■ 2차 정규화 (2NF)

- 제1정규형에 속하고, 기본키가 아닌 모든 속성이 기본키에 완전 함수 종속되면 제2정규형에 속함
- PK가 여러키로 구성된 복합키인 경우
- 복합키의 일부분에만 종속되는 속성들이 존재하는 경우 (부분 함수 종속 관계)

■ 3차 정규화 (3NF)

- 제2정규형에 속하고, 기본키가 아닌 모든 속성이 기본키에 이행적 함수 종속이 되지 않으면 제3정규형에 속함
- 이행적 함수 종속 관계 제거 ($X \rightarrow Y, Y \rightarrow Z \Rightarrow X \rightarrow Z$)
- PK에 의존하는 것처럼 보이지만 실제로 일반 칼럼에 의존하는 칼럼 분리

■ 반정규화

- 정규화된 엔터티, 속성, 관계에 대해 성능 향상과 개발, 운영의 단순화를 위해 중복, 통합, 분리 등을 수행하는 데이터 모델링 기법
- 데이터 무결성이 깨질 수 있는 위험을 무릅쓰고 데이터를 중복하여 반정규화를 적용함
- 데이터를 조회할 때 디스크 I/O량이 많거나 경로가 멀거나 칼럼을 계산하여 읽어야 하는 경우 성능 저하가 예상되어 반정규화를 수행함

■ 반정규화 고려할 때 판단 요소

- 반정규화 정보에 대한 재현의 적시성으로 판단.
- 탐색 대상 데이터의 크기로 판단.

다량 데이터 탐색의 경우 인덱스가 아닌 파티션 및 데이터 클러스터링 등의 물리 저장 기법을 활용하여 성능 개선 유도 가능.

BUT 하나의 결과셋을 추출하기 위해 다량의 데이터를 탐색하는 처리가 반복적으로 빈번하게 발생한다면 반정규화 고려함.

- 집계 테이블 이외에도 다양한 유형에 대하여 반정규화 테이블 적용 필요할 수 있음 (ex. 다수 테이블의 키 연결 테이블 등)

■ 테이블의 반정규화

기법 분류	반정규화 기법
테이블 병합	1:1 관계 테이블병합
	1:M 관계 테이블병합
	슈퍼/서브타입 테이블병합
테이블 분할	수직 분할
	수평 분할
테이블 추가	중복 테이블 추가
	통계 테이블 추가
	이력 테이블 추가
	부분 테이블 추가

■ 칼럼의 반정규화

반정규화 기법
중복 칼럼 추가
파생 칼럼 추가
이력 테이블 칼럼 추가
PK에 의한 칼럼 추가
응용시스템 오작동을 위한 칼럼 추가

■ 반정규화 절차

- ① 반정규화 대상 조사 : 범위처리 빈도수 조사 / 대량의 범위처리 조사 / 통계성 프로세스 조사 / 테이블 조인 개수
- ② 다른 방법 유도 검토 : 뷰 테이블 / 클러스터링 적용 / 인덱스 조정 / 응용애플리케이션
- ③ 반정규화 적용 : 테이블 반정규화 / 속성의 반정규화 / 관계의 반정규화

■ 반정규화의 대상에 대해 다른 방법으로 처리

- 뷰 테이블 : 지나치게 많은 조인이 걸려 데이터를 조회하는 작업이 기술적으로 어려울 경우에 사용
- 클러스터링 조정 or 인덱스 조정 : 대량의 데이터처리나 부분처리에 의해 성능이 저하되는 경우에 적용
- 파티셔닝 기법 : 대량의 데이터를 Primary Key의 성격에 따라 여러 개의 부분적인 테이블로 분리
- 응용 애플리케이션 : 로직을 구사하는 방법을 변경

■ 슈퍼/서브 타입 데이터 모델의 변환 기술

- 개별로 발생하는 트랜잭션에 대해서는 개별 테이블로 구성
- 슈퍼타입 + 서브타입에 대해 발생하는 트랜잭션에 대해서는 슈퍼타입 + 서브타입 테이블로 구성
- 전체를 하나로 묶어 트랜잭션이 발생할 때는 하나의 테이블로 구성
- * 트랜잭션은 항상 전체를 통합하여 분석 처리하는데 슈퍼-서브타입이 하나의 테이블로 통합되어 있으면 하나의 테이블에 집적된 데이터만 읽어 처리할 수 있기 때문에 다른 형식에 비해 더 성능이 우수함 (조인 감소)

■ PK 인덱스

- 인덱스를 효율적으로 이용할 수 있도록 순서 지정
- 앞쪽에 위치한 속성의 값이 비교자로 있어야 인덱스가 좋은 효율을 가짐
- 상수값으로 'EQUAL' 조건으로 조회되는 칼럼이 가장 앞으로, 범위조회하는 유형의 칼럼이 그 다음에 오도록 하는 것이 베스트

■ FK 인덱스

- 엔터티 간에 논리적 관계가 있을 경우, 상호간에 조인이 자주 발생한다는 것을 의미함 => 조인 성능을 향상시키기 위한 인덱스 생성
- 데이터베이스에 생성하는 FK Constraints는 데이터 모델에 표현된 논리적 관계에 따라
관련 인스턴스 간에 일관성을 보장하기 위해 설계된 제약조건을 구현할 수 있도록 DBMS가 제공하는 지원 기능

■ 분산 데이터베이스 장단점

■ 장점

- 지역 자치성, 점증적 시스템 용량 확장
- 신뢰성과 가용성
- 효율성과 융통성
- 빠른 응답 속도와 통신 비용 절감
- 데이터의 가용성과 신뢰성 증가
- 시스템 규모의 적절한 조절
- 각 지역 사용자의 요구 수용 증대

■ 단점

- 소프트웨어 개발 비용
- 오류의 잠재성 증대
- 처리 비용의 증대
- 설계, 관리의 복잡성과 비용
- 불규칙한 응답 속도
- 통제의 어려움
- 데이터 무결성에 대한 위협

■ 분산데이터베이스 환경에서 데이터베이스 분산 설계를 적용하여 효율성 증대

- 공통코드, 기준정보 등과 같은 마스터 데이터는 분산데이터베이스에 복제분산 적용
- 거의 실시간 업무적인 특성을 가지고 있을 때 분산데이터베이스를 사용하여 구성 가능
- 백업 사이트를 구성할 때 간단하게 분산기능을 적용하여 구성 가능