

## *KONSEP DAN RANCANGAN BASIS DATA TERDISTRIBUSI*

### **SISTEM BASIS DATA TERDISTRIBUSI**

#### **DEFINISI**

---

Basis Data Terdistribusi adalah kumpulan data logic yang saling berhubungan secara fisik terdistribusi dalam jaringan komputer.

File merupakan kumpulan data yang dirancang untuk suatu aplikasi.

## Contoh Basis Data Terdistribusi

---

- Misalnya sebuah bank yang memiliki banyak cabang, bahkan di sebuah kota bisa terdiri dari beberapa cabang/kantor.
- Masing-masing lokasi memiliki jaringan lokal sendiri, dan semua jaringan lokal itu dihubungkan satu sama lain membentuk sebuah jaringan nasional.

## Struktur Basis Data Terdistribusi

---

- Sebuah sistem basis data terdistribusi hanya mungkin dibangun dalam sebuah sistem jaringan komputer.
- Topologi Jaringan
  - a. Topologi bintang (star)
  - b. Topologi Cincin (Ring)
  - c. Topologi Bus

## Struktur Basis Data Terdistribusi

---

- Perbedaan utama di antara berbagai topologi di atas terletak pada:
  - a. Biaya Instalasi**

Biaya dalam membangun hubungan link (link) antar simpul.
  - b. Biaya Komunikasi**

Waktu dan biaya dalam pengoperasian sistem berupa pengiriman data dari satu simpul ke simpul lain.

## Struktur Basis Data Terdistribusi

---

- c. Keandalan**

Frekuensi/tingkat kegagalan komunikasi yang terjadi.
- d. Ketersediaan**

Tingkat kesiapan data yang dapat diakses sebagai antisipasi kegagalan komunikasi.

## Jenis Transaksi

---

- **Transaksi Lokal**

Transaksi yang mengakses data pada suatu simpul (mesin/server) yang sama dengan simpul dari mana transaksi tersebut dijalankan.

- **Transaksi Global.**

Transaksi yang membutuhkan pengaksesan data di simpul yang berbeda dengan simpul dimana transaksi tersebut dijalankan, atau transaksi dari sebuah simpul yang membutuhkan pengaksesan data ke sejumlah simpul lainnya.

## Ciri Ciri Basis Data Terdistribusi

---

- Data disimpan di sejumlah tempat
- Processor pada tempat yang berbeda tersebut dihubungkan dengan jaringan komputer
- Sistem basis data terdistribusi bukan terdiri dari sekumpulan file yang berada pada berbagai tempat tetapi pada sebuah basis data di berbagai tempat
- Setiap tempat secara mandiri memproses permintaan user yang membutuhkan akses ke data di tempat tersebut dan juga mampu untuk memproses data yang tersimpan di tempat lain.

## Keuntungan yang diberikan oleh sistem basis data terdistribusi

---

- Pengelolaan secara transparan data yang terdistribusi
- Mengacu pada struktur organisasi
- Meningkatkan untuk berbagai dan otonomi lokal
- Meningkatkan ketersediaan data
- Meningkatkan kehandalan
- Meningkatkan performasi kerja
- Memudahkan pengembangan sistem

## Kerugian yang diberikan sistem basis data terdistribusi

---

- Kompleksitas manajemen
- Kontrol integritas lebih sulit
- Biaya pengembangan
- Keamanan
- Sulitnya standarisasi
- Menambah kebutuhan penyimpanan
- Lebih sulit dalam mengatur lingkungan data

## DESAIN BASIS DATA TERDISTRIBUSI

---

- Ada beberapa pendekatan yang berkaitan dengan penyimpanan data/tabel dalam sebuah sistem basis data terdistribusi, yaitu
- 1. Replikasi
- 2. Fragmentasi
- 3. Replikasi dan Fragmentasi

## DESAIN BASIS DATA TERDISTRIBUSI

---

### 1. Replikasi

- Sistem memelihara sejumlah salinan/duplikat tabel-tabel data.
- Setiap salinan tersimpan dalam simpul yang berbeda, yang menghasilkan replikasi data

## DESAIN BASIS DATA TERDISTRIBUSI

---

### 2. Fragmentasi

- Data dalam tabel dipilah dan disebar ke dalam sejumlah fragmen.
- Tiap fragmen disimpan di sejumlah simpul yang berbeda-beda.
- Fragmentasi dapat berbentuk fragmentasi horizontal (pemilahan record data) atau fragmentasi vertikal pemilahan field/atribut data)

## DESAIN BASIS DATA TERDISTRIBUSI

---

### 3. Replikasi dan Fragmentasi

- Merupakan kombinasi dari replikasi dan fragmentasi.
- Data/tabel dipilah dalam sejumlah fragmen.
- Sistem lalu mengelola sejumlah salinan dari masing-masing fragmen tadi di sejumlah simpul.

## Replikasi Data

---

- Keuntungan :
  1. Ketersediaan yang tinggi.
  2. Peningkatan Keparalelan (Increased parallelism)
  3. Peningkatan beban perubahan data (Increased overhead on update)

## Fragmentasi Data

---

- Dapat diterapkan operasi union ataupun operasi natural join.
- Contoh  
Tabel nasabah bank dengan struktur dan data sebagai berikut :  
**nasabah = (no\_nas, nama\_nas, alamat\_nas, kota, saldo\_simpan, saldo pinjam)**

## Fragmentasi Data

---

- Tabel Nasabah

No_nas	Nama_nas	Alamat	Kota	Saldo_simp an	Saldo_pinj am
2001001	Jamaludin	Jl. Suci 10	Medan	100000	0
2001002	Nurhaliza	Jl. Abdi 22	Medan	2300000	1450000
2001003	Nur Alam	Jl. Aceh 10	Medan	1200000	3500000
2001004	Setia	Jl. Jawa 112	Medan	1500000	1000000
2002001	Tini	Jl. Adil 20	Padang	50000	0
2002002	Ahmad	Jl. Taat 12	Padang	750000	0
2002003	Wisnu	Jl Damai 11	Padang	2600000	5000000
2002004	Alif	Jl. Mulia 1	Padang	100000	12000000

## Fragmentasi Data

---

- Fragmentasi Horizontal

nasabah1 =  $\sigma_{kota='Medan'}$  (nasabah)

nasabah2 =  $\sigma_{kota='Padang'}$  (nasabah)

## Fragmentasi Data - Horizontal

---

- Akan diperoleh 2 buah fragemen
  - Fragmen Nasabah1
  - Fragmen Nasabah2
- Terapkan operasi union  
**nasabah = nasabah1  $\cup$  nasabah2**

## Fragmentasi Data

---

- Fragmentasi Vertikal

nasabah1 =  $\pi$ no\_nas, nama\_nas, alamat, kota (nasabah)

nasabah2 =  $\pi$ no\_nas, saldo\_simpan (nasabah)

nasabah3 =  $\pi$ no\_nas, saldo\_pinjam (nasabah)

## Fragmentasi Data

---

- **Replikasi dan Fragmentasi**

Upaya untuk replikasi dan fragmentasi data dapat dilakukan sekaligus pada sebuah tabel, yaitu dengan mereplikasi fragmen-fragmen yang telah terbentuk. Misalnya, dalam sebuah sistem terdistribusi terdapat 5 buah simpul S1, S2, S3, S4 dan S5.

Kita dapat memilah tabel nasabah sesuai dengan kota dimana nasabah mengajukan permohonan menjadi nasabah1, nasabah2 dan nasabah3.

Kemudian dengan pertimbangan tertentu, fragmen nasabah1 disimpan di simpul S1, S2 dan S3. lalu fragmen nasabah2 disimpan di simpul S3 dan S4 dan fragmen nasabah3 disimpan di simpul S1 dan S3.

## PEMROSESAN QUERY DALAM BASIS DATA TERDISTRIBUSI

---

- Dalam sistem terpusat, kriteria utama untuk mengetahui cost dari sebuah strategi query adalah jumlah/waktu akses ke disk.
- Faktor-faktor yang perlu dipertimbangkan:
  1. Biaya/waktu untuk transmisi data
  2. Potensi peningkatan karena adanya sejumlah simpul yang dapat melaksanakan query secara paralel.

## Transformasi Query

---

- Jika tabel telah direplikasi atau difragmentasi atau sekaligus direplikasi dan fragmentasi maka kita dapat memenuhi query tersebut dengan memilih salah satu simpul tempat suatu tabel berada kemudian mengeksekusi query.
- Jika tabel tidak direplikasi atau difragmentasi, pemilihan simpul akan didasarkan pada simpul yang memberikan ongkos transmisi data yang paling rendah.

## Transformasi Query

---

- Jika tabel difragmentasi dan ditempatkan di berbagai simpul yang berbeda, maka kita harus melakukan operasi Join atau Union untuk merekonstruksi isi seluruh ini disamping tergantung pada query, juga tergantung pada jenis fragmentasi yang diterapkan terhadap tabel yang terlibat dalam query.

## Transformasi Query

---

- Jika fragmentasi yang dilakukan horizontal maka operasi Union dapat dilakukan.
- Jika fragmentasi vertikal dan query menghendaki penayangan semua atribut maka operasi Natural Join yang harus digunakan.

## Contoh :

---

- Tabel Mahasiswa
- Ekspresi Standar dari query :  
 **$\sigma_{kota = 'Bandung'}(mahasiswa)$**
- Jika tabel mahasiswa difragmentasi secara horizontal di dua fragmen (diberi nama mahasiswa1 dan mahasiswa2) sehingga tabel mahasiswa sesungguhnya merupakan hasil operasi Union dari keduanya.

**$mahasiswa1 \cup mahasiswa2$**

## Contoh :

---

- Maka query di atas dapat kita translasi menjadi :  
 **$\sigma_{\text{kota}=\text{'Bandung'}}(\text{mahasiswa1} \cup \text{mahasiswa2})$**