|  |  |  |
| --- | --- | --- |
| 기법분류 | 기법 | 내용 |
| 테이블병합 | 1:1 관계 테이블병합 | 1:1 관계를 통합하여 성능향상 |
| 1:M 관계 테이블병합 | 1:M 관계를 통합하여 성능향상 |
| 슈퍼/서브타입 테이블병합 | 슈퍼/서브 관계를 통합하여 성능향상 |
| 테이블병합 | 수직분할 | 칼럼단위의 테이블을 디스크 I/O를 분산처리하기 위해 테이블을 1:1로 분리하여 성능향상 (트랜잭션이 처리되는 유형을 파악하는 것이 선행되어야 함) |
| 수평분할 | 로우 단위로 집중 발생되는 트랜잭션을 분석하여 디스크 I/O 및 데이터접근의 효율성을 높여 성능을 향상하기 위해 ROW 단위로 테이블을 쪼갬 (관계가 없음) |
| 테이블추가 | 중복테이블 추가 | 다른 업무이거나 서버가 다른 경우 동일한 테이블 구조를 중복하여 원격조인을 제거하여 성능을 향상 |
| 통계테이블 추가 | SUM, AVG 등을 미리 수행하여 계산해 둠으로써 조회 시 성능을 향상 |
| 이력테이블 추가 | 이력테이블 중에서 마스터 테이블에 존재하는 레코드를 중복하여 이력테이블에 존재하는 방법은 반정규화의 유형 |
| 부분테이블 추가 | 하나의 테이블의 전체 칼럼 중 이용 빈도가 집중되는 칼럼들이 있을 때 디스크 I/O를 줄이기 위해 해당 칼럼들을 모아둔 별도의 반정규화된 테이블을 생성 |

|  |  |
| --- | --- |
| 반정규화 기법 | 내용 |
| 중복칼럼 추가 | 조인에 의해 처리할 때 성능저하를 예방하기 위해 즉, 조인을 감소시키기 위해 중복된 칼럼을 위치시킴 |
| 파생칼럼 추가 | 트랜잭션이 처리되는 시점에 계산에 의해 발생되는 성능저하를 예방하기 위해 미리 값을 계산해서 칼럼에 보관한다. AKA. Derived Column |
| 이력테이블 칼럼추가 | 대량의 이력데이터를 처리할 때 불특정 일자 조회 혹은 최근 값 조회 시에 나타날 수 있는 성능저하를 예방하기 위해 이력테이블에 기능성 칼럼 (최근값 여부, 시작과 종료일자)을 추가함 |
| PK에 의한 칼럼 추가 | 복합의미를 갖는 PK를 단일 속성으로 구성하였을 경우 발생됨. 단일 PK 안에서 특정 값을 별도로 조회하는 경우 성능저하가 발생될 수 있음. 이때 이미 PK 안에 데이터가 존재하지만 성능향상을 위해 일반속성으로 포함하는 방법이 PK에 의한 칼럼추가 반정규화임 |
| 응용시스템 오작동을 위한 칼럼 추가 | 업무적으로는 의미가 없지만 사용자가 데이터처리를 하다가 잘못 처리하여 원래 값으로 복구하기를 원하는 경우 이전 데이터를 임시적으로 중복하여 보관하는 기법. 칼럼으로 이것을 보관하는 방법은 오작동 처리를 위한 임시적인 기법이지만 이것을 이력데이터 모델로 풀어내면 정상적인 데이터 모델의 기법이 될 수 있음 |