

# Tugas Basis Data Sesi 13

Aldi Maulana Iqbal – 20210801222

## **Langkah capture diimplementasikan dengan menggunakan dua pendekatan, sebutkan dan jelaskan.**

Dalam database terdistribusi, langkah capture dapat diimplementasikan dengan menggunakan dua pendekatan utama:

1. Pendekatan "Two-Phase Commit" (2PC): Pendekatan ini menggunakan mekanisme komit transaksi yang terdistribusi untuk menjamin bahwa transaksi yang dilakukan pada basis data terdistribusi akan selalu dalam kondisi konsisten. Proses komit transaksi dilakukan dengan melibatkan dua fase, yaitu fase prepare dan fase commit. Pada fase prepare, sistem mengecek apakah transaksi dapat dilakukan pada semua basis data yang terlibat. Jika semua basis data dapat menerima transaksi, maka sistem akan melanjutkan ke fase commit dan melakukan komit pada semua basis data yang terlibat. Jika salah satu basis data tidak dapat menerima transaksi, maka sistem akan melakukan rollback pada semua basis data yang telah menerima transaksi.
2. Pendekatan "Practical Byzantine Fault Tolerance" (PBFT): Pendekatan ini menggunakan algoritma konsensus yang didesain untuk mengatasi masalah Byzantine Fault pada sistem terdistribusi. PBFT mengimplementasikan mekanisme konsensus yang didasarkan pada komunikasi antara node-node dalam sistem. Setiap transaksi yang dilakukan harus mendapatkan persetujuan dari sebagian besar node dalam sistem sebelum dapat dikomitkan. Jika sebagian besar node dalam sistem setuju untuk mengeksekusi transaksi, maka transaksi akan dikomitkan pada semua basis data yang terlibat. Namun, jika sebagian besar node dalam sistem tidak setuju untuk mengeksekusi transaksi, maka transaksi akan dibatalkan.

## **Jelaskan beberapa cara menangani penguncian obyek.**

Beberapa cara untuk menangani penguncian obyek dalam sistem basis data adalah:

1. Penguncian objek berskala kecil (Fine-grained locking): Dalam cara ini, sistem basis data mengunci obyek dengan skala yang sangat kecil, misalnya mengunci sebuah baris atau sebuah kolom dalam tabel. Hal ini memungkinkan beberapa transaksi untuk berlangsung secara bersamaan pada obyek yang berbeda, namun akan terjadi konflik jika beberapa transaksi mengakses obyek yang sama.
2. Penguncian objek berskala besar (Coarse-grained locking): Dalam cara ini, sistem basis data mengunci obyek dengan skala yang lebih besar, misalnya mengunci seluruh tabel atau seluruh basis data. Hal ini memungkinkan hanya satu transaksi untuk berlangsung pada suatu saat, namun akan mengurangi kemungkinan terjadinya konflik.
3. Penguncian obyek berskala dinamis (Dynamic-grained locking): Dalam cara ini, sistem basis data mengunci obyek dengan skala yang dinamis, yaitu mengunci obyek sesuai dengan kebutuhan transaksi yang berlangsung. Hal ini memungkinkan beberapa transaksi untuk berlangsung secara bersamaan, namun akan mengurangi kemungkinan terjadinya konflik dengan menyesuaikan skala penguncian obyek sesuai dengan kebutuhan transaksi yang berlangsung.
4. Penguncian obyek berskala transaksi (Transaction-level locking): Dalam cara ini, sistem basis data mengunci obyek pada tingkat transaksi. Transaksi yang saling

tergantung akan dikunci secara bersama-sama, sehingga memungkinkan beberapa transaksi untuk berlangsung secara bersamaan namun mengurangi kemungkinan terjadinya konflik.

5. Penguncian obyek berskala temporal (Temporal-grained locking): Dalam cara ini, sistem basis data mengunci obyek dengan skala temporal, yaitu mengunci obyek sesuai dengan jangka waktu yang ditentukan. Hal ini memungkinkan beberapa transaksi untuk berlangsung secara bersamaan, namun akan mengurangi kemungkinan terjadinya konflik dengan menyesuaikan skala penguncian obyek sesuai dengan jangka waktu yang ditentukan.

## **Jelaskan keuntungan DBMS terdistribusi dibandingkan dengan DBMS tersentralisasi**

DBMS (Database Management System) terdistribusi memiliki beberapa keuntungan dibandingkan dengan DBMS tersentralisasi, diantaranya:

1. Skalabilitas: DBMS terdistribusi dapat mengelola data yang lebih besar dan menangani lebih banyak koneksi dibandingkan dengan DBMS tersentralisasi. Hal ini karena DBMS terdistribusi dapat mengelola data yang tersebar di beberapa lokasi atau mesin yang berbeda.
2. Ketersediaan tinggi: DBMS terdistribusi dapat menyediakan akses data yang lebih tinggi dibandingkan dengan DBMS tersentralisasi. Hal ini karena DBMS terdistribusi dapat menyediakan replikasi data yang dapat digunakan untuk meningkatkan ketersediaan data.
3. Performa yang lebih baik: DBMS terdistribusi dapat menyediakan performa yang lebih baik dibandingkan dengan DBMS tersentralisasi. Hal ini karena DBMS terdistribusi dapat mengelola data yang tersebar di beberapa lokasi atau mesin yang berbeda, sehingga dapat meningkatkan performa dengan meningkatkan paralelisme.
4. Lokasi independen: DBMS terdistribusi dapat menyediakan akses data yang independen dari lokasi fisik. Hal ini karena DBMS terdistribusi dapat menyediakan akses data yang dapat diakses dari mana saja.
5. Pemulihan dari kesalahan: DBMS terdistribusi dapat menyediakan pemulihan dari kesalahan yang lebih baik dibandingkan dengan DBMS tersentralisasi. Hal ini karena DBMS terdistribusi dapat menyediakan replikasi data yang dapat digunakan untuk memulihkan data setelah terjadi kesalahan.

## **Jelaskan perbedaan antara replikasi synchronous dan asynchronous**

Replikasi adalah proses menyalin data dari satu lokasi ke lokasi lain agar data dapat digunakan secara bersamaan pada beberapa lokasi atau mesin yang berbeda. Ada dua jenis replikasi yang umum digunakan, yaitu replikasi synchronous dan asynchronous.

1. Replikasi synchronous: Replikasi synchronous adalah proses menyalin data secara real-time atau segera setelah data diperbarui pada sumber data. Dalam replikasi synchronous, setiap perubahan data pada sumber data akan segera diterapkan pada semua replika yang ada. Keuntungan dari replikasi synchronous adalah data yang tersedia pada semua replika selalu up-to-date dan konsisten. Namun, replikasi synchronous memerlukan koneksi jaringan yang cepat dan stabil serta dapat

menurunkan performa karena setiap perubahan data harus diterapkan pada semua replika.

2. Replikasi asynchronous: Replikasi asynchronous adalah proses menyalin data secara periodik atau tidak segera setelah data diperbarui pada sumber data. Dalam replikasi asynchronous, perubahan data pada sumber data tidak segera diterapkan pada semua replika, tapi akan diterapkan pada interval waktu yang telah ditentukan. Keuntungan dari replikasi asynchronous adalah performa sistem tidak terpengaruh oleh proses replikasi, dan tidak memerlukan koneksi jaringan yang cepat dan stabil. Namun, data yang tersedia pada replika tidak selalu up-to-date dan konsisten.