**10. PARALLEL PROCESSING**



**Parallel Processing**

|  |  |  |  |
| --- | --- | --- | --- |
| Penggunakan | lebih dari | satu CPU | untuk |
| menjalankan | sebuah | program | secara |
| simultan. |  |  |  |

Tujuan Utama

# Untuk meningkatkan performa komputasi.



**Komputasi Parallel**

### Salah satu teknik melakukan komputasi secara bersamaan dengan memanfaatkan beberapa komputer secara bersamaan.

Komputasi Paralel membutuhkan

### Algoritma

1. Bahasa pemrograman
2. Compiler



**Pemrograman Parallel**

### Teknik pemrograman komputer yang memungkinkan eksekusi perintah atau operasi secara bersamaan baik dalam komputer dengan satu (prosesor tunggal) ataupun banyak (prosesor ganda dengan mesin paralel) CPU.



**Bahasa Pemrograman yang digunakan pada Pemrograman Parallel**

# 1. MPI (Message Passing Interface)

2. PVM (Parallel Virtual Machine)



**MPI (Message Passing Interface)**

## Sebuah standard pemrograman yang memungkinkan pemrogram untuk membuat sebuah aplikasi yang dapat dijalankan secara paralel.



**MPI (Message Passing Interface)**

Suatu standar protokol yang digunakan untuk pemrograman paralel dan terdistribusi.

Proses pertukaran pesan atau data antar proses adalah dengan mengirimkan pesan melalui media komunikasi.

Model ini juga dapat diimplementasikan pada

bermacam-macam platform, seperti shared-memory.



**Kegunaan MPI**

#### Menyediakan fungsi-fungsi untuk menukar pesan.

1. Menulis kode paralel secara portable.
2. Mendapatkan performa yang tinggi dalam pemrograman paralel.
3. Menghadapi permasalahan yang melibatkan hubungan data irregular atau dinamis yang tidak begitu cocok dengan model data paralel.



**Implementasi MPI**

* + *MPICH (MPI/Chameleon)*

MPICH2 adalah implementasi Message Passing Interface (MPI).

MPI merupakan standar spesifikasi library untuk program message-passing, yang diajukan sebagai standar oleh vendor, implementor, dan user.



**PVM (Private Virtual Machine)**

* + Perangkat lunak yang memungkinkan sekumpulan komputer yang *heterogen* terlihat seperti satu sistem komputer paralel dan dapat digunakan sebagai sebuah sumber daya komputasi yang *koheren*.
  + Istilah *virtual machine* mengacu pada konfigurasi komputer dengan memori tersebar, sedangkan istilah *host* ditujukan untuk komputer yang merupakan anggota dari *virtual machine*. *Host* dapat terdiri dari komputer yang berbeda-beda dan terpisah lokasinya secara fisik.



**Komponen PVM**

* + **PVM *daemon***

Daemon dari PVM, atau disebut PVM daemon (pvmd).

Daemon merupakan semacam program yang berjalan di “belakang” dan biasanya menangani program dari klien. Komponen ini berada pada setiap komputer yang terhubung dalam *virtual machine*.

PVM Berguna untuk membentuk konfigurasi host dalam PVM dan mengkoordinir komunikasi antar *host.*



**Komponen PVM**

* + **PVM libraries**

Komponen ini berisi rutin-rutin antarmuka pemakai PVM untuk melakukan pengiriman pesan, pembuatan proses, koordinasi proses dan modifikasi *virtual machine*.

Pustaka PVM ini memungkinkan adanya interaksi antar task, yang bisa dilakukan dengan menggunakan fungsi *packing* dan *unpacking* pesan.



**Implementasi PVM**

* Secara umum, langkah implementasi komputasi parallel sebagai berikut :

1. Jalankan PVM daemon pada setiap mesin dalam cluster
2. Jalankan program master pada master daemon
3. Master daemon akan menjalankan proses slave.



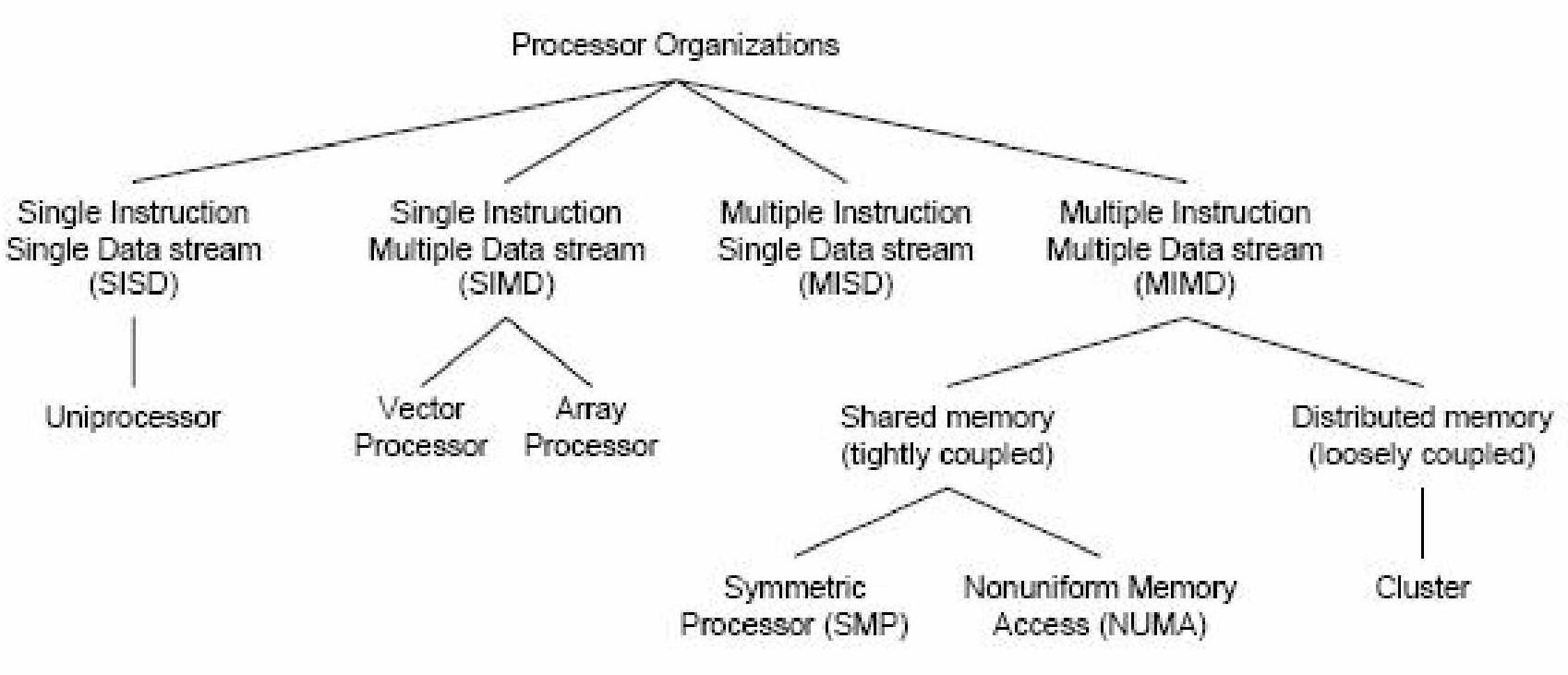
**Implementasi PVM**

* Untuk mengimplementasikannya, dapat memakai tools :

- PVM versi 3.4.5, virtual machine dan routine untuk

komputasi parallel

* + rsh (*remote shell*), aplikasi untuk authentikasi dan komunikasi proses antar komputer.
  + xpvm versi 1.2, interface grafis untuk PVM dengan animasi eksekusi komputasi parallel yang dapat dilihat dilayar



**Arsitektur Komputer Parallel**



Komputer SISD

## Semua instruksi dikerjakan terurut satu demi satu. Pada umumnya komputer terdiri atas satu buah pemroses (*single processor*).

Contoh

### UNIVAC1 4. IBM 360

1. CDC 7600 5. Cray 1
2. PDP 1



Komputer SIMD

## terdapat lebih dari satu elemen pemrosesan yang dikendalikan oleh sebuah unit pengendali yang sama

Contoh

1. ILLIAC IV 4. Cray Y-MP
2. MasPar 5. Thingking Machine CM-2
3. Cray X-MP, 6. Cell Processor (GPU)



Komputer MISD

Memiliki *n* unit pemroses yang masing-masing menerima dan mengoperasikan instruksi yang berbeda terhadap aliran data yang sama

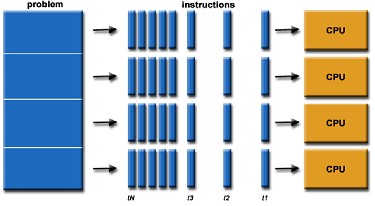
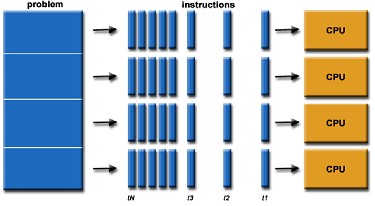
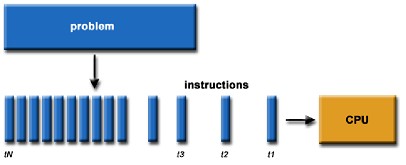
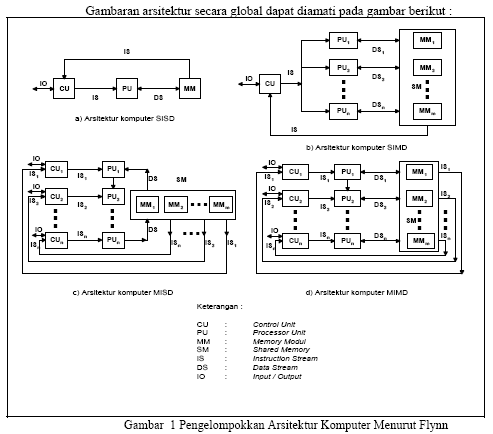


Komputer MIMD

Murni terdapat interaksi di antara *n* pemroses, bersifat tightly coupled jika tingkat interaksi antara pemroses tinggi dan loosely coupled jika tingkat interaksi antara pemroses rendah

Contoh

1. IBM POWER5 4. HP/Compaq AlphaServer
2. Intel IA32 5. AMD Opteron
3. Cray XT3 6. IBM BG/L



Perbedaan antara komputasi tunggal dengan

komputasi paralel

Komputasi Tunggal Komputasi Parallel