

Sistem Operasi

DEADLOCK

Hendri Sopryadi, S.Kom.

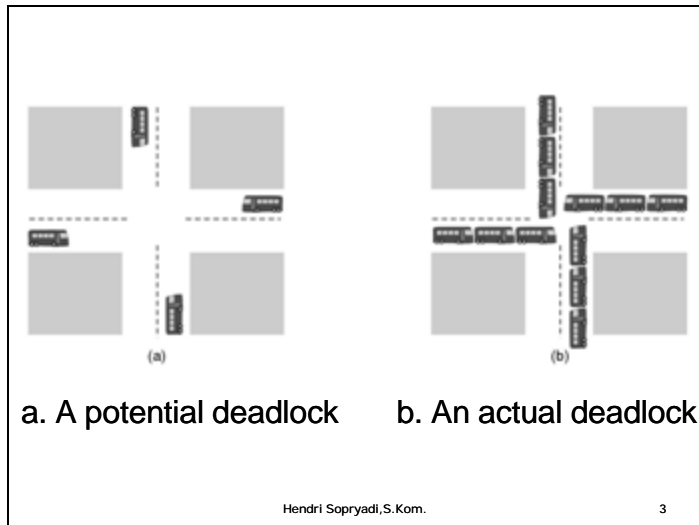
1

Deadlock

- ◆ Kondisi saling menunggu antara 2 proses/lebih yang diakibatkan oleh suatu event (fasilitas) yang ditunggu-tunggu tidak kunjung tiba

Hendri Sopryadi, S.Kom.

2



Hendri Sopryadi, S.Kom.

3

Contoh lain :

Hendri Sopryadi, S.Kom.

4

Keterangan

Hendri Sopryadi, S.Kom.

5

Resources

◆ **Preemptable**

Resource yang bisa digunakan secara simultan/bersama-sama secara bergantian tanpa menyebabkan kesalahan/error

Ex. Memory

◆ **Non preemptable**

- Resource yang tidak dapat digunakan secara bersama-sama harus secara bergantian
- Sekali resource digunakan harus di hold(ditunggu) hingga selesai
- Jika digunakan bersama, kemungkinan terjadi kesalahan /error
- Ex. Printer

Hendri Sopryadi, S.Kom.

6

Kondisi Deadlock

◆ **Kondisi Mutual Exclusion**

- Jika suatu proses telah menggunakan suatu resource maka tidak boleh ada proses lain yang menggunakan resource tersebut

Hendri Sopryadi, S.Kom.

7

◆ **Kondisi Hold and Wait**

- Pada saat suatu proses sedang mengakses suatu resource, proses tersebut dapat meminta izin untuk mengakses resource lain

Hendri Sopryadi, S.Kom.

8

◆ **Kondisi No Preemption**

- Jika suatu proses meminta ijin untuk mengakses resource sementara resource tersebut tidak tersedia, maka permintaan ijin tersebut tidak dapat diabaikan sehingga terjadi proses menunggu
- Contoh : kondisi "waiting for 5 second" pada printer

Hendri Sopryadi, S.Kom.

9

◆ **Kondisi Circular Wait**

- Jika proses P_0 sedang mengakses R_1 dan meminta ijin untuk mengakses R_2 sedang pada saat bersamaan P_1 juga sedang mengakses R_2 dan meminta ijin untuk mengakses R_1

Hendri Sopryadi, S.Kom.

10

Contoh :

Hendri Sopryadi, S.Kom.

11

4 strategi mengatasi deadlock

- ◆ **Ignore**
- ◆ **Deadlock prevention**
- ◆ **Deadlock avoidance**
- ◆ **Deadlock detection and recovery**

Hendri Sopryadi, S.Kom.

12

Ignore

- ◆ Membiarkan kejadian terjadinya deadlock dengan asumsi tidak akan terulang kembali

Hendri Sopryadi, S.Kom.

13

Pencegahan deadlock *Deadlock prevention*

Meniadakan 4 penyebab deadlock :

- ◆ Meniadakan *Mutual Exclusion (Mutex)*
- ◆ Meniadakan syarat *hold and wait*
- ◆ Meniadakan kondisi *Non Preemption*
- ◆ Meniadakan kondisi menunggu sirkular (*circular wait*)

Hendri Sopryadi, S.Kom.

14

Penghindaran deadlock *Deadlock avoidance*

- ◆ Tujuannya menghindari kondisi-kondisi yang paling mungkin menimbulkan deadlock agar memperoleh utilisasi sumber daya lebih baik, dengan
 - ***Safe state***
 - ***Unsafe state***

Hendri Sopryadi, S.Kom.

15

Metode deteksi dan pemulihan *Deadlock detection and recovery*

- ◆ **Metode deteksi** digunakan pada sistem yang mengizinkan terjadinya *deadlock* dengan menentukan proses dan *resource* yang terlibat *deadlock* secara presisi
- ◆ Contoh : mendeteksi status proses dengan ***window task manager***

Hendri Sopryadi, S.Kom.

16

- ◆ **Metode pemulihan** dengan cara proses-proses yang terlibat *deadlock* secepatnya menyelesaikan eksekusi dan membebaskan *resource* yang sedang dipakai
- ◆ Contoh : menghentikan proses 'deadlock' dengan ***End Task***