

# PERTEMUAN 4

## LANJUTAN

### Komunikasi Langsung

- Proses harus diberi nama secara jelas:
  - Send (P, message) – kirim pesan ke proses P
  - Receive (Q, message) – terima pesan dari proses Q
- Properti jalur komunikasi
  - Jalur dibangun secara otomatis
  - Setiap jalur memiliki pasangan masing-masing dalam proses komunikasi
  - Jalur komunikasi tersebut biasanya directional

### Komunikasi Tidak Langsung

- Pesan dikirim dan diterima melalui mailboxes (yang ditunjuk sebagai port)
  - Proses
  - Processes can communicate only if they share a mailbox.
- Properti jalur komunikasi
  - Jalur komunikasi hanya dibangun jika proses di-share dalam mailbox
  - Jalur merupakan gabungan beberapa proses
  - Setiap pasangan proses dibagi kedalam beberapa jalur komunikasi.

### Komunikasi Tidak Langsung

- Operasi
  - Membuat mailbox baru
  - Mengirim dan menerima pesan melalui mailbox
  - Menghapus / memusnahkan mailbox
- Primitive didefinisikan: send (A, message) – kirim pesan ke mailbox A receive (A, message) – terima pesan dari mailbox A

### Komunikasi Tidak Langsung

- Mailbox sharing
  - P1, P2, dan P3 berbagi (share) mailbox A.
  - P1, send; P2 and P3 receive.
  - Siapa yang mendapat pesan?
- Solusi
  - Memperbolehkan suatu jalur yang merupakan gabungan lebih dari dua proses

# PERTEMUAN 4

## LANJUTAN

- Hanya memperbolehkan satu proses pada suatu waktu untuk mengeksekusi operasi receive .
- Memperbolehkan sistem untuk memilih receiver. Sender diberitahu siapa yang menjadi receiver.

### Sinkronisasi

- Pesan yang disampaikan dapat diblok atau tidak (non-blocking)
- Blocking dikenal dengan synchronous.
- Non-blocking dikenal dengan asynchronous

### Buffering

- Antrian pesan yang dihubungkan dalam suatu jalur, diimplementasikan dengan tiga jalan:
  1. Zero capacity –tidak ada pesan -Sender harus menunggu receiver (rendezvous).
  2. Bounded capacity –memiliki panjang yang terbatas (finite length) dari pesan. -Sender menunggu pada saat jalur penuh.
  3. Unbounded capacity –memiliki panjang tidak terbatas(infinite length) - Sender tidak pernah menunggu.

### Komunikasi Client-Server

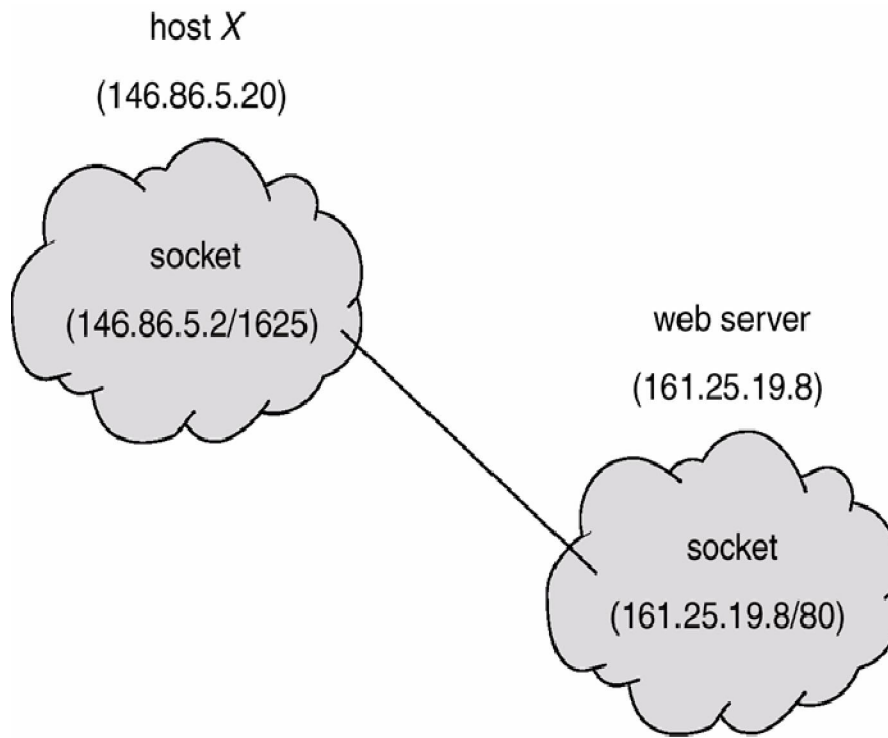
- Sockets
- Remote Procedure Calls (RPC)
- Remote Method Invocation (Java)

### Sockets

- Suatu socket didefinisikan sebagai titik akhir (endpoint) komunikasi
- A socket is defined as an endpoint for communication.
- Gabungan IP address dan port
- Socket 161.25.19.8:1625 mengacu pada port 1625 pada host 161.25.19.8
- Komunikasi berada diantara pasangan socket

### Komunikasi Socket

## PERTEMUAN 4 LANJUTAN



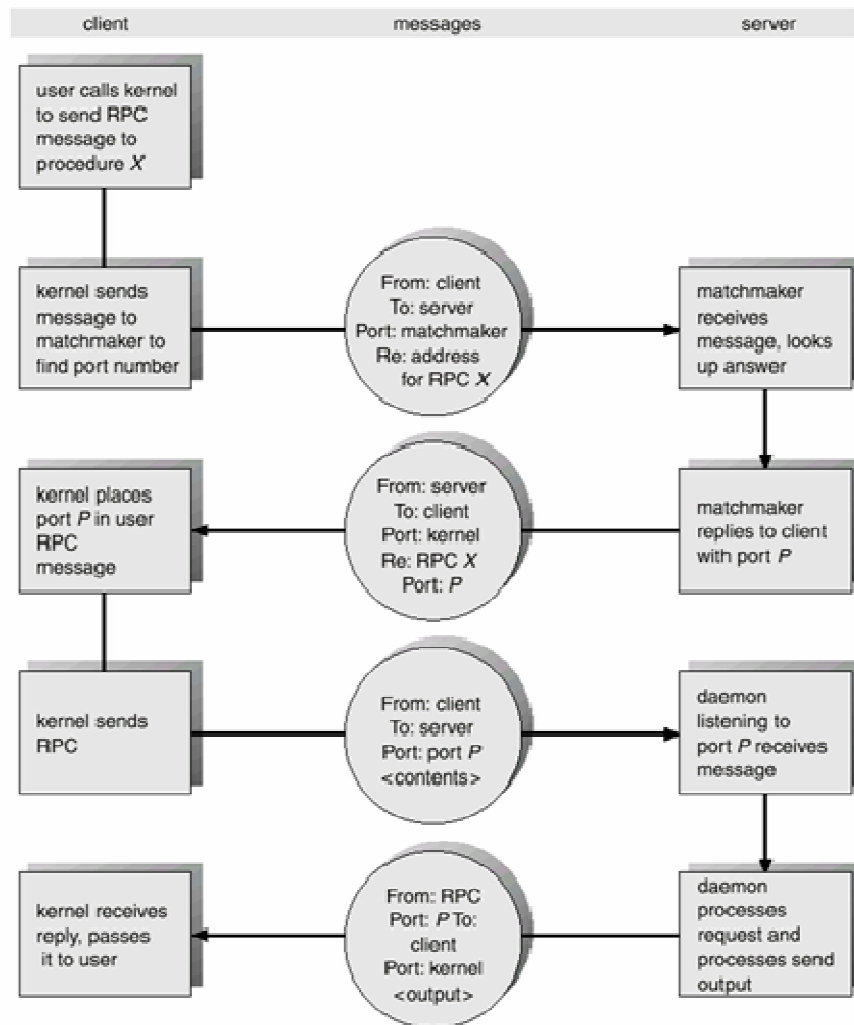
### Remote Procedure Calls (RPC)

- Remote Procedure Call (RPC) adalah abstraksi pemanggilan prosedur diantara proses pada sistem jaringan
- Stubs—proxy sisi client untuk prosedur aktual pada server
- Stub sisi client ditempatkan di server dengan parameter marshalls.
- Stub sisi server menerima pesan, membongkarnya dengan parameter marshall dan menjalankan prosedur pada server.

# PERTEMUAN 4

## LANJUTAN

### Eksekusi RPC



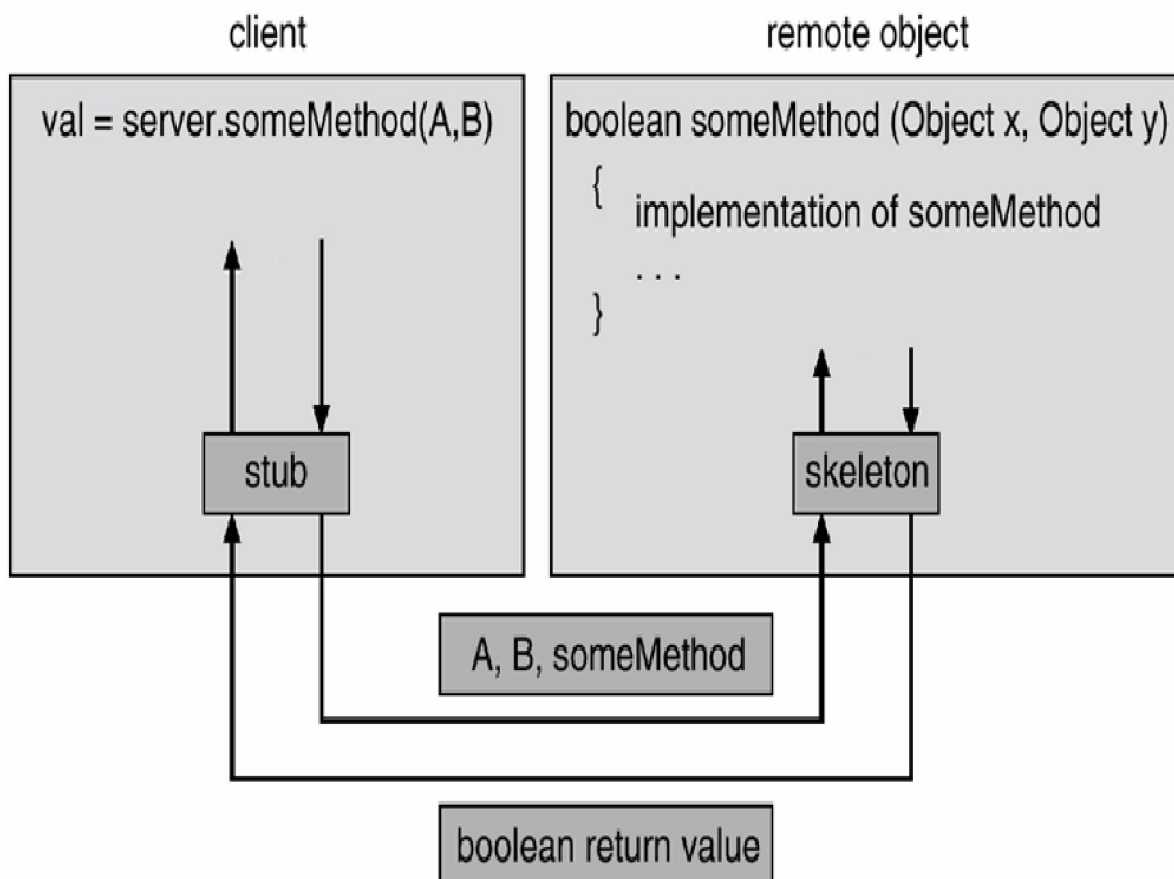
### Remote Method Invocation (RMI)

- Remote Method Invocation (RMI) adalah mekanisme pada JAVA yang hampir sama dengan RPC
- RMI membolehkan program JAVA pada satu mesin untuk menggunakan metode untuk melakukan remote objek.

## PERTEMUAN 4 LANJUTAN



]



### A. SOAL LATIHAN/TUGAS

1. Jelaskan pengertian proses?
2. Jelaskan status proses?
3. Jelaskan penjadwalan proses?

### B. DAFTAR PUSTAKA

Buku

Bambang Hariyanto. 1997. Sistem Operasi, Bandung:Informatika Bandung.

# PERTEMUAN 4

## LANJUTAN

Dali S. Naga. 1992. Teori dan Soal Sistem Operasi Komputer, Jakarta: Gunadarma.

Operating System Concepts (6th or 7th Edition). Silberschatz, Galvin, Gagne, ISBN: 0-471-25060-0. Wiley

Silberschatz Galvin. 1995. 4 Edition Operating System Concepts: Addison Wesley.

Sri Kusumadewi. 2000. Sistem Operasi. Yogyakarta: J&J Learning.

Tanenbaum, A. 1992. Modern Operating Systems. New York: Prentice Hall

Link and Sites:

<http://www.ilmukomputer.com>  
<http://vlsm.bebas.org>  
<http://www.wikipedia.com>