

Fella Attaqi

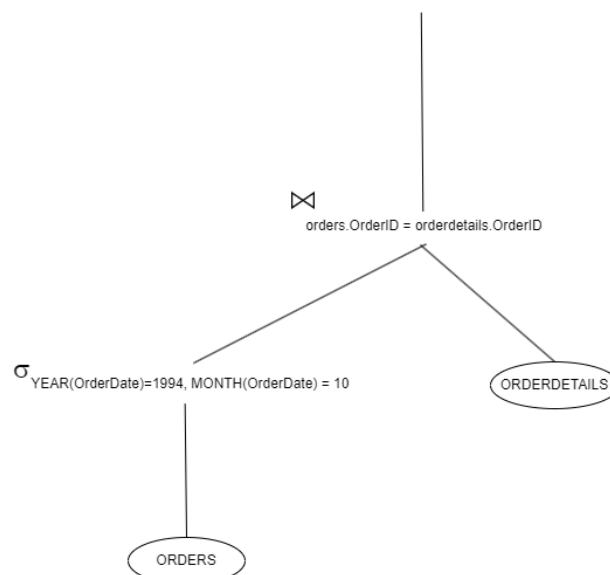
0102520051

Kuis BDL

### Bagian A

1. Yang harus dikerjakan oleh sistem agar optimum adalah melakukan filtrasi data penjualan yang hanya ada di bulan oktober tahun 1994 pada tabel order, kemudian melakukan join dengan tabel orderdetail, dan kemudian sistem menghitung data penjualan sesuai filter yg telah dilakukan dan akan menghasilkan output yg diinginkan.

$\pi$  MONTH(orders.OrderDate) AS 'bulan', YEAR(orders.OrderDate) AS 'tahun', SUM(orderdetails.UnitPrice \* Quantity \* (1 - orderdetails.Discount)) AS 'total\_penjualan'



2. Karena jika suatu saat terdapat perubahan data, canonical query tree tidak dapat secara otomatis menyesuaikan rencana eksekusi yang ingin diubah. sehingga dapat mengakibatkan perubahan dalam kualitas kinerja dan menyebabkan waktu eksekusi yang tidak optimal.

### Bagian B

1. Konsep atomicity menggambarkan transaksi yang tidak dapat dibagi-bagi atau terpecahkan. Jika semua operasi dalam transaksi berhasil dilaksanakan, maka perubahan-perubahan tersebut akan dikonfirmasi dan menjadi bagian dari basis data. Namun, jika terjadi kegagalan pada salah satu operasi di dalam transaksi, maka transaksi akan di-rollback dan semua perubahan yang dilakukan sebelumnya akan dibatalkan karena akan membuat basis data tidak konsisten, sehingga basis data tidak akan mencerminkan perubahan yang tidak konsisten.
2. Interleaving process dalam konsep pemrosesan transaksi yaitu di mana operasi dari beberapa transaksi yang berbeda dieksekusi secara bergantian atau saling

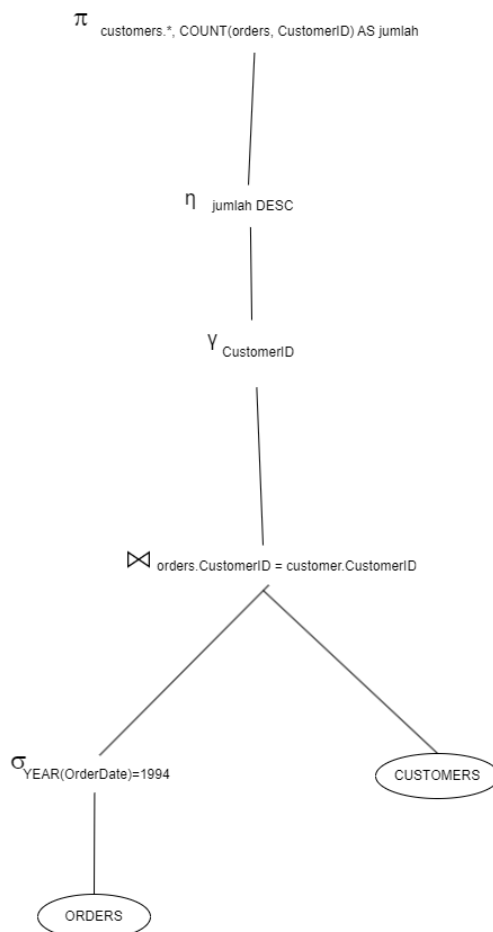
bergiliran. Dasar dari interleaving process dalam pemrosesan transaksi adalah untuk meningkatkan kinerja dan efisiensi sistem basis data.

### Bagian C

1. Pimpinan Perusahaan ingin melihat customer mana yang mendatangkan pendapatan tertinggi bagi perusahaan tersebut pada tahun 1994.
  - a. Buat SQL untuk mendapatkan data yang dibutuhkan. Tampilkan data customer dan diurutkan berdasarkan jumlah transaksi tertinggi.

```
SELECT customers.*, COUNT(orders.CustomerID) AS jumlah
FROM customers, orders
WHERE customers.CustomerID = orders.CustomerID
AND YEAR(orders.OrderDate) = 1994
GROUP BY orders.CustomerID
ORDER BY jumlah DESC;
```

- b. Sebutkan 2 customer yang mendatangkan pendapatan tertinggi sesuai hasil perintah SQL yang Anda buat.  
Horst Kloss dan Paula Wilson
    - c. Buat query tree-nya.



- d. Buat relational algebra sesuai dengan query tree yang Anda buat.

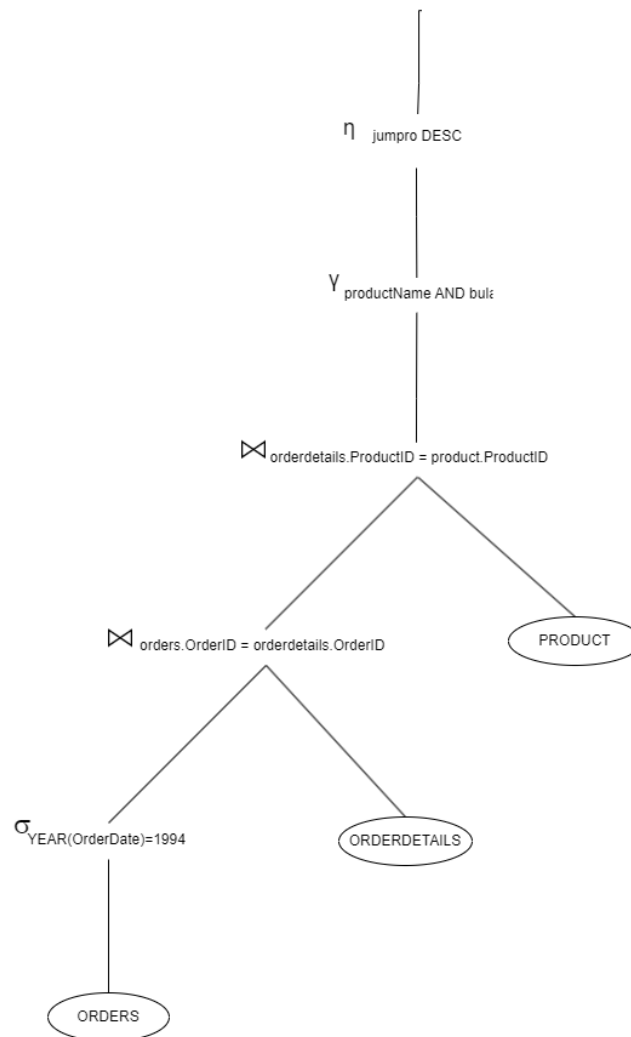
$\pi_{\text{customers.*}, \text{COUNT}(\text{orders.CustomerID}) \text{ AS jumlah}} (\gamma_{\text{CustomerID} \uparrow \text{jumlah DESC}} ((\sigma_{\text{YEAR}(\text{OrderDate}) = 1994} (\text{ORDER}))) \bowtie_{\text{customerID} = \text{customerID}} (\text{CUSTOMER}))$

2. Pimpinan Perusahaan ingin melihat produk apa yang paling banyak mendatangkan penghasilan bagi perusahaan setiap bulannya selama tahun 1994. Tampilkan data dengan atribut | Bulan | Nama Produk | Jumlah Produk | Jumlah Harga | Tampilkan data berurutan dari jumlah harga dan jumlah produk terjual tertinggi setiap bulannya (dari Januari hingga Desember 1994).
  - a. Buat SQL untuk mendapatkan data yang dibutuhkan. Tampilkan nama produk, jumlah dan total harga produk terjual dan diurutkan berdasarkan jumlah produk terjual terbanyak.

```
SELECT MONTH(orders.OrderDate) AS bulan, products.ProductName AS
nama_produk, COUNT(orderdetails.ProductID) AS jumlah_produk,
SUM(orderdetails.UnitPrice * orderdetails.Quantity) AS jumlah_harga
FROM orders, orderdetails, products
WHERE orders.OrderID = orderdetails.OrderID
AND orderdetails.ProductID = products.ProductID
AND YEAR(orders.OrderDate) = 1994
GROUP BY products.ProductName, bulan
ORDER BY jumlah_produk DESC;
```

- b. Sebutkan dua nama produk yang paling banyak mendatangkan penghasilan di bulan September 1994, dari hasil perintah SQL yang Anda buat di Soal 2a.  
Guaraná Fantástica dan Camembert Pierrot
  - c. Buat query tree-nya.

$\pi_{\text{MONTH}(\text{orders.OrderDate}) \text{ AS bulan, products.ProductName, COUNT}(\text{orderdetails.ProductID}) \text{ AS jumpro, SUM}(\text{orderdetails.UnitPrice} * \text{orderdetails.Quantity}) \text{ AS jumhar}}$



d. Buat relational algebra sesuai dengan query tree yang Anda buat.

$\pi_{\text{MONTH}(\text{orders.OrderDate}) \text{ AS bulan, products.ProductName, COUNT}(\text{orderdetails.ProductID}) \text{ AS jumpro, SUM}(\text{orderdetails.UnitPrice} * \text{orderdetails.Quantity}) \text{ AS jumhar}} (\gamma_{\text{productName AND bulan}} (\eta_{\text{jumpro DESC}} (((\sigma_{\text{YEAR}(\text{OrderDate}) = 1994} (\text{ORDER})) \bowtie_{\text{orderID} = \text{orderID}} (\text{ORDERDETAILL})) \bowtie_{\text{productID} = \text{productID}} (\text{PRODUK}))))$

## Bagian D

1. Terdapat 3 transaksi sebagai berikut:

T1:  $r1(x)$ ,  $r1(q)$ ,  $\text{add1} (x := x+q)$ ,  $w1(x)$

T2:  $r2(p)$ ,  $\text{display2} (p)$ ,  $r2(x)$ ,  $\text{multiply2} (x := x*2)$ ,  $w2(x)$

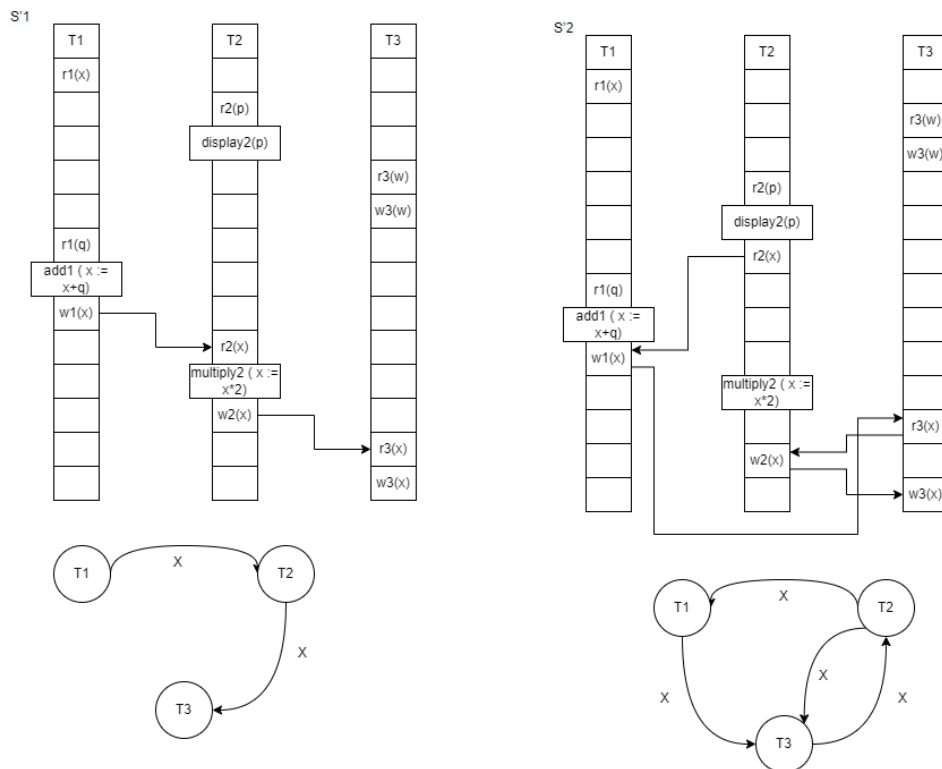
T3:  $r3(w)$ ,  $w3(w)$ ,  $r3(x)$ ,  $w3(x)$

Terdapat S' sebagai berikut:

S'1:  $r1(x)$ ,  $r2(p)$ ,  $\text{display2} (p)$ ,  $r3(w)$ ,  $w3(w)$ ,  $r1(q)$ ,  $\text{add1} (x := x+q)$ ,  $w1(x)$ ,  $r2(x)$ ,  $\text{multiply2} (x := x*2)$ ,  $w2(x)$ ,  $r3(x)$ ,  $w3(x)$

S'2:  $r1(x)$ ,  $r3(w)$ ,  $w3(w)$ ,  $r2(p)$ ,  $\text{display2} (p)$ ,  $r2(x)$ ,  $r1(q)$ ,  $\text{add1} (x := x+q)$ ,  $w1(x)$ ,  $\text{multiply2} (x := x*2)$ ,  $r3(x)$ ,  $w2(x)$ ,  $w3(x)$

a. Analisis keduanya dengan precedence diagram.



b. Simpulkan hasil analisis Anda dan jelaskan alasan kesimpulan tersebut.

Pada S'1 tidak terjadi konflik maka merupakan serializability sementara di S'2 terjadi cycle maka merupakan non serializability

2. Terdapat 4 transaksi sebagai berikut:

T1: r1(a), r1(b), r1(c), r1(d), w1(b), w1(c)

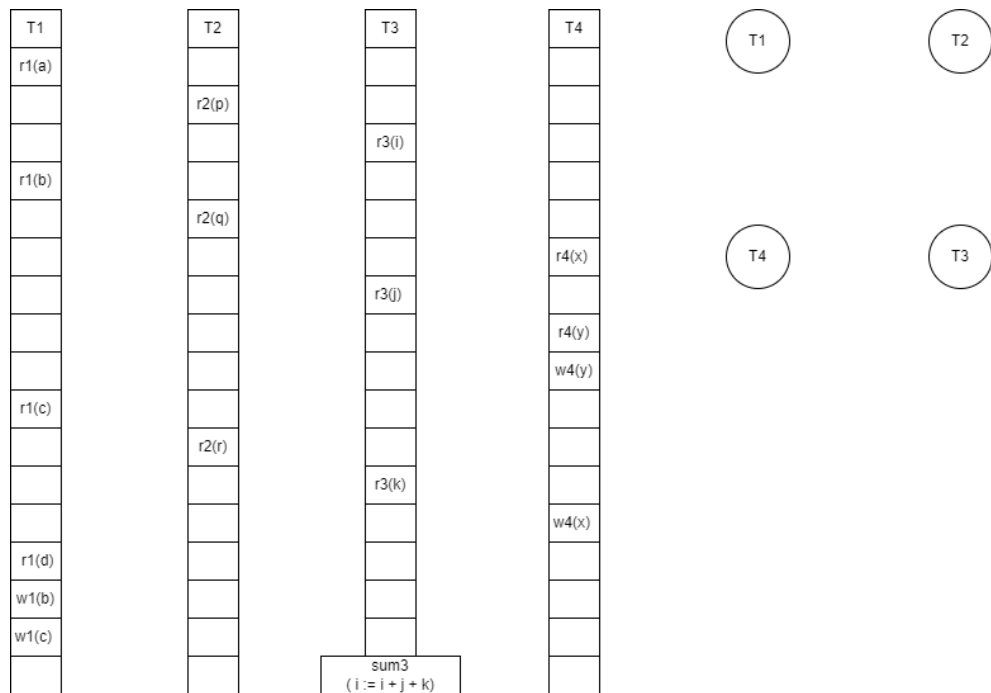
T2: r2(p), r2(q), r2(r) T3: r3(i), r3(j), r3(k), sum3 (i := i + j + k)

T4: r4(x), r4(y), w4(y), w4(x)

Terdapat S1' sebagai berikut:

S1': r1(a), r2(p), r3(i), r1(b), r2(q), r4(x), r3(j), r4(y), w4(y), r1(c), r2(r), r3(k), w4(x), r1(d), w1(b), w1(c), sum3 (i := i + j + k)

a. Analisis dengan conflict serializability dan simpulkan beserta penjelasan singkat.



Schedule S1' merupakan serializability karena di dalam precedence graph tidak terdapat cycle

- b. Analisis dengan view serializability dan simpulkan beserta penjelasan singkat.

S1=T1->T2->T3->T4

Schedule S1' merupakan view serializability karena memenuhi ke 4 syarat

T1: r1(a), r1(b), r1(c), r1(d), w1(b), w1(c)

T2: r2(p), r2(q), r2(r) T3: r3(i), r3(j), r3(k), sum3 (i := i + j + k), w3(i)

T4: r4(a), w4(a), r4(p), w4(p), r4(k), w4(k)

Terdapat S2' sebagai berikut:

S2': r1(a), r2(p), r3(i), r1(b), r2(q), r4(a), w4(a), r3(j), r4(p), w4(p), r1(c), r2(r), r3(k), r4(k), w4(k), r1(d), w1(b), w1(c), sum3 ( i := i + j + k ), w3(i)

- c. Analisis dengan conflict serializability dan simpulkan beserta penjelasan singkat.

d. Analisis dengan view serializability dan simpulkan beserta penjelasan singkat.  
 $S_2 = T_1 \rightarrow T_2 \rightarrow T_3 \rightarrow T_4$   
 Schedule  $S_2'$  merupakan view serializability karena memenuhi ke 4 syarat