이중화(Replication)

2006-07-18

김정준
데이타베이스 연구실

차례

이중화(Replication)란?

- 이중화(Replication)
  - 물리적으로 떨어져 있는 여러 개의 데이터베이스에 대하여 로컬 데이터베이스의 변경된 내용을 원격 데이터베이스에 복제하고 관리하는 것을 말함

이중화의 목적

- 이중화 목적
  - 데이터베이스의 무정지 서비스를 가능
    - 사용자는 하나의 데이터베이스에 대해서만 작업을 수행
    - 데이터베이스 이중화 시스템에 연결되어 있는 다른 데이터베이스에도 작업 내용이 동일하게 적용
    - 여러 개의 데이터베이스를 동시에 관리

이중화의 특징

- High Availability
  - 서비스 중이던 시스템 또는 S/W가 고장이 나면 사용 가능한 시스템으로 즉시 접근이 가능하여야 함
- Database Consistency
  - 하나의 데이터베이스 서버 내에서 이중화 트랜잭션과 로컬 트랜잭션이 동시에 같은 데이터를 접근하는 경우가 발생
  - 이런한 데이터의 충돌(conflict) 발생시 데이터 충돌이 해결되어야 함
- High Performance
  - 이중화하는데 수반되는 오버헤드를 최소화
  - 독립 시스템(standalone)으로 트랜잭션을 처리할 때의 성능을 유지하도록 하여야 함
- Load Balancing & Scalability
  - 다중 서버 운영 환경에서 서비스하는 트랜잭션들을 두 그룹 이상으로 나누어, 각각의 트랜잭션이 해당 서버에서 수행되도록 하여야 함
  - 각 서버에서 변경되는 데이터베이스 내용을 다른 서버들에 반영시킴으로써 서버에 걸리는 부하를 분산시킬 수 있도록 하여야 함

이중화의 분류

❖ 이중화의 분류

◆ 데이터 변경 내용을 전달하는 방법에 의하여 “Eager” 기법과 “Lazy 기법”으로 분류

■ Eager 기법

- 한 객체에 대한 변경 내용이 그 트랜잭션의 일부로 인식되어 수행되어짐
- 트랜적션 수행중에 발생한 변경은 발생 즉시 모든 이중화 서버로 전달되어 연쇄적으로 변경 내용이 반영

■ Lazy 기법

- 트랜잭션의 수행이 완전히 완료된 후에 그 변경 사실에 대한 새로운 트랜잭션을 작성하여 각 노드에게 전달하는 기법
- 각 노드마다 또 다른 새로운 트랜잭션이 수행되어 지는 것으로 간주

	전역 완료	일관성	응답 시간	상호 조화	데드로크 비율
Eager	2단계 완료 3단계 완료	엄격함	늦음	불필요	높음
Lazy	지역 완료	약화됨	빠름	필요	낮음

이중화의 분류

❖ 이중화의 분류

◆ 객체의 소유권 및 변경의 주체에 따라서 “그룹 기법”과 마스터 기법” 으로 분류

■ 그룹 기법

- 임의의 노드가 임의의 객체를 변경할수 있으며, 그 변경사실을 모드 노드에게 전달하는 기법

■ 마스터 기법

- 객체의 변경할수 있는 노드가 결정되어 있어서 그 노드에서만 객체를 변경할 수 있고, 그변경 내용을 다른 모든 노드에게 전달하는 기버

	변경 가능 노드	전역로크 처리	응용 유연성	충돌 해결	데드로크 비율
그룹	임의 노드에서 변경 가능	필요	높음	필요, 타임스탬프 기반	높음
마스터	마스터만 변경 가능	불필요	낮음	부분적 필요	낮음

이중화의 분류

❖ 상용 시스템들의 이중화 기법

변경시점 / 변경장소	Eager	Lazy
마스터 기법	- ALTIBASETM의 초기 방안 - INGRES - ORACLE Synchronous Replication	- ALTIBASETM 이중화 기법 - SYBASE 이중화 서버 - IBM Data Propagator - ORACLE Placement 전략
그룹 기법	- ROWA/ROWAA(쿼럼 방식) (Read-one/Write-all, Read-one/Write-all-available) - ORACLE Synchronous Replication	- Timesten - ORACLE Symmetric Replication (asynchronous)

이중화의 예

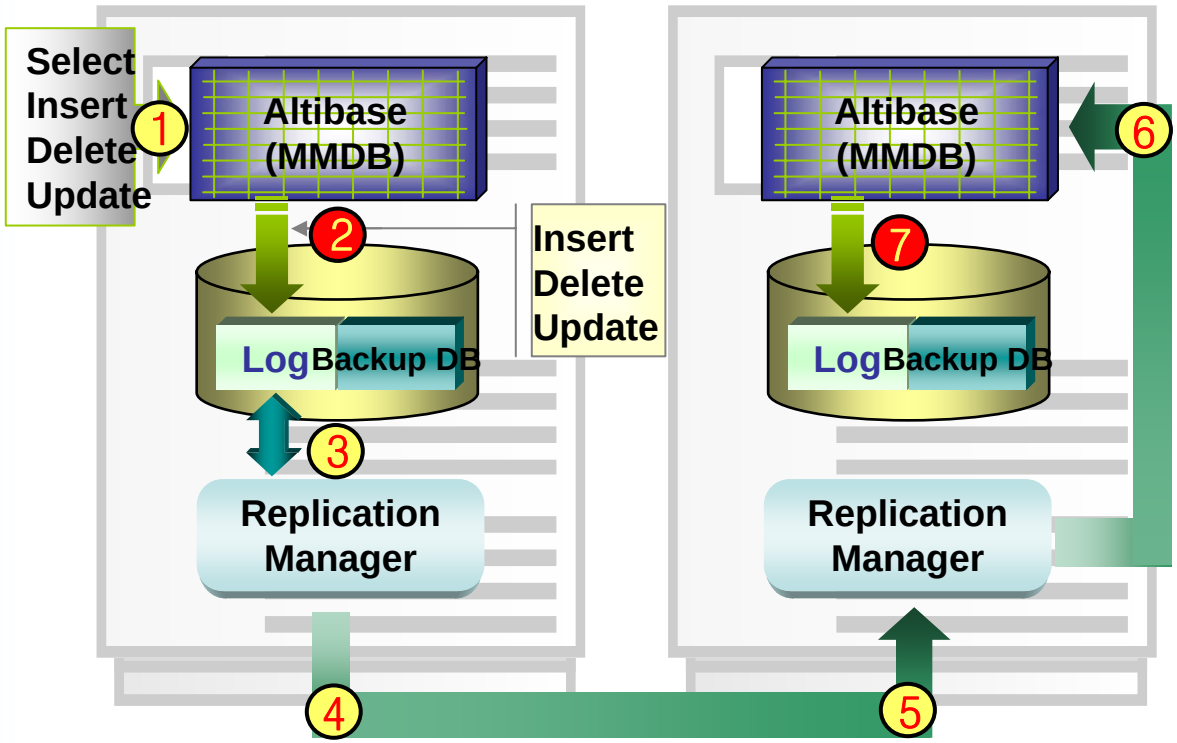
- 알티베이스의 이중화
  - 로그 기반
    - 데이터베이스 로그를 기반으로 이중화 구현
      - 서버 부하를 최소화
  - 테이블 단위 이중화
    - 전체 데이터베이스 중 일부 테이블만 이중화
      - 운영 효율성 증대
  - 실시간 이중화
    - 알티베이스 실시간 트랜잭션 처리와 함께, 실시간으로 이중화를 제공
      - 실시간 서비스 구현 용이
    - 가용성 증대
      - 장애 시 실시간 이중화된 서버로 즉시 접근하여 중단 없이 서비스 가능

이중화의 예

- 알티베이스의 이중화
  - 구현 및 운영의 용이성
    - TCP/IP Network을 이용한 이중화 구현
      - 추가적인 Hardware 투자 불필요
    - SQL과 유사한 사용자 인터페이스 제공
      - 이중화 정의 및 운영이 간편
  - 높은 안정성
    - 원격 이상(원격 서버 고장 및 네트웍 단절) 감지 및 Failure Handling
      - 이중화 회복을 통해 데이터 일치성 제공
    - 이중화 트랜잭션 컨텍스트 관리
      - 동등한 이중화 트랜잭션 처리 제공
      - 로컬 및 원격 트랜잭션간 독립성 보장
  - 다양하고 유연한 시스템 구성
    - 업무에 따라, Point-Point 및 Propagation방식으로 구성
      - 다양한 무정지 시스템 구성이 가능
    - N-WAY 이중화 제공
      - 확장이 용이한 부하 분산(Load Balancing) 시스템 구성이 가능

이중화의 예

❖ 알티베이스의 이중화



1. SQL문 처리
2. 트랜잭션 처리
3. 이중화 로그 변환
4. 이중화 로그 송신
5. 이중화 로그 수신
6. 실행계획 변환
7. 이중화 트랜잭션 처리

이중화의 예

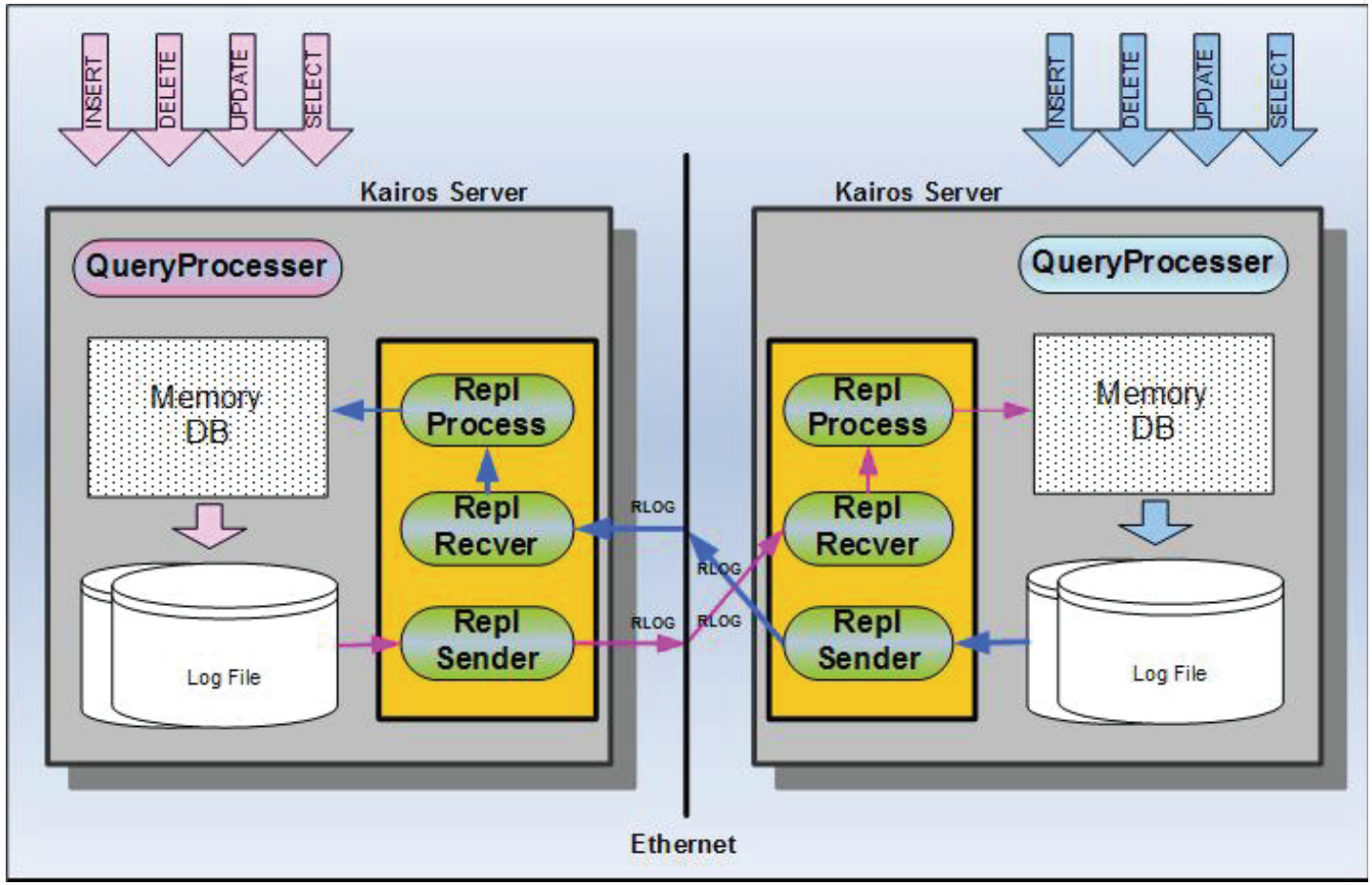
- 카이로스의 이중화
  - 서버 이중화 방법으로 Active-Standby 로 또는 Active-Active 방법을 제공
  - 서버간의 데이터 복제하는 옵션으로 동기적(Synchronous) 또는 비동기적(Asynchronous) 옵션을 제공
  - 이중화는 2대의 Kairos 서버(로컬서버, 리모트 서버)로 구성
  - 각 서버에는 이중화 처리를 위한 Alivecheck, Sender, Receiver, Processing agent들이 존재
  - 로그 기반의 이중화
  - 키 기반의 충돌 해결

이중화의 예

- 카이로스의 이중화
  - A-S 방식
    - 서버둘 중 한 서버만 서비스를 수행하는 경우
  - A-A 방식
    - 두 개의 Kairos 서버가 모두 서비스를 할 수 있는 경우
  - Asynchronous 방식
    - 로컬서버 변경정보가 리모트서버로 전달되는 시점이 log 에 기록되는 시점과는 완전히 분리된 방식
  - Synchronous 방식
    - 로컬서버 변경정보를 log에 기록하기 전에 상대편 Kairos 서버로 전달이 된후에 기록을 하는 방식

이중화의 예

❖ 카이로스의 이중화

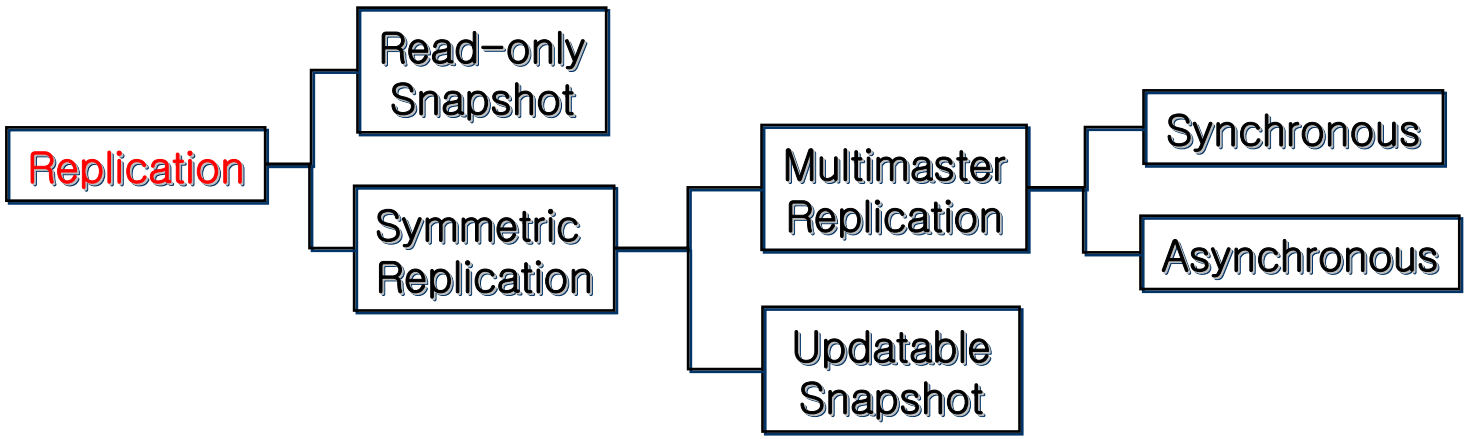


이중화의 예

- 오라클의 이중화
  - 분산 데이타베이스상에서 테이블 등의 데이타베이스 오브젝트를 여러 데이타베이스로 복사하여 유지하는 기법을 지칭
  - Read-only Snapshot
    - 한 데이타베이스의 마스터 테이블에 대해서 다른 분산 데이타베이스가 그 테이블에대한 복사본, 즉 스냅샷(Snapshot)을 유지
    - 마스터 테이블에 대해서 발생한 변경사항이 주기적으로 스냅샷으로 반영되어 스냅샷 사이트는 마스터 테이블에 대한 이미지를 스냅샷을 통해 쉽게 읽을 수 있음
    - 단, 마스터 테이블의 변경 사항이 실시간으로 반영되어야 하거나, 스냅샷 사이트에서도 해당 스냅샷에 대한 변경이 꼭 필요한 사항이라면 Read-only Snapshot이 적합하지 않을 수 있음
  - Symmetric Replication
    - 양쪽의 복사본이 모두 변경 가능하고, 각 변경 사항은 같은 리플리케이션 환경에 포함된 데이타베이스로 모두 전달

이중화의 예

- 오라클의 이중화
  - Synchronous
    - 한 마스터 사이트의 변경 사항이 바로 실시간으로 다른 마스터 사이트로 반영되어야 하고, 마스터 사이트간의 데이타는 서로 완전히 일치해야 하는 경우
  - Asynchronous
    - 로컬 데이타베이스의 테이블에 변경이 발생하면 그 변경 정보를 로컬 큐에 저장한 후, 일정한 시간 간격마다 해당 정보를 리모트 사이트로 전달



이중화의 예

❖ 오라클의 이중화