ESTIMASI QUERY

Sistem Basis Data

Gentisya Tri Mardiani, S.Kom

Estimasi Query

- Optimizer query akan membuat informasi statistik yang tersimpan dalam katalog DBMS untuk memperkirakan besarnya biaya dari sebuah rencana query.

- Informasi yang tersimpan meliputi:
  - Banyaknya baris data (record) dalam sebuah tabel
  - Banyaknya blok berisi baris data dalam tabel
  - Ukuran setiap baris data dari tabel (satuan byte)
  - Banyaknya nilai unik dalam tabel untuk suatu atribut

Pengukuran Biaya Query

- Biaya evaluasi query dapat diukur dari banyaknya sumber daya (resource) sistem yang terpakai, meliputi pengaksesan disk, waktu CPU mengerjakan query, dan untuk sistem basis data paralel atau terdistribusi.

- Pada sistem basis data yang besar dapat memberikan kontribusi biaya query yang dominan.

Ekuivalensi alajabar relasional dan SQL

	Aljabar Relasional	**SQL**
Selection	$\sigma_P^{(E)}$ Contoh: $\sigma_{\text{kota='Bantul'}}^{(\text{Pribadi})}$	Select * from E where P Contoh: Select * from pribadi where kota='Bantul'
Projection	$\pi_{\text{column}}^{(E)}$ Contoh: $\pi_{\text{NIP, Nama}}^{(\text{Pribadi})}$	Select column from E Contoh: Select NIP, Nama from Pribadi

	Aljabar Relasional	**SQL**
Union	**E1 ∪ E2** Contoh: Pribadi ∪ Pekerjaan	select * from E1 union select * from E2 Contoh: Select * from pribadi union select * from pekerjaan
Set Difference	**E1 - E2** Contoh: $\pi_{NIP}(Pribadi) - \pi_{NIP}(Pekerjaan)$	Select * from E1 except select * from E2 Contoh: Select NIP from pribadi except select NIP from pekerjaan
Cartesian Product	**E1 x E2** Contoh: $\pi_{NIP, Nama, Gaji}(\sigma_{Pribadi.NIP=Pekerjaan.NIP}(Pribadi \times Pekerjaan))$	Select * from E1, E2 Contoh: Select Pribadi.NIP, Pribadi.Nama, Pekerjaan.Gaji from Pribadi, Pekerjaan where Pribadi.NIP = Pekerjaan.NIP;

	Aljabar Relasional	**SQL**
Set Intersection	**E1∩ E2** Contoh: Pribadi ∩ Pekerjaan	select * from E1 intersect select * from E2 Contoh: Select * from pribadi intersect select * from pekerjaan
Join	**E1 ∞ E2** Contoh: Pribadi $\infty_{\text{Pribadi.NIP=Pekerjaan.NIP}}$ Pekerjaan	Select * from E1 join E2 Contoh: Select * from pribadi join Pekerjaan on Pribadi.NIP=Pekerjaan. NIP

Ekivalensi Ekspresi Relasional

$$\pi_{\text{Nama}}(\sigma_{\text{nama\_bag}=\text{'Akunting'}}(\textbf{bagian} \infty \textbf{pekerjaan} \infty \textbf{pribadi}))$$

Karena atribut nama_bag itu terdapat di tabel bagian, maka dapat direduksi terlebih dahulu banyaknya baris data yang akan dilibatkan dalam operasi join, dengan mengubah ekspresi di atas menjadi ekspresi yang ekivalen:

$$\pi_{\text{Nama}}(\sigma_{\text{nama\_bag}=\text{'Akunting'}}(\textbf{bagian}) \infty (\textbf{pekerjaan} \infty \textbf{pribadi}))$$

Ekivalensi ekspresi operasi Seleksi

- Aturan:
  - Jalankan operasi seleksi seawal mungkin (prioritaskan operasi seleksi)
    contoh: $\pi_{\text{Nama}}(\sigma_{\text{nama\_bag}=\text{'Akunting'}}(\textbf{bagian}) \infty (\textbf{pekerjaan} \infty \textbf{pribadi}))$
  - Ganti ekspresi yang berbentuk
    $\sigma_{P1 \wedge P2}(E)$ menjadi $\sigma_{P1}(\sigma_{P2(E)})$

Ekivalensi ekspresi operasi natural join

- Memilih urutan operasi Join yang optimal, untuk semua relasi r1, r2, r3 maka:

$$(r1 \infty r2) \infty r3 \text{ menjadi } r1 \infty (r2 \infty r3)$$

meskipun ekspresi di atas sama, namun secara komputasi operasi bisa berbeda

Aturan Ekivalensi

1. Operasi seleksi konjungtif dapat direkonstruksi ke dalam sebuah sekuen seleksi individual

$$\sigma_{P1 \wedge P2}(E) = \sigma_{P1}(\sigma_{P2}^{(E)})$$

2. Operasi seleksi bersifat komutatif

$$\sigma_{P1}(\sigma_{P2}^{(E)}) = \sigma_{P2}(\sigma_{P1}^{(E)})$$

3. Hanya operasi final dalam sekuen operasi proyeksi yang diperlukan

$$\pi_{L1}(\pi_{L2}(E)) = \pi_{L1}(E)$$

4. Seleksi dapat dikombinasikan dengan cartesian product dan theta join

$$\sigma_{\theta}(E1 \times E2) = E1 \infty_{\theta} E2$$

Aturan Ekivalensi

5. Operasi theta join bersifat komutatif

$$E1 \infty_{\theta} E2 = E2 \infty_{\theta} E1$$

6. Operasi natural join berisfat asosiatif

$$(E1 \infty E2) \infty E3 = E1 \infty (E2 \infty E3)$$

7. Operasi union dan intersection bersifat komutatif

$$E1 \cup E2 = E2 \cup E1 , \quad E1 \cap E2 = E2 \cap E1$$

8. Operasi union dan intersection bersifat asosiatif

$$(E1 \cup E2) \cup E3 = E1 \cup (E2 \cup E3)$$
$$(E1 \cap E2) \cap E3 = E1 \cap (E2 \cap E3)$$

Aturan Ekivalensi

9. Operasi seleksi dapat didistribusikan ke operasi union, intersection, dan set difference

   $\sigma_P(E1 - E2) = \sigma_P(E1) - E2 = \sigma_P(E1) - \sigma_P(E2)$

10. Operasi proyeksi dapat didistribusikan ke operasi union

    $\pi_L(E1 \cup E2) = \pi_L(E1) \cup \pi_L(E2)$

Struktur Sistem Basis Data

- Tujuan utama dari sistem basis data adalah untuk memudahkan dan memfasilitasi akses ke data.
- Faktor utama yang menjadi parameter kepuasan user terhadap sistem basis data adalah performansinya.
- Performansi sistem tergantung pada:
  - Efisiensi struktur data (penyimpanan) yang digunakan/ dipilih
  - Seberapa efisien sistem tersebut dapat beroperasi pada struktur data tersebut

Struktur DBMS untuk Pemrosesan Query

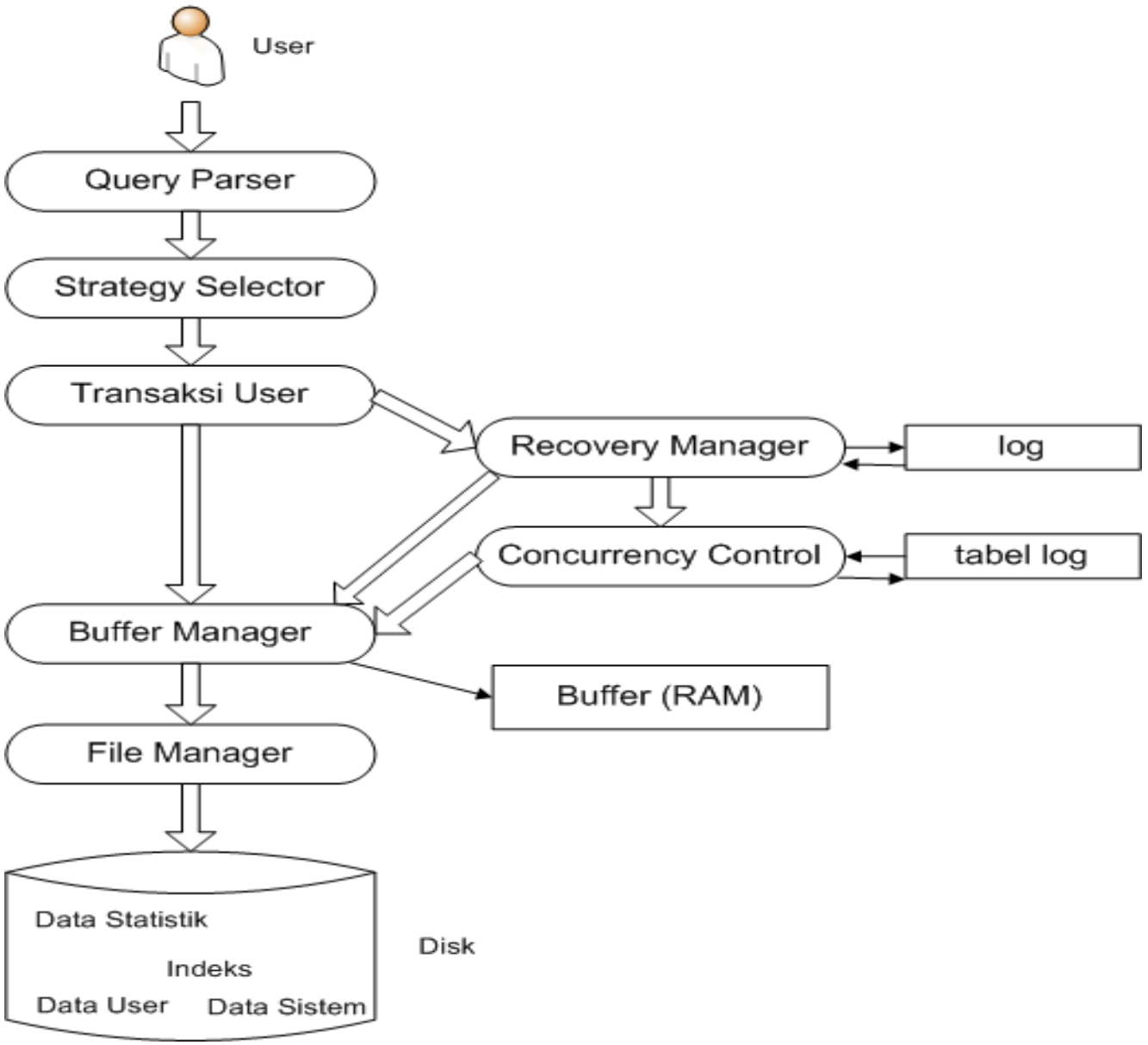
- ***File manager***, yang mengelola alokasi dalam disk dan struktur data yang digunakan untuk merepresentasikan informasi yang tersimpan dalam disk
- ***Buffer manager***, yang bertanggung jawab dalam pentransferan informasi antara disk dan memori utama
- ***Query parser***, yang menerjemahkan perintah dalam query language ke dalam bahasa mesin
- ***Strategy selector***, yang mentransformasikan permintaan user ke dalam bentuk lain yang sama tetapi lebih efisien, kemudian menentukan strategi terbaik untuk menjalankan query

Struktur DBMS untuk Pemrosesan Query

- ***Authorization / integrity manager***, yang memeriksa pemenuhan batasan- batasan integritas dan otoritas user untuk mengakses data
- ***Recovery manager***, yang menjamin bahwa basis data dapat tetap konsisten setelah kegagalan/ kerusakan sistem insidental
- ***Concurrecy controller***, yang menjamin interaksi pada basis data secara konkuren dilaksanakan tanpa adanya konflik antar user

Struktur data yang dibutuhkan dalam implementasi fisik

- **File Data**, yang merupakan basis data itu sendiri
- **File Data Sistem**, yang menyimpan informasi tentang struktur basis data, contoh isi file data sistem adalah kamus data
- **Data Statistik**, yang menyimpan informasi spesifik tentang data dalam basis data. Informasi ini bermanfaat bagi pemilihan strategi operasi yang diminta user


User
Query Parser
Strategy Selector
Transaksi User
Recovery Manager
log
Concurrency Control
tabel log
Buffer Manager
Buffer (RAM)
File Manager
Data Statistik
Indeks
Data User
Data Sistem
Disk