

Sistem Terdisreibusi



Koordinasi Proses Dalam Sistem Terdistribusi

TKB6252 – Sistem Terdistribusi

Chalifa Chazar

www.script.id

chalifa.chazar@gmail.com

Pendahuluan

- Jika ada lebih dari satu proses yang siap running, maka sistem operasi akan menentukan salah satu proses untuk running terlebih dahulu.
- Oleh karena itu penjadwalan yang baik dibutuhkan dalam manajemen proses.

Kriteria Manajemen Proses

- Fairness
 - Memastikan setiap proses mendapat giliran yang sama dan adil
- Efisiensi
 - Penggunaan waktu CPU seoptimal mungkin
- Respon time
 - Mempercepat waktu tanggap dengan pemakai secara interaktif
- Turn around time
 - Meminimalkan waktu tunggu dari mulai proses datang sampai proses selesai running
- Through out
 - Memaksimalkan jumlah job yang dapat diproses persatuan waktu

- Dalam koordinasi antar proses dalam sistem terdistribusi sistem bisa deadlock
- Sistem deadlock adalah suatu kondisi dimana 2 proses atau lebih tidak dapat meneruskan eksekusinya
- Penyebab deadlock antara lain:
 - Mutual Exclusion
 - Hold & Wait
 - No Preemption
 - Circular Wait

Penyebab Deadlock

- Mutual Exclusion
 - Jika proses telah menggunakan suatu resource, maka tidak boleh ada proses lain yang menggunakan resource tsb
- Hold & Wait
 - Pada saat suatu proses sedang mengakses suatu resource, proses tsb dapat meminta ijin untuk mengakses resource lain
- No Preemption
 - Jika suatu proses meminta ijin untuk mengakses resource, sementara resource tsb dapat dibatalkan
- Circular Wait
 - Jika proses P0 sedang mengakses resource R1 dan minta ijin untuk mengakses resource R2 di mana pada saat bersamaan P1 sedang mengakses resource R0 dan meminta ijin untuk mengakses R1

Event Ordering

- Manajemen koordinasi antar proses pada sistem terdistribusi berbeda pada sistem tersentralisasi
- Beberapa parameter yang perlu diperhatikan antara lain:
 - Memori & clock tidak tunggal
 - Tidak mungkin menyatakan urutan dua kejadian
 - Hanya dapat ditentukan partial ordering (urutan bagian): relasi Happened-Before

Relasi Happened-Before

- Relasi happned-before adalah suatu relasi yang memperhatikan urutan kejadian, karena kejadian yang satu dengan yang lain saling terkait
- Pada relasi happened-before berlaku:
 - Proses-proses sekuensial
 - Hukum sebab-akibat
 - Simbol relasi happened-before adalah simbol panah “→”

Relasi Happened-Before

- Jika kejadian A & B berada dalam satu proses dan A dieksekusi sebelum B, maka ditulis $A \rightarrow B$
- Jika A adalah kejadian pengiriman pesan (sending) oleh sebuah proses dan B adalah sebuah kejadian penerima pesan (receiving) oleh proses lain, maka ditulis $A \rightarrow B$ (sending dilakukan sebelum receiving)
- Jika $A \rightarrow B$ dan $B \rightarrow C$, maka $A \rightarrow C$
- Karena suatu event tidak dapat terjadi sebelum dirinya sendiri terjadi, maka relasi happened-before tidak bersifat refleksi (dipasangkan ke dirinya sendiri)

- Jika 2 event A & B tidak memenuhi relasi $\rightarrow R$, maka kedua event tersebut dieksekusi bersamaan (concurrent)
- Konkurensi & happened-before dijelaskan dengan menggunakan diagram space-time

Diagram Space-Time

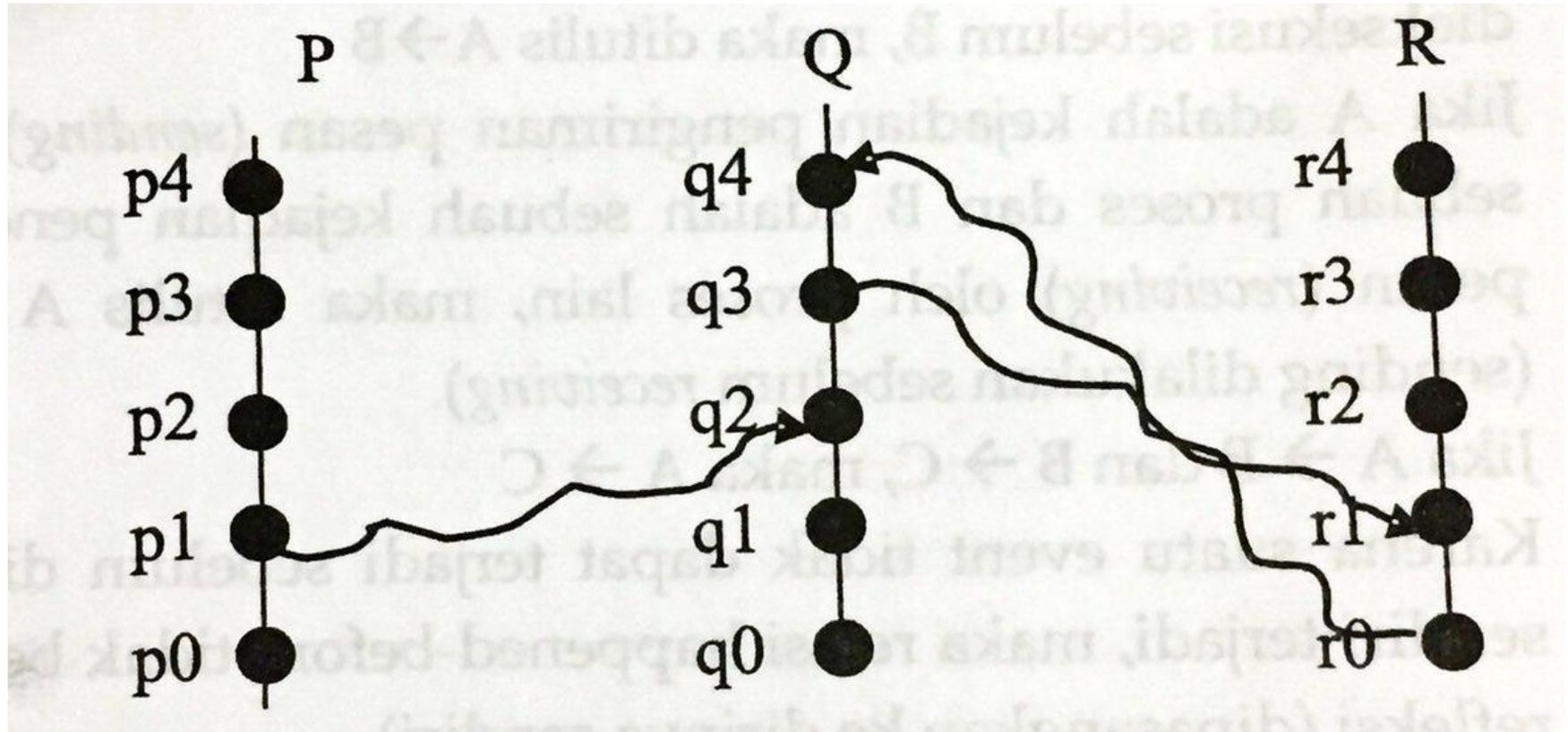
- Arah horizontal menyatakan space (proses berbeda)
- Arah vertikal menyatakan time
- Garis vertikal menunjukkan processor (huruf besar) dan lingkaran kecil menunjukkan event (huruf kecil)
- Garis gelombang menunjukkan pengiriman pesan/lintasan (path)
- Untuk event konkuren tidak ada path antara keduanya

Diagram Space-Time

Contoh:

- Sejumlah event direlasikan dengan happened-before: $p1 \rightarrow q2$, $r0 \rightarrow q4$, $q3 \rightarrow r1$, $p1 \rightarrow q4$
- Sejumlah event konkuren dari sistem adalah $q0$ dan $p2$, $r0$ dan $q3$, $r0$ dan $p3$, $q3$ dan $p3$
- Event p pada processor P , q pada Q , dan r pada R

Diagram Space-Time



- Untuk menentukan event A terjadi sebelum event B diperlukan sebuah clock umum atau suatu set pensinkron clock yang perfect
- Dalam manajemen koordinasi proses pada sistem terdistribusi kedua hal tsb tidak mungkin dilakukan
- Sehingga diperlukan definisi khusus dari relasi happened-before tanpa menggunakan clock physics

Timestamp

- Untuk manajemen koordinasi antar proses pada sistem terdistribusi dikenal konsep Timestamp
- Timestamp untuk suatu event adalah nilai logikal clock untuk event tsb
- Dimana setiap event sistem diasosiasikan dengan timestamp, kemudian didefinisikan Global Ordering Requirement (GOR)
- Untuk setiap pasangan event A & B berlaku, jika $A \rightarrow B$ maka $\text{timestamp } A < \text{timestamp } B$

Contoh Timestamp

- Mula-mula definisikan dalam tiap proses P_i suatu logical clock (LC_i)
- Dimana LC_i dapat diimplementasikan sebagai counter sederhana yang di-increment-kan antara 2 event tereksekusi berurutan dalam suatu proses
- LC_i dapat menunjukkan nomor event secara unik, sehingga jika event terjadi sebelum event B dalam proses P_i , maka $LC_i(A) < LC_i(B)$

Contoh Timestamp

- Misalnya ada 2 proses P1 dan P2 yang sedang berkomunikasi:
 - P1 mengirim pesan ke P2 dan L_{Ci} event A = 200
 - P2 menerima pesan dan L_{Ci} event B = 192
 - Diasumsikan proses P2 lebih lambat dari proses P1
- Pada contoh tsb dapat mengacaukan GOR, karena jika $A \rightarrow B$, maka $L_{Ci}(A) < L_{Ci}(B)$

Contoh Timestamp

- Oleh karena itu solusi yang diperlukan adalah proses meng-advance LC_i -nya jika menerima pesan dengan timestamp lebih besar dari LC_i penerima
- Cara: jika proses P_i menerima pesan (event B) dengan timestamp = t dan $LC_i(B) \leq t$, maka proses harus menaikkan clock, yaitu sebesar:

$$LC_i(B) = t + 1$$

Mutual Exclusion

- Adalah suatu kondisi dimana sumber daya (resource) tidak dapat dibagi ke penggunaanya
- Mutual exclusion dapat terjadi karena:
 - Tidak ada dua proses yang pada saat bersamaan berada dalam critical section
 - Tidak ada proses yang berjalan di luar critical section yang dapat menghambat proses lain
 - Tidak ada proses yang tidak dapat masuk ke dalam critical section
 - Kondisi tersebut diatasi dengan menggunakan beberapa macam algoritma

Algoritma pada Mutual Exclusion

- Centralized Distributed Approach (CDA)
- Fully Distributed Approach (FDA)
- Token Parsing Approach (TPA)

Tugas

- Cari tau maksud dari algoritma berikut:
 - Centralized Distributed Approach (CDA)
 - Fully Distributed Approach (FDA)
 - Token Parsing Approach (TPA)



</TERIMA KASIH>

Chalifa Chazar, S.T, M.T

Email: chalifa.chazar@gmail.com

script.id

Copyright @2016