

Sistem Operasi

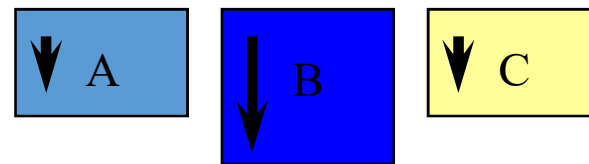
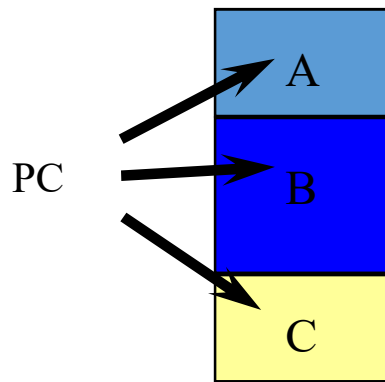
Proses (Process)

michrandi@telkomuniversity.ac.id

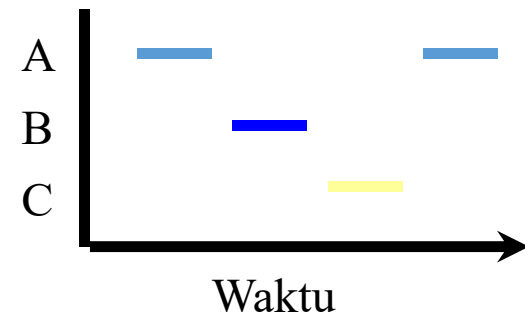
2016

Overview

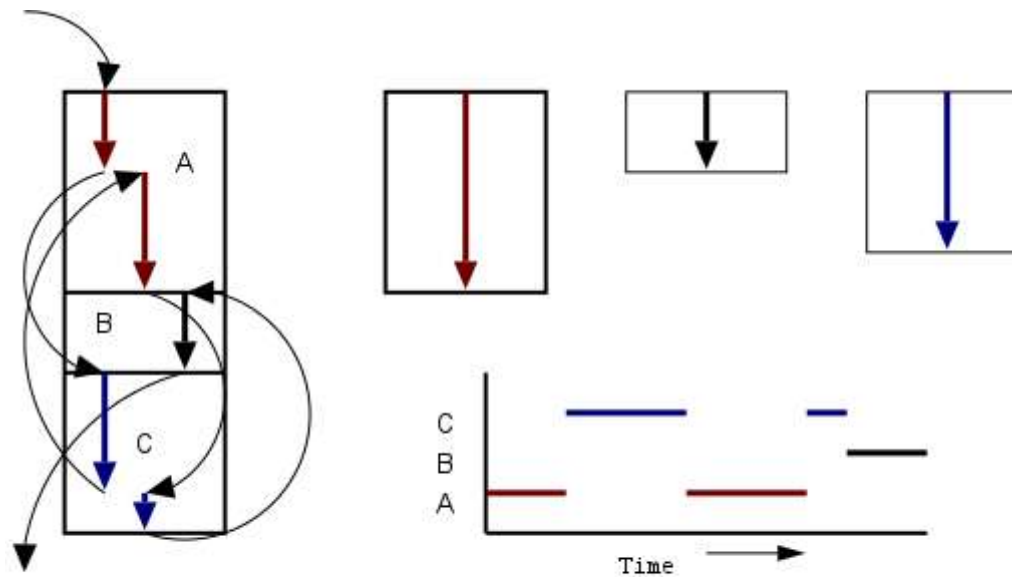
- Proses adalah sebuah program yang sedang dieksekusi
- OS Multiprogram mendukung adanya banyak proses
 - Pseudo parallelism
 - Concurrent (bersamaan)



Gambaran Konseptual



Overview-Adv.Look



Three views of multiprogramming with three processes:
The real program counter; each virtual program counter;
time history of process execution. (We assume no jumps.)

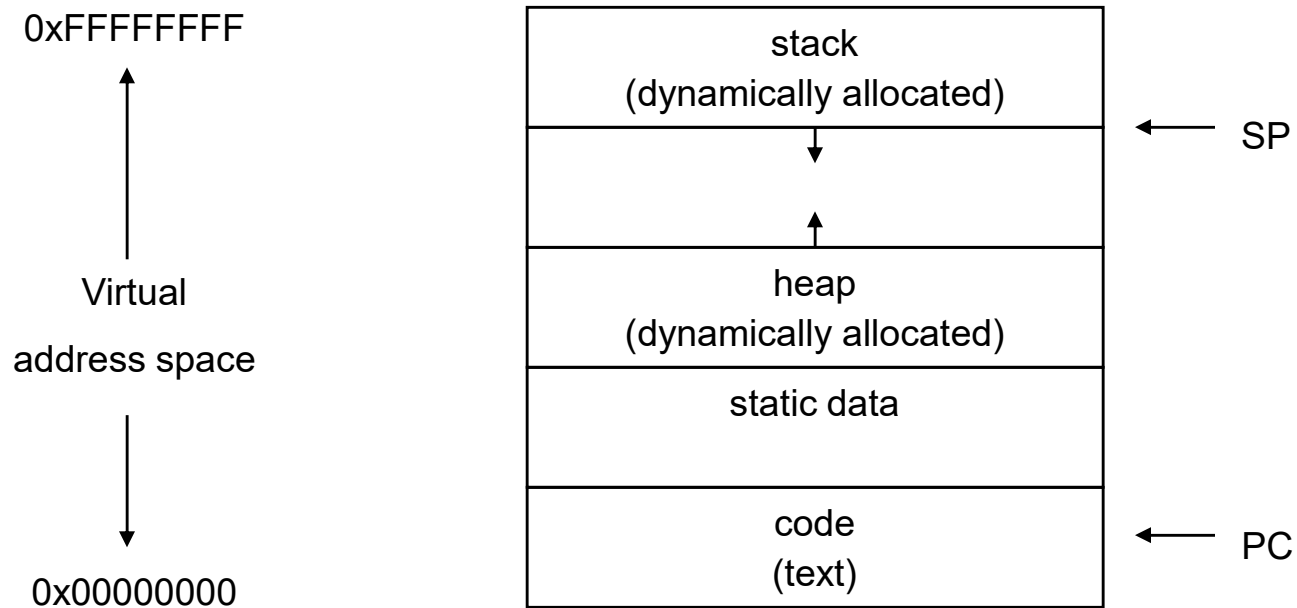
Definisi

- Proses adalah abstraksi OS untuk eksekusi program
 - Unit eksekusi
 - Sebuah unit penjadwalan
 - Kontek eksekusi dinamis
 - Sebuah program adalah item statik – satu atau lebih file
- Proses disebut juga dengan istilah **job**, **task**, atau **proses sekuensial**

Isi Proses

- Sebuah proses terdiri atas (paling tidak satu buah):
 - Sebuah ruang alamat – biasanya terproteksi dan virtual – dipetakan ke memori fisik
 - Code untuk program yang berjalan
 - data untuk program yang berjalan
 - Sebuah stack eksekusi dan stack pointer (SP)
 - program counter (PC)
 - Satu set register prosesor - general purpose dan status
 - Satu set sumber daya sistem
 - file, koneksi jaringan, privilege, ...

Proses – Ruang Alamat



Proses di OS - Representasi

- Untuk pemakai (dan proses lain): sebuah proses dikenali melalui identitas yang disebut **Process ID (PID)**
- Didalam OS, proses dinyatakan sebagai masukan data dalam sebuah **Process Table (PT)**
 - PID menunjuk ke sebuah masukan PT
 - Masukan PT = **Process Control Block (PCB)**
- PCB adalah struktur data (pada kernel OS) yang sangat besar berisi atau menunjuk semua informasi yang mengenai proses-proses yang ada.
 - Linux – didefinisikan di **task_struct** : *lihat* `include/linux/sched.h` – lebih dari 70 field
 - NT – didefinisikan di **EPROCESS** – sekitar 60 field

Process Table

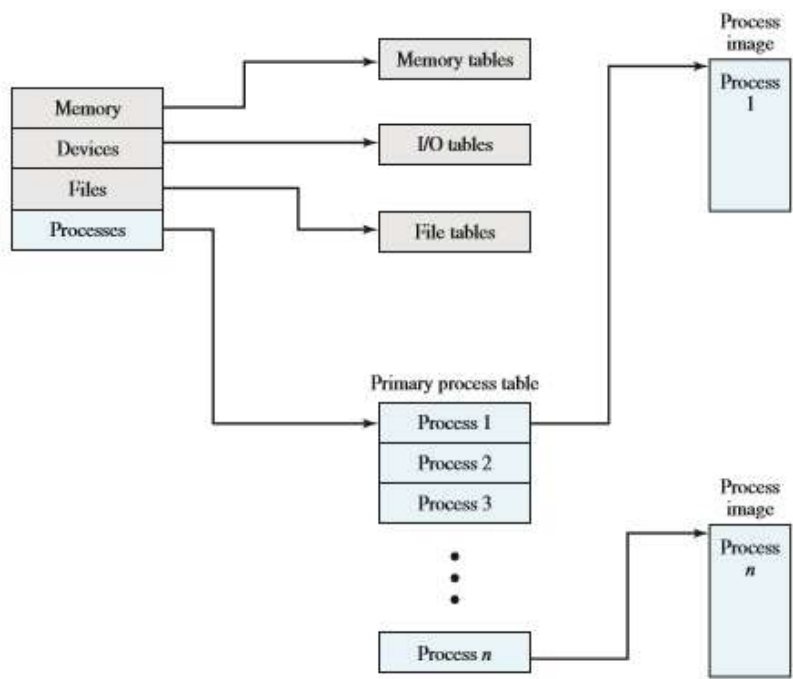
- Struktur data yang melakukan pengaturan pada sistem operasi yang memfasilitasi switching, scheduling, dan aktifitas lainnya.
- Tabel tersebut biasa disebut **context block**.
- Blok ini berisikan informasi mengenai proses, seperti nama dan status proses, prioritas, register-register, dan sinyal-sinyal yang masih diproses.
- Isi dari context block tergantung dari sistem operasi yang digunakan.
- Sebagai contoh, jika sistem operasi yang digunakan mendukung paging, maka context block dapat berisikan sebuah entry dari page table.

Proses di OS - PCB

- PCB umumnya berisikan:
 - Kondisi eksekusi
 - Register data, PC dan SP – disimpan ketika proses tidak aktif
 - Informasi manajemen memori
 - Informasi hak akses (privilege) dan pemakai
 - Prioritas penjadwalan
 - Informasi sumber daya
 - Informasi akuntansi

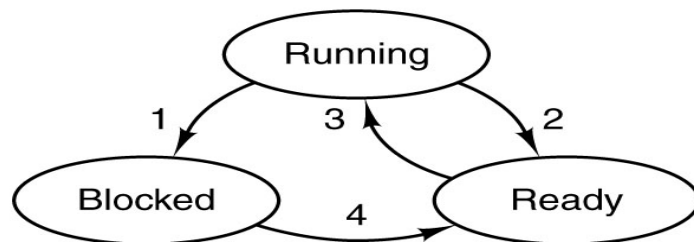
Identifier
State
Priority
Program Counter
Memory Pointers
Context Data
I/O Status Information
Accounting Information
• • •

Tabel Kontrol OS



Proses - State

- Proses mempunyai state eksekusi
 - ready: menunggu untuk dijalankan CPU
 - running: eksekusi di CPU
 - waiting: menunggu sebuah event, misal: I/O



1. Process blocks for input
2. Scheduler picks another process
3. Scheduler picks this process
4. Input becomes available

Proses-contoh

- Dua proses pemakai (PID 101 and 102)
 - Mempunyai prioritas sama
- Proses file server prioritas tinggi (PID 5)

Proses-contoh

Events/action	101	102	5
Initial Stats	Run	Ready	Block
101 wait keyboard input	Block	Run	Block
Request from file sever	Block	Run	Ready
	Block	Ready	Run
File server handled req	Block	Run	Block
Keyboard input ready	Ready	Run	Block
	Run	Ready	Block
Clock interrupt, timeslice	Ready	Run	Block

Proses-contoh

- Suatu sistem terdiri atas:
 - Dispatcher – mulai alamat 100
 - Proses A – mulai alamat 5000
 - Proses B – mulai alamat 8000
 - Memulai operasi I/O setelah 4 instruksi
 - Proses C – mulai alamat 12000
- Asumsikan dapat dijalankan 6 instruksi sebelum terjadi timer interrupt
- Asumsikan semua proses berada di memori
- Asumsikan dispatcher memerlukan 6 instruksi

1	5000		27	12004
2	5001		28	12005
3	5002		– Timeout –	
4	5003	A	29	100
5	5004		30	101
6	5005		31	102
– Timeout –			32	103
7	100		33	104
8	101		34	105
9	102		35	5006
10	103	Ds	36	5007
11	104		37	5008
12	105		38	5009
13	8000		39	5010
14	8001	B	40	5011
15	8002		– Timeout –	
16	8003		41	100
– Start I/O –			42	101
17	100		43	102
18	101		44	103
19	102		45	104
20	103		46	105
21	104		47	12006
22	105		48	12007
23	12000		49	12008
24	12001	C	50	12009
25	12002		51	12010
26	12003		52	12011
			– Timeout –	

Process Creation

- Submission of a batch job
- User logs on
- Created to provide a service such as printing
- Process creates another process

Process Termination

- Batch job issues *Halt* instruction
- User logs off
- Quit an application
- Error and fault conditions

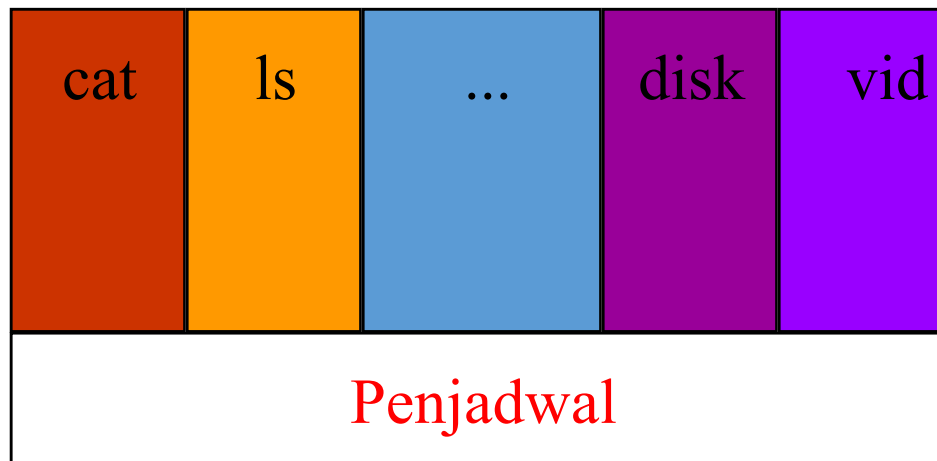
Reasons for Process Termination

- I/O failure
- Invalid instruction
 - happens when try to execute data
- Privileged instruction
- Data misuse
- Operating system intervention
 - such as when deadlock occurs
- Parent terminates so child processes terminate
- Parent request
- Normal completion
- Time limit exceeded
- Memory unavailable
- Processes
- Bounds violation
- Protection error
 - example write to read-only file
- Arithmetic error
- Time overrun
 - process waited longer than a specified maximum for an event

Proses – State Antrian

- OS memelihara sekumpulan antrian *state proses*
 - Umumnya satu antrian untuk setiap state – misal: ready, waiting, ...
 - Setiap PCB disimpan ke antrian state tergantung statenya saat ini
 - Saat sebuah proses berubah state, PCBnya ditarik dari satu antrian dan ditaruh diantrian lain
- State proses dan antriannya berubah sebagai respon ke suatu event - interrupt, timeout

Penjadwal Proses



- Semua layanan adalah proses
- Ada penjadwal menangani interrupt, menghentikan dan memulai proses

Proses - Switching

- Ketika sebuah proses berjalan, state hardware berada di CPU : PC, SP, register proses
- Ketika OS men-suspend sebuah proses, state hardwarenya disimpan di PCB
- **Context switch** adalah kegiatan CPU men-switch dari satu proses ke proses lain

Context Switch

- Tanpa overhead tambahan (Pure overhead)
- Sangat ... cepat.
 - Umumnya 1 sampai 1000 mikrodetik
- Memerlukan hardware khusus untuk mempercepat
- Real-Time menggunakan “worse case”
- Pengambilan keputusan kapan melakukan switch contexts ke proses lain disebut *process scheduling*

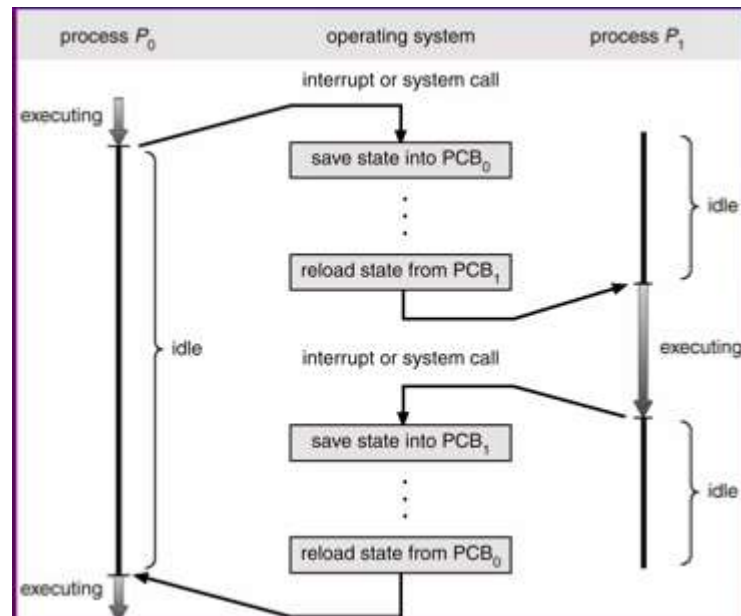
When to Switch a Process

- Clock interrupt □
 - process has executed for the maximum allowable time slice □ □
- I/O interrupt
- Memory fault
 - memory address is in virtual memory so it must be brought into main memory
- Trap
 - error occurred
 - may cause process to be moved to Exit state
- Supervisor call
 - such as file open

Pengolahan Interrupt

1. Devais I/O mengeluarkan sinyal interrupt
2. Proses saat ini (misalnya A), menyelesaikan instruksi yang sedang dikerjakan
3. CPU meng-acknowledge interrupt dan membuang sinyal interrupt tersebut
4. Hardware mendorong PC(IP) dan register status dari proses A ke system stack (cepat dan sederhana)
5. Hardware memuat PC baru dari rutin interrupt-handler (misal B)
6. B meneruskan menyimpan informasi A ke system stack (Sama untuk semua rutin interrupt-handler)
7. B dapat memproses interrupt
8. B mengembalikan informasi A dari system stack
9. Isi PC(IP) dan register status A diambil kembali dan proses A diteruskan

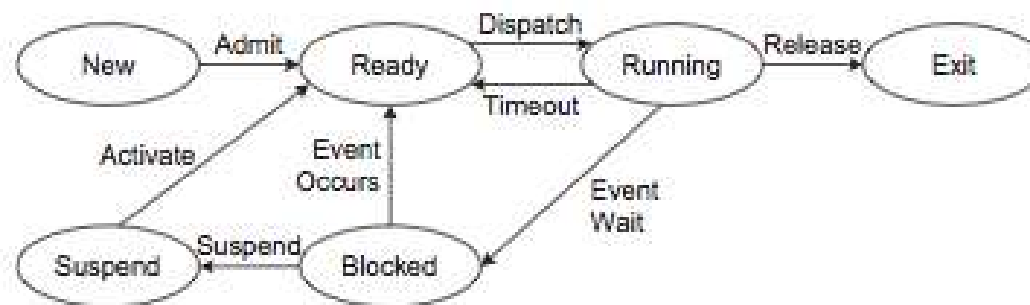
Interrupt



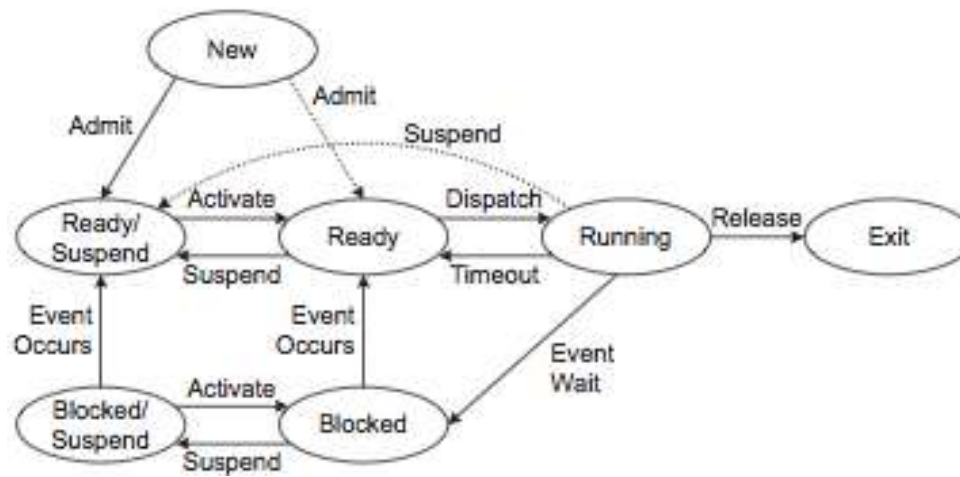
Suspended Processes

- Processor is faster than I/O so all processes could be waiting for I/O
- Swap these processes to disk to free up more memory
- Blocked state becomes suspend state when swapped to disk
- Two new states
 - Blocked, suspend
 - Ready, suspend

One Suspend State



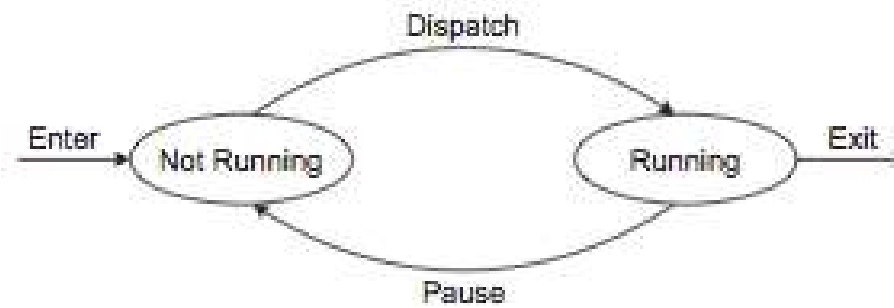
Two Suspend State



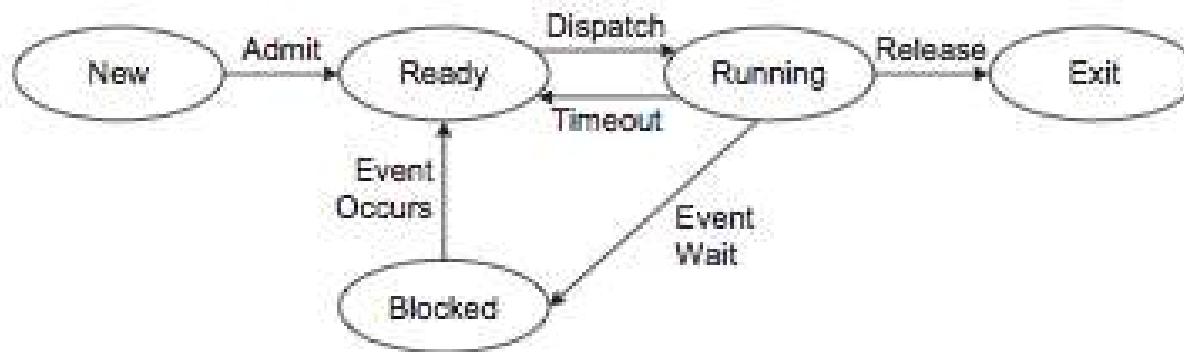
Reasons for Process Suspension

Swapping	The operating system needs to release sufficient main memory to bring in a process that is ready to execute.
Other OS Reason	The operating system may suspend a background or utility process or a process that is suspected of causing a problem.
Interactive User Request	A user may wish to suspend execution of a program for purposes of debugging or in connection with the use of a resource.
Timing	A process may be executed periodically (e.g., an accounting or system monitoring process) and may be suspended while waiting for the next time interval.
Parent Process Request	A parent process may wish to suspend execution of a descendent to examine or modify the suspended process, or to coordinate the activity of various descendent.

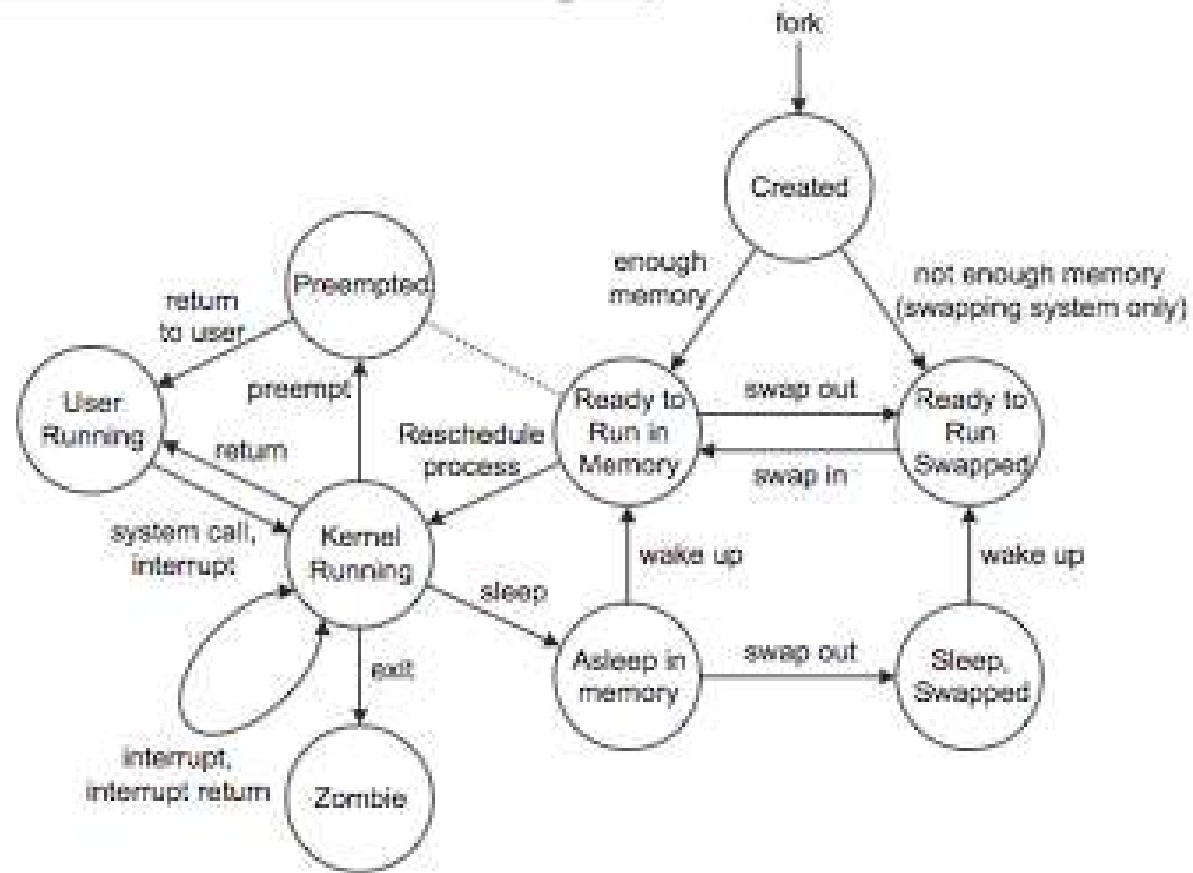
Two State Process Model



Five State Process Model



UNIX Process State Transition Diagram



Review

- Apa hubungannya dan perbedaan antara proses dan program
- Berapa state yang dimiliki oleh sebuah proses?
- Apakah ada antrian proses running? Kenapa atau kenapa tidak ada?