

8. CLUSTERING (PC-CLUSTERING)

8.1 Konsep Cluster

8.1.1 Motivasi

8.1.2 Review SMP

8.1.3 Taxonomy Tron

8.1.4 Benefits of Cluster

8.1.5 Model & Konfigurasi Cluster

8.1.6 Issue: Desain Sistem Operasi

8.1.7 Arsitektur Cluster

8.1.8 Perbandingan dengan SMP

8.1.9 Contoh Cluster: WINNT, SUN, Linux-Beowulf

8.2 Membangun PC-Clustering

8.2.1 Resep untuk sebuah cluster kecil

8.2.2 Comodity Computers

8.2.3 Interkoneksi

8.3.3 Sistem operasi

8.3.4 Library untuk aplikasi paralel

8.3.5 Compiler dan tools yang terkait

8.3.6 Merakit Cluster

8.3.7 Aplikasi



8.3 Cluster WINNT (Wolfpack)

8.3.1 Motivasi

8.3.2 Aplikasi Target

8.3.3 Model Cluster

8.3.4 Desain dan komponen Wolfpack

8.3.5 Resources dan contoh

8.3.6 Interkoneksi

8.3.7 Tools manajemen cluster

8.3.8 Manajemen aplikasi

8.4 Cluster Linux (Beowulf)

8.4.1 Motivasi

8.4.2 Aplikasi Target

8.4.3 Model dan Arsitektur Cluster

Referensi :



8.1 Konsep Clusters



8.1.1 Motivasi

- Alternatif dari Symmetric Multi-Processor
- Kinerja tinggi (high performance)
- Ketersediaan tinggi (high availability)
- Aplikasi server

Definisi

- Sekumpulan komputer yang terkoneksi seluruhnya bekerja bersama sebagai satu kesatuan sumber daya atau sistem
 - Illusinya menjadi satu mesin
 - Setiap komputer disebut ach "node"



8.1.2 Review SMP

- **Komputer “stand alone” dengan karakteristik sbb:**
 - Dua atau lebih prosesor yang mirip dan kemampuan sama/hampir sama
 - Processors share memori dan I/O yang sama
 - Processors terkoneksi melalui sebuah bus atau koneksi internal lainnya
 - Waktu akses memory secara aproksimasi sama u/ setiap processor
 - Seluruh processors share access ke I/O
 - Baik melalui channels yang sama atau berbeda menghasilkan “paths” ke devais yang sama
 - Seluruj processors dapat melakukan fungsi-2 yang sama (sehingga disebut ***symmetric***)
 - Sistem dikontrol oleh OS yang terintegrasi
 - Melayani/membantu interaksi antara processors
 - Interaksi pada level job, task, file dan data element

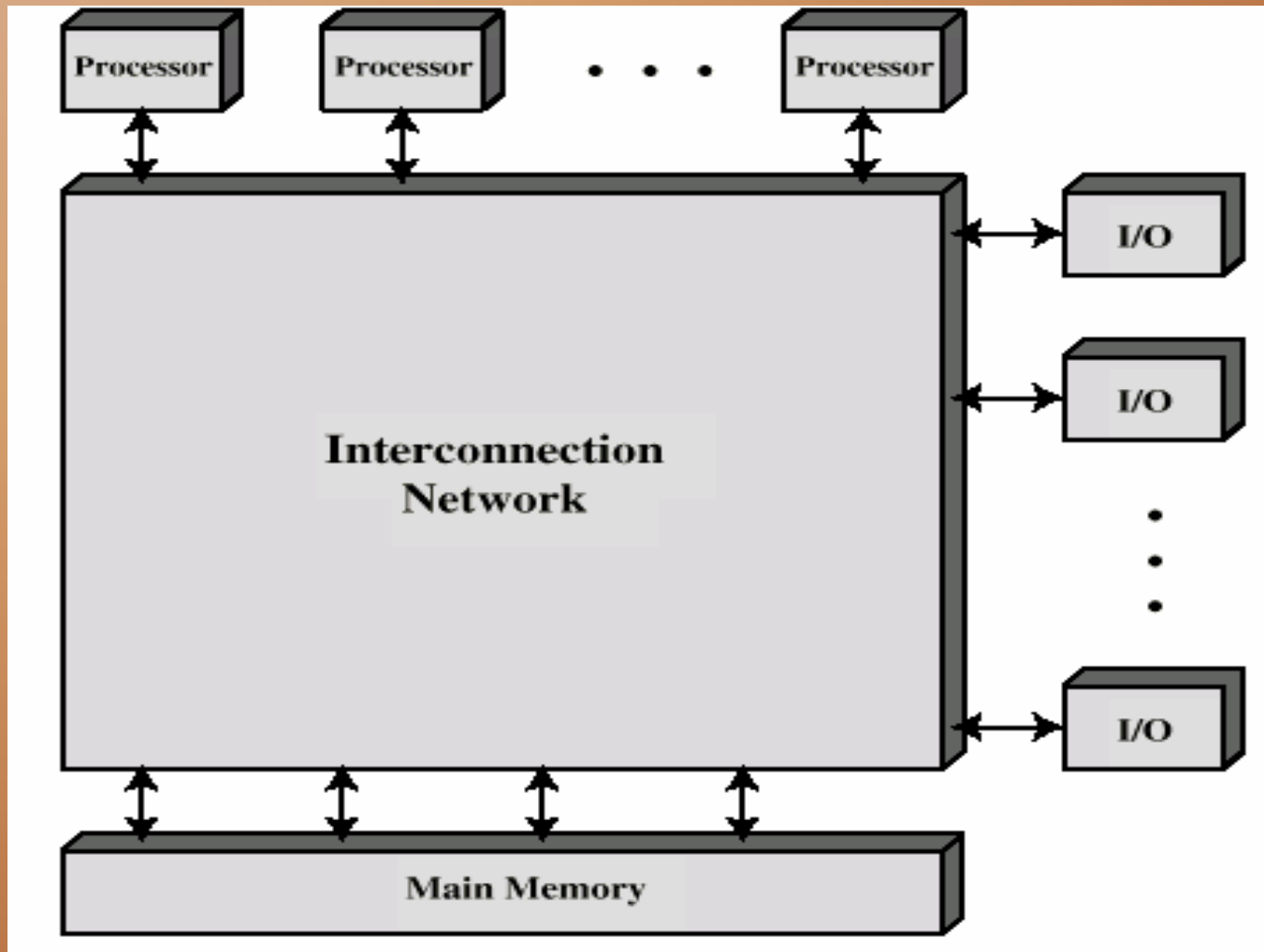


Keuntungan SMP (review cont'd)

- Kinerja
 - Beberapa job/work dapat dilakukan secara paralel
- Availabilitas
 - karena seluruh processors dapat memalukan fungsi-2 yang sama, kegagalan dari sebuah processor tidak menghambat/menyetop sistem
- Peningkatan inkremental (Incremental growth)
 - User dapat meningkatkan kinerja dengan menambahkan additional processors
- Penskalaan (Scaling)
 - Vendors dapat meingkat range produk berdasarkan pada jumlah processors



Diagram Block Diagram MP terkopel secara kaku dan terikat (Tightly Coupled Multiprocessor) (review cont'd)

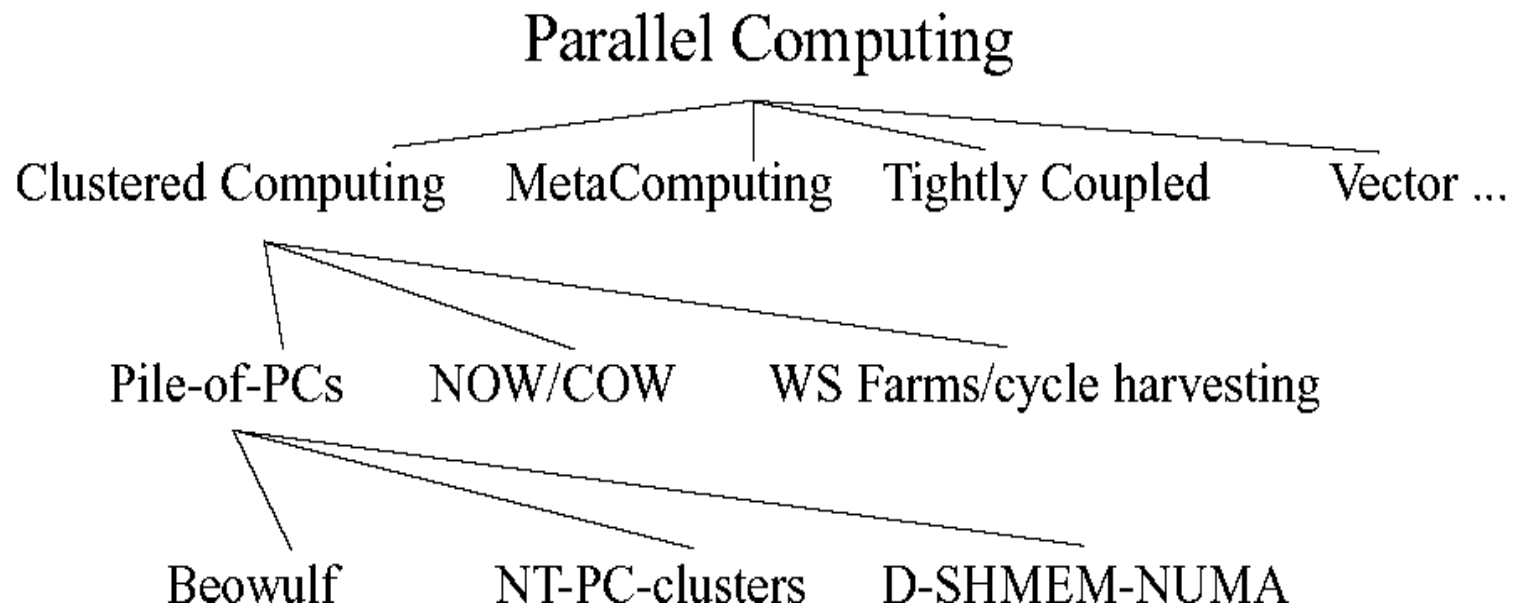


8.1.3 Taxonomy Tron

TAXONOMY

According to Tron

(incomplete, inconsistent, and entirely biased)



8.1.4 Benefits of Clusters

- **Skalibilitas absolut (absolute scalability)**
 - Dapat mempunyai lusian mesin-mesin yang masing-masing multiprocessor
- **Incremental scalability**
 - Penambahan sistem-2 baru dengan inkremen yang kecil (small increments)
- **High availability**
 - Kegagalan satu node tidak berarti “loss of service”
- **Superior price/performance**
 - Dengan biaya yang jauh lebih murah, Cluster dapat menghasilkan power komputasi yang sama atay lebih tinggi dari riil komputer paralel

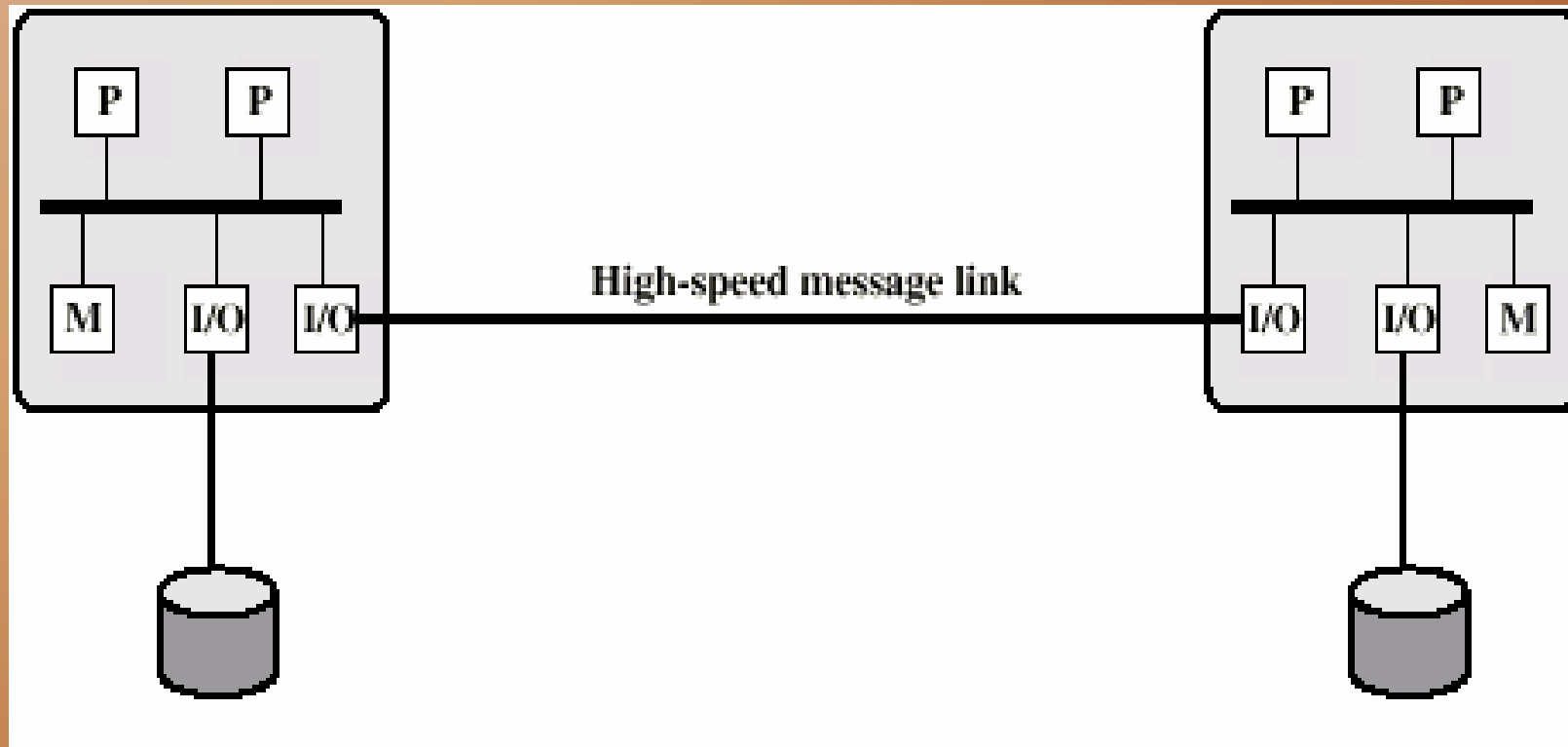


8.1.5. Model dan Konfigurasi Clusters

- **Server Terpisah**
 - Setiap komputer merupakan sebuah server terpisah
 - Tidak men- shared disks
 - Perlu software manajemen atau scheduling
 - Data harus secara konstan dikopi diantara sistem



Konfigurasi Cluster – Standby Server , No-Shared Disk



8.1.5. Model dan Konfigurasi Clusters (cont'd)

- **Tidak men-shared apapun (non-shared)**
 - Mereduksi overhead komunikasi
 - Beberapa servers terkoneksi pada disk-disk bersama (common disks)
 - Disks terpartisi menjadi volume-2
 - Setiap volume dimiliki oleh sebuah komputer
 - Jika sebuah komputer gagal, komputer yang lainnya memperoleh kepemilikan (ownership) volume

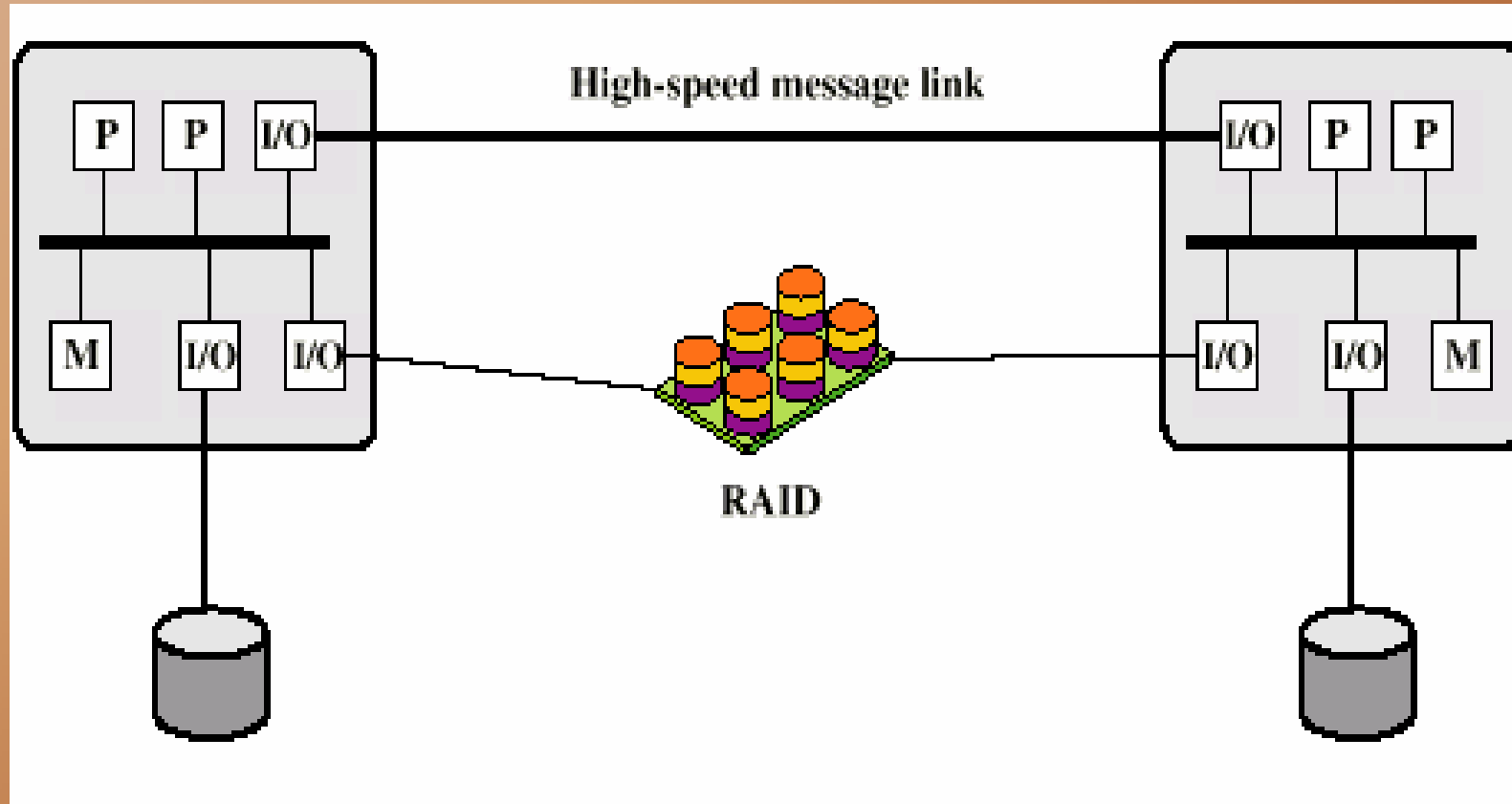


8.1.5. Model dan Konfigurasi Clusters (cont'd)

- **Shared disk**
 - Multiple computers share disks yang sama pada waktu yang sama
 - Setiap komputer memiliki akses ke seluruh volume pada seluruh disks



Konfigurasi Cluster - Shared Disk



8.1.6 Issue: Desain Sistem Operasi

- **Manajemen kegagalan (Failure management)**
 - “Highly available” cluster menimbulkan suatu probabilitas yang tinggi bhw seluruh resources akan dilayani
 - Tidak ada jaminan mengenai keadaan dari transaksi yang dieksekusi secara parsial, jika kegagalan terjadi
 - “Fault-tolerant” cluster menjamin dan memastikan bahwa seluruh resources akan selalu tersedia



8.1.6 Issue: Desain Sistem Operasi (cont'd)

- **Penyeimbangan beban (Load balancing)**
 - Jika komputer baru ditambahkan ke cluster, fasilitas load-balancing secara otomatis akan memasukkan komputer tsb dalam “scheduling” aplikasi-aplikasi
- **Parallelisasi Komputasi**
 - Parallelisasi kompiler
 - Aplikasi ter-parallel
 - Perhitungan (computing) parametrik



8.1.7 Arsitektur Komputer Cluster

- **Fungsi-2 dan layanan-2 middleware cluster**
 - Single entry point
 - Single file hierarchy
 - Single control point
 - Single virtual networking
 - Single memory space
 - Single job-management system



8.1.7 Arsitektur Komputer Cluster (Cont'd)

- **Fungsi-2 dan layanan-2 middleware cluster**
 - Single user interface
 - Single I/O space
 - Single process space
 - Checkpointing
 - Process migration



8.1.8 Perbandingan dengan SMP

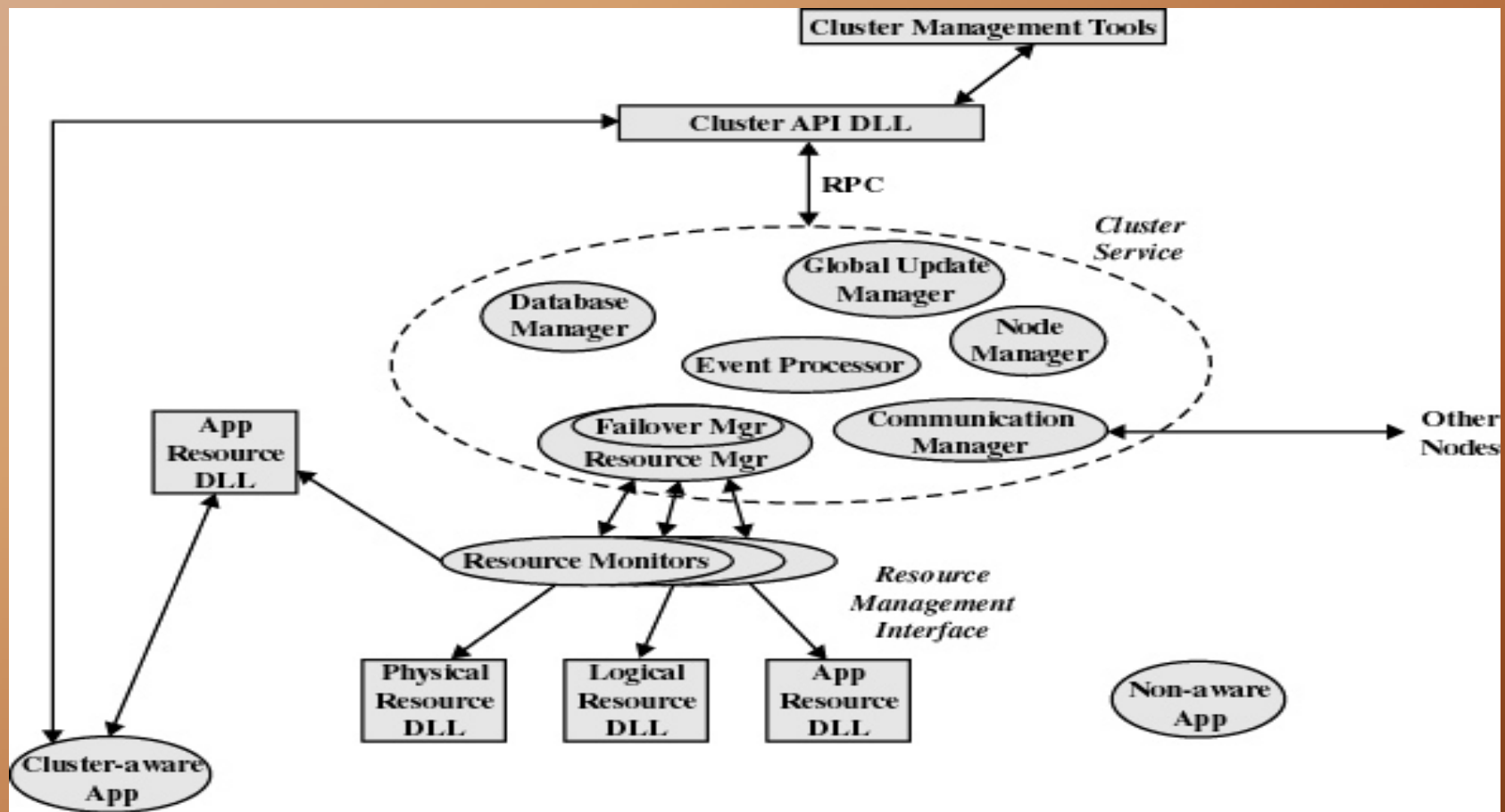
- SMP mudah di manage dan dikonfigurasi
- SMP menempati ruang yang lebih kecil dan power supply yang lebih kecil
- Clusters lebih baik dalam “incremental and absolute scalability”
- Clusters availibilitasnya superior



8.1.9 Contoh Cluster: Layanan Cluster WINNT

- Layanan cluster (cluster Service)
 - Koleksi software pada setiap node yang mengatur (manages) seluruh aktivitas spesifik-cluster
- Resource
 - Item dimanage oleh layanan cluster
- Online
 - Online pada node jika node menyediakan layanan pada node spesifik
- Group
 - Koleksi resources diatur sebagai sebuah unit tunggal (single unit)



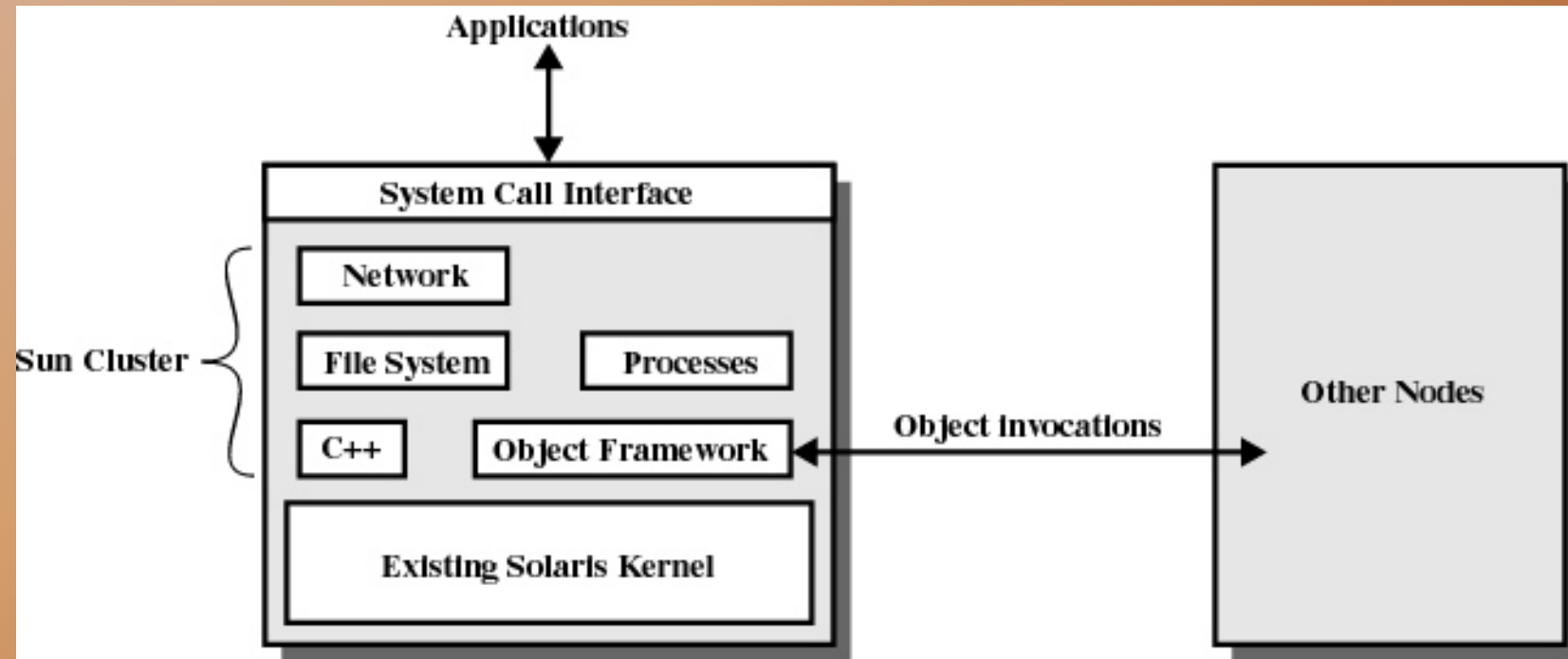


Block diagram server cluster WINNT (SHOR97)

8.1.9 Contoh Cluster: Cluster SUN

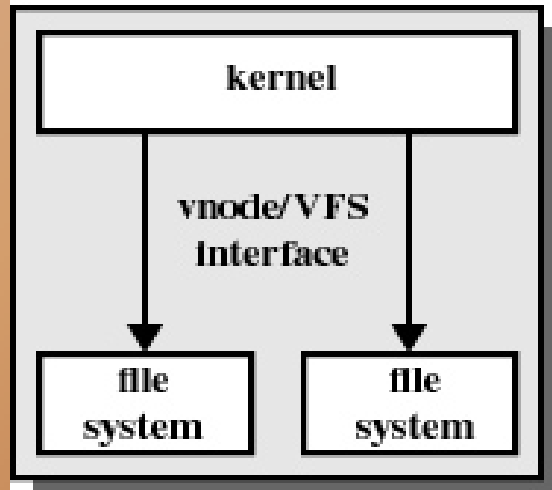
- Komponen utama
 - Dukungan komunikasi dan obyek (object and communication support)
 - Manajemen proses (process management)
 - Jaringan (networking)
 - Sistem file terdistribusi global



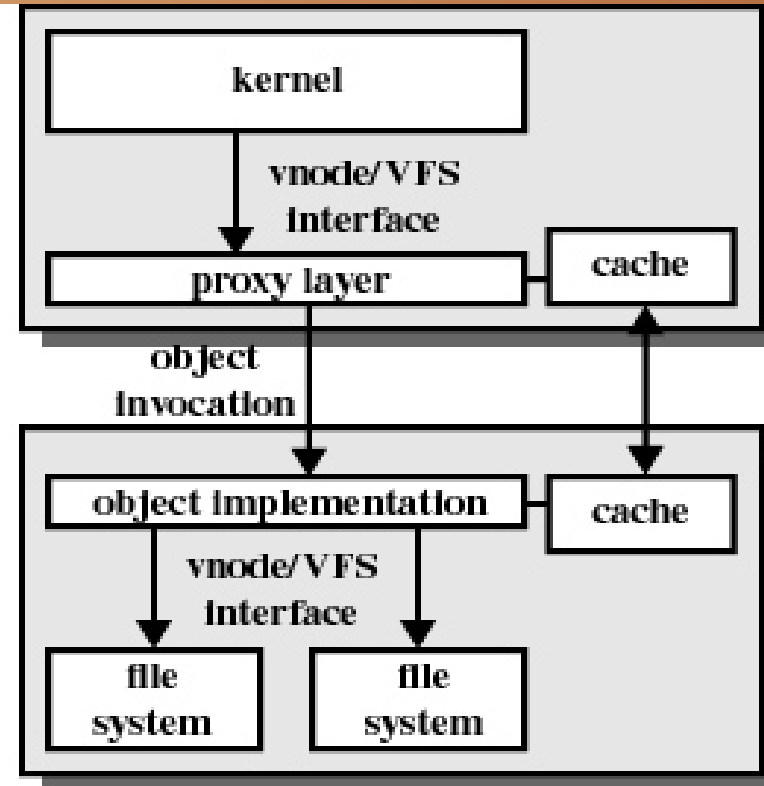


Struktur cluster SUN





(a) Standard Solaris



(b) Sun Cluster

Ekstensi sistem file cluster SUN

8.1.9 Contoh Cluster: Cluster Linux (Beowulf)

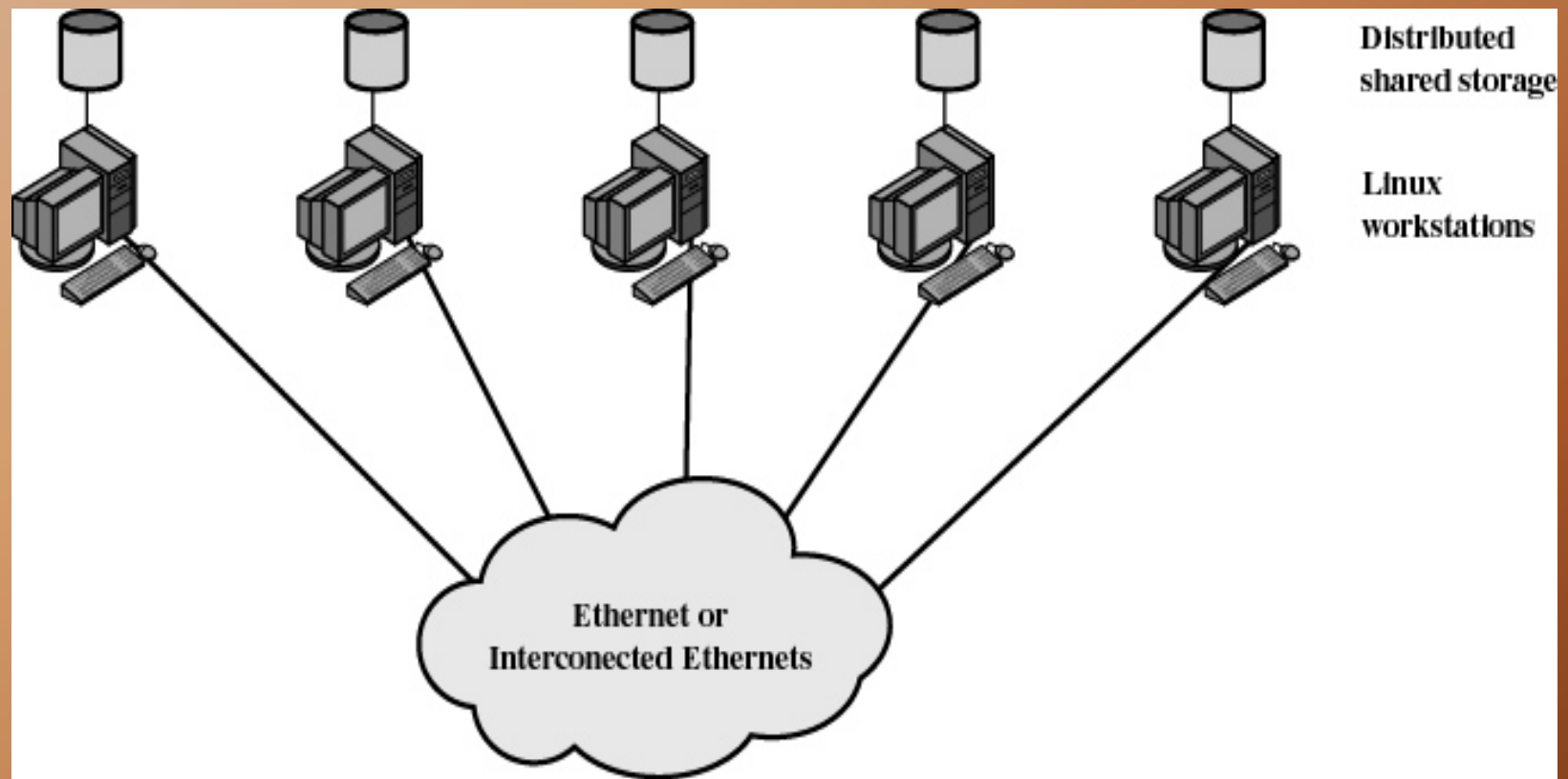
- **Fitur-fitur utama**
 - Mass market commodity components
 - Dedicated processors (rather than scavenging cycles from idle workstations)
 - A dedicated, private network (LAN or WAN or interneted combination)
 - No custom components
 - Easy replication from multiple vendors



8.1.9 Contoh Cluster: Cluster Linux (Beowulf) (cont'd)

- **Fitur-fitur utama**
 - Scalable I/O
 - A freely available software base
 - Using freely available distribution computing tools with minimal changes
 - Returning the design and improvements to the community





Konfigurasi Beowulf generik

8.2 Membangun Cluster untuk Komputasi Paralel



Sub-topik

- **Resep untuk Cluster kecil**
- **Commodity Computers** →
 - Pemilihan komponen: Motherboards, Memory and Cache, and Assembly Complete Systems
 - HINT Benchmarks
- **Interkoneksi** →
 - Myrinet, Gigabit Ethernet, ATM, Fast Ethernet, FDDI, HIPPI
 - Performance of Commodity Interconnects
- **Sistem Operasi** → Linux, FreeBSD, NetBSD, Windows NT, Windows95
- **Libraries u/ aplikasi parallel** → MPICH, PVM, DIPC, ScaLAPACK
- **Compilers dan Tools terkait** → Absoft, GNU, Paralogic, The Portland Group
- **Merakit Cluster**
- **Contoh aplikasi**



8.2.1 Resep Clusters sederhana: Empat-node

Komponen penting:

- 4 Pentium II PCs dengan RAM 64MB atau lebih RAM, disk 2GB atau lebih besar, dan drive EIDE CD-ROM
- 5 Fast Ethernet Network Interface Cards, misal SMC 9332 EtherPower 10/100
(Empat cards u/ menghubungkan node-2 cluster, dan satu card u/ menghubungkan satu dari node cluster ke sisa lain dari jaringan kita)
- 5 “kategory 5” kabel UTP dgn RJ45
- 1 Fast Ethernet (100BASE-Tx) Hub atau Fast Ethernet Switch, misalkan Bay Networks 350T
- Linux CD (misal Extreme Linux);
- MPICH



8.2.1 Resep Clusters sederhana (cont'd)

- Langkah-langkah:
 1. Masukkan dua Fast Ethernet NIC ke PC yang akan berfungsi sebagai cluster server dan penghubung ke sisa lain jaringan kita. Masukkan satu Fast Ethernet card ke masing-masing tiga PC lainnya
 2. Sambungkan empat Category 5 kabel ke Fast Ethernet hub atau switch ke masing-masing PC
 3. Sisa kabe Category 5 cable untuk koneksi ke sisa lainnya dalam jaringan
 4. Instal Linux pada setiap PC. Juga install GNU C compiler and C libraries pada server
 - Ketika mengkonfigurasi TCP/IP, disarankan digunakan IP addresses 192.168.1.1, 192.168.1.2, 192.168.1.3, dan 192.168.1.4 ke PC pertama, kedua, ketiga dan keempat
 - PC pertama akan menjadi node server
 - Extra network card pada node server node ditetapkan sebuah IP address oleh institution's network manager.



8.2.1 Resep Clusters sederhana (cont'd)

5. Dengan Linux pada setiap PC , edit file /etc/hosts pada keempat PC sehingga mengandung baris-baris berikut:
 - **192.168.1.1 node1 server**
 - **192.168.1.2 node2**
 - **192.168.1.3 node3**
 - **192.168.1.4 node4**
6. Edit file /etc/hosts.equiv pada keempat PC sehingga mengandung baris-baris berikut
 - **Node1**
 - **Node2**
 - **Node3**
 - **Node4**
7. Konfigurasi ini perlu agar devais MPICH's p4 dapat digunakan untuk mengeksekusi suatu aplikasi paralel terdistribusi.



8.2.1 Resep Clusters sederhana (cont'd)

8. Pada node server node, buatlah sebuah directory `/mirror`. Konfigurasi server menjadi sebuah NFS server, dan tambahkan pada `/etc/exports` baris berikut:
`/mirror node1(rw) node2(rw) node3(rw) node4(rw)`
9. Pada node lainnya, buat directory `/mirror`. Tambahkan baris berikut pada `/etc/fstab`:
`server:/mirror /mirror nfs rw,bg,soft 0 0`
Perintah ini akan mengeksport directory `/mirror` dari server dan mounts directory tersebut pada masing-masing 2 clients agar distribusi software antara node lebih mudah.
10. Pada node server, install MPICH. Dokumentasi MPICH dapat diperoleh pada pages berikut
<http://www.mcs.anl.gov/mpi/mpich/docs.html>.

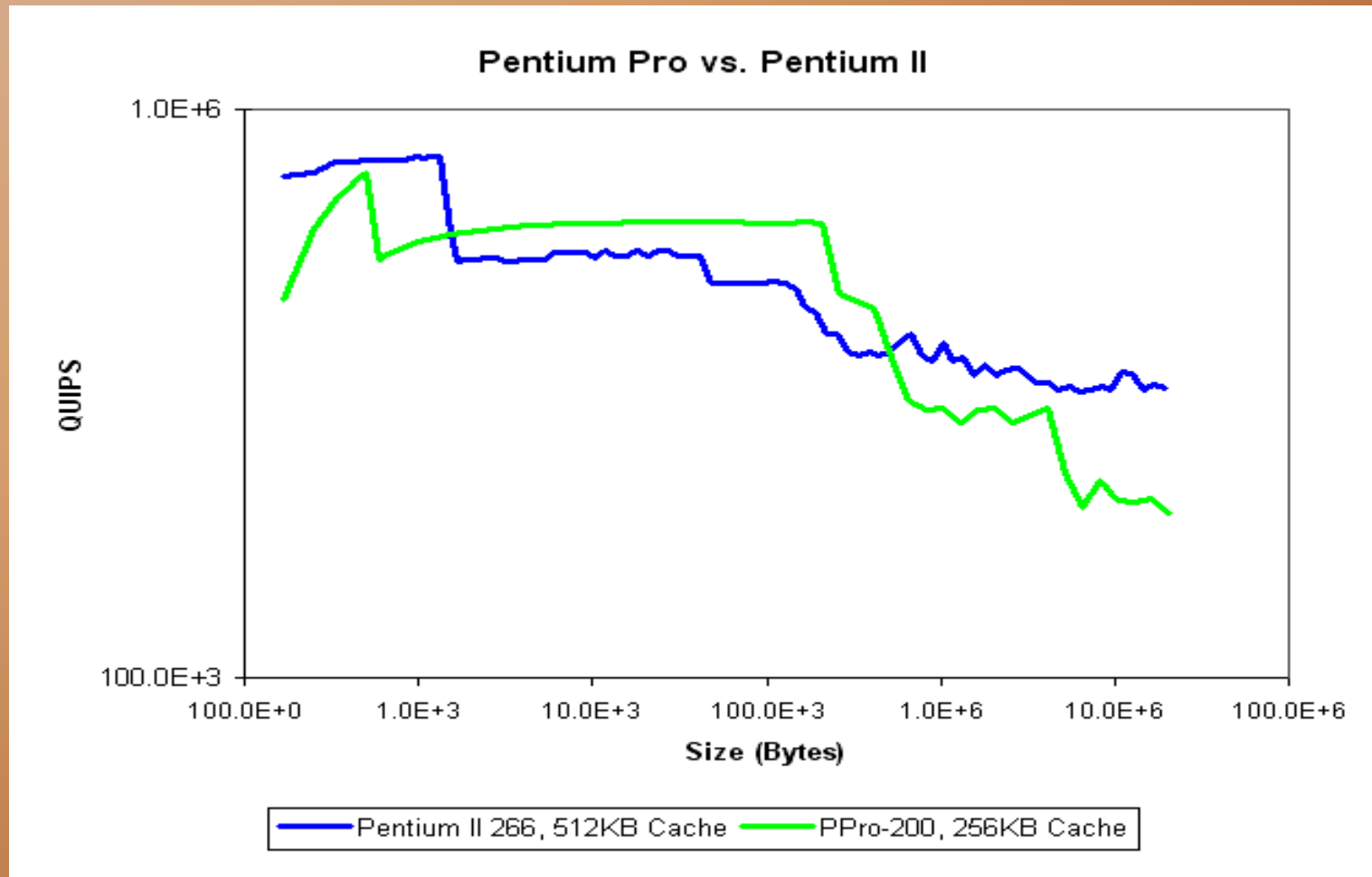


8.2.2 Commodity Computers

- Pemilihan komponen-komponen:
 - Motherboards, Memory and Cache, and Assembly Complete Systems
- HINT Benchmarks dari Commodity Systems



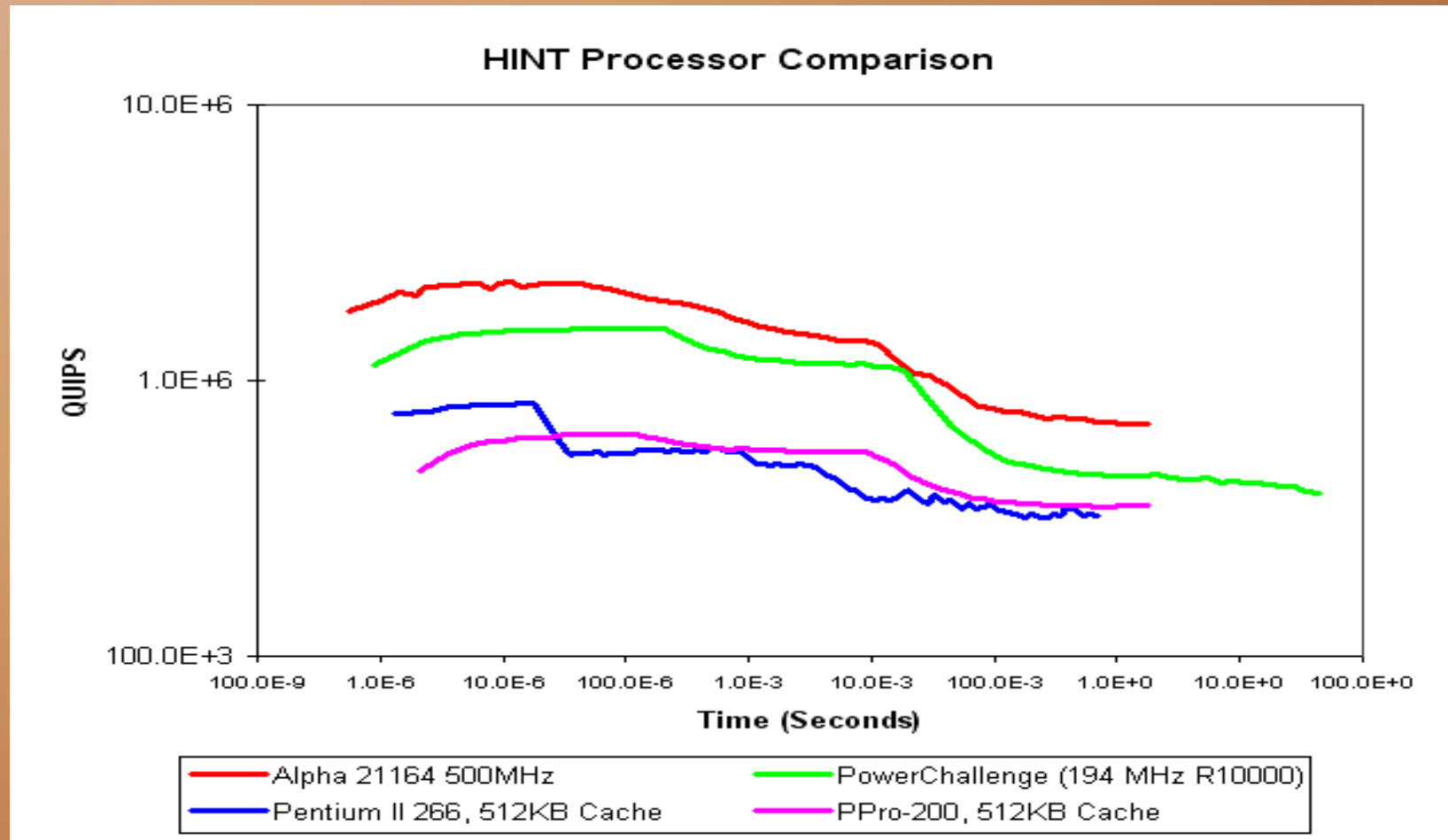
HINT Benchmarking



HINT Comparison dari Pentium Pro and Pentium II



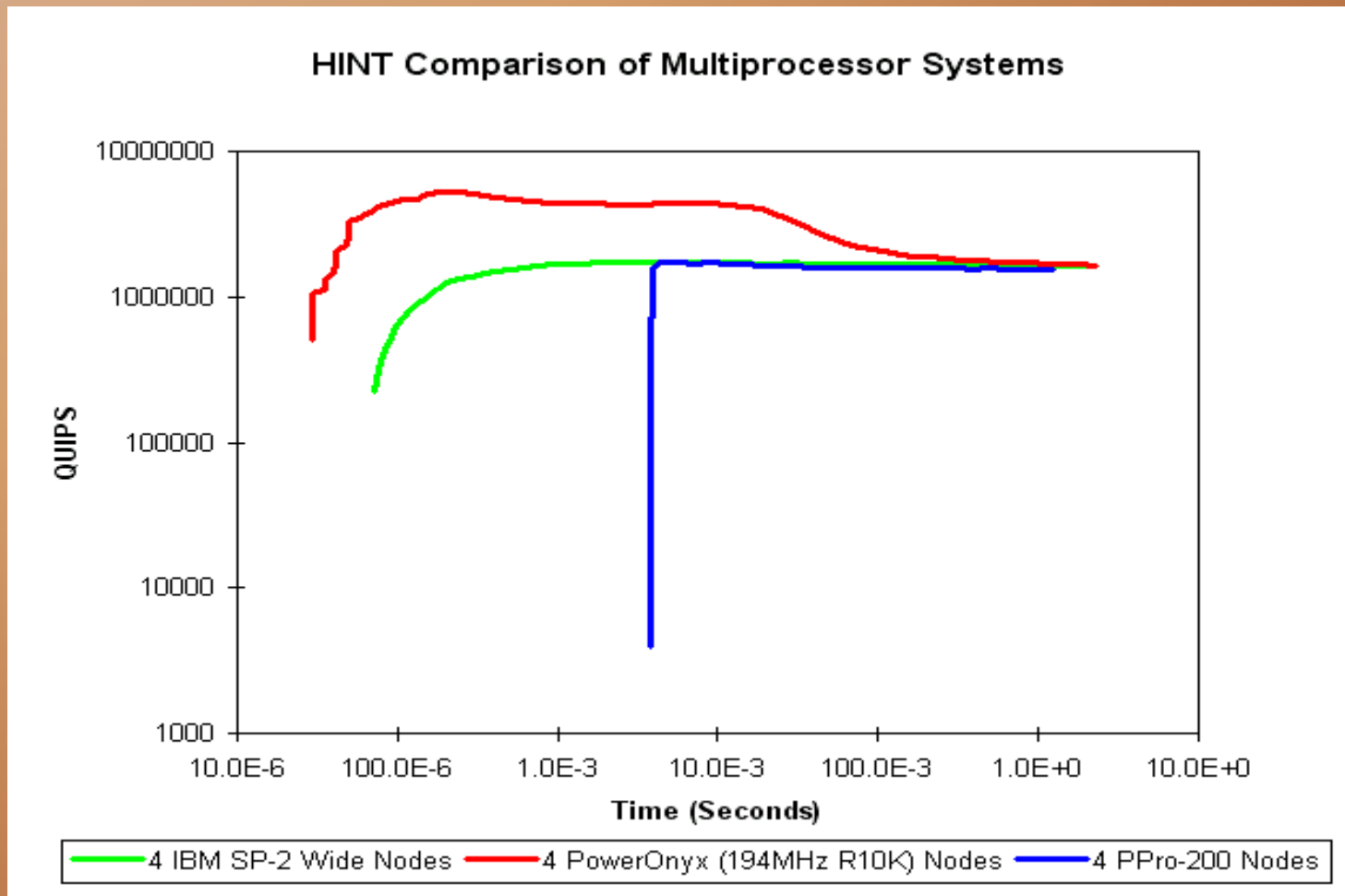
HINT Benchmarking



Perbandingan: Pentium Pro dan Pentium II thp Competitive CPUs



HINT Benchmarking



Perbandingan antara sistem Multiple-processor



8.2.3 Interkoneksi

- **ATM**

- a switched virtual-circuit technology.
- runs at 155Mbps.
- memerlukan ATM network interface cards (NICs)
- fiber optic atau Category 5 unshielded twisted pair (UTP) copper cables (depending on the selected equipment), dan paling sedikit satu ATM switch u/ interkoneksi.
- Cukup mahal dan cenderung latency-nya tinggi jika dibandingkan dengan dgn LAN

- **FDDI**

- a token ring fiber optic network → runs at 100Mbps.
- latency lebih tinggi ketimbang Fast Ethernet atau Gigabit Ethernet karena meskipun beban rendah sebuah station harus selalu menunggu “token” sebelum dapat mentransmisi data-nya.
- Spt. ATM, cukup mahal dan biasanya tidak digunakan untuk clusters.



8.2.3 Interkoneksi (cont'd)

- **Ethernet**

- a time-tested network technology that has become very cheap but at 10Mbps does not offer much throughput.
- The most popular implementation of Ethernet is 10BASE-T
- a very cheap cluster, but would be a serious bottleneck for a cluster of modern PCs.

- **Fast Ethernet**

- an upgrade of Ethernet that provides 100Mbps transmission speeds.
- The most popular implementation is 100BASE-TX
- Fast Ethernet network interface cards have become a true commodity item, with prices for high-quality boards
- A Fast Ethernet repeater that is used for one of our clusters



8.2.3 Interkoneksi (cont'd)

- **Gigabit Ethernet**
 - an up-and-coming technology that has recently been standardized.
 - Gigabit Ethernet layers the Ethernet media access control protocol (CSMA/CD) over the established ANSI-standard Fibre Channel physical technology with a minor adjustment to increase real data throughput to 1Gbps.
 - Gigabit Ethernet menghubungkan NICs dengan sebuah Gigabit Ethernet repeater atau switch dgn multimode fiber optic cable.
 - Packet Engines → disebut "full-duplex repeater" yang, spt. full-duplex switch, menghindari collisions tetapi, spt. repeater hub, "floods" packets ke seluruh ports.
 - Gigabit Ethernet masih mahal (continue to fall)



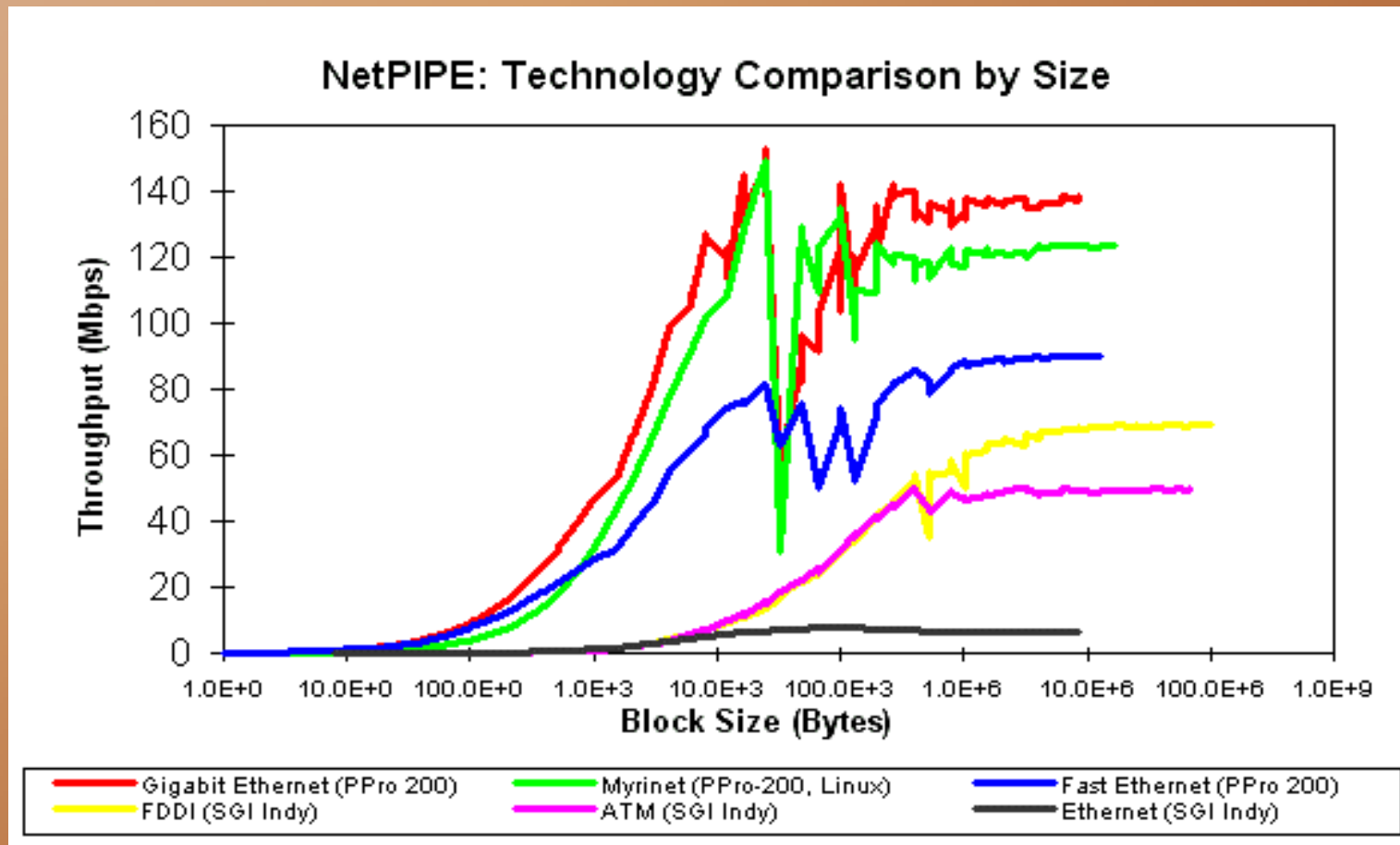
8.2.3 Interkoneksi (cont'd)

- **Myrinet**

- Penghubung high performance yang biasa digunakan pada kebanyakan cluster yang lebih mahal.
- 1.2Gbps technology yang menggunakan short-distance (10 feet) copper cables u/ menghubungkan NICs ke switches.
- Menyediakan low-level messaging melalui protokol-2 yang meningkat latency melalui TCP dengan mereduksi overhead.
- Saat ini, Myrinet dan Gigabit Ethernet memerlukan baik PC NICs dan interconnects (hubs/switches) dari a single vendor.



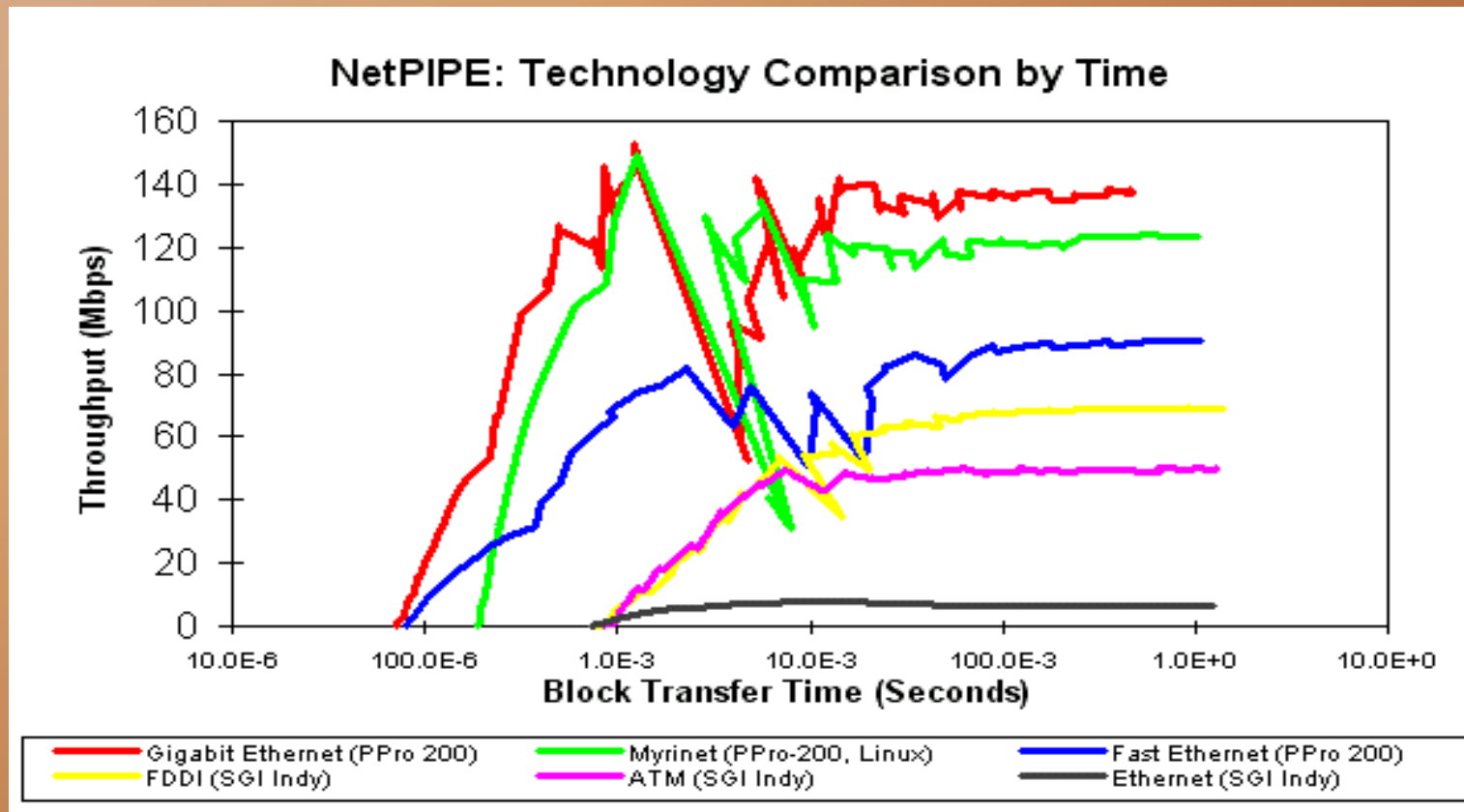
8.2.3 Interkoneksi (cont'd)



Throughput: Ethernet, FDDI, ATM, Fast Ethernet, Gigabit Ethernet, dan Myrinet



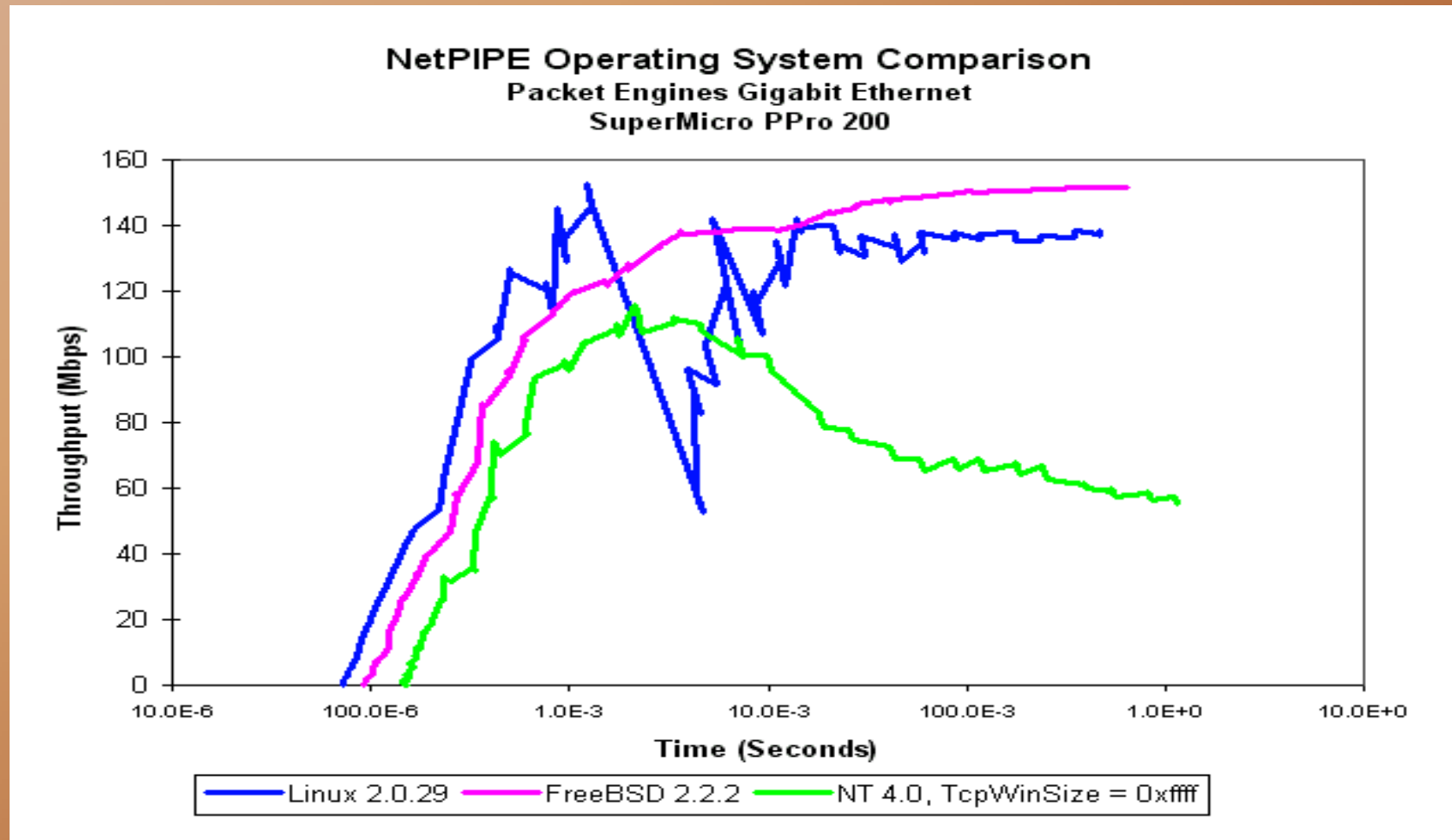
8.2.3 Interkoneksi (cont'd)



Latency of Ethernet, FDDI, ATM, Fast Ethernet, Gigabit Ethernet, and Myrinet



8.2.4 Sistem Operasi



Performance of Gigabit Ethernet on Linux, FreeBSD, and NT



8.2.5 Libraries u/ komputasi Paralel

- **Message Libraries**

- **MPI**

- portable standard u/ message-passing libraries
 - Didesain u/ menyediakan suatu spesifikasi library bersama u/ massively parallel processor vendors.
 - Aplikasi Parallel yang dikembangkan dgn MPI dapat di running tanpa recoding pada sistem-sistem mulai clusters komputer murah sampai ke superkomputer
 - Paling sedikit dua implementasi MPI tersedia bebas u/ penggunaan pada cluster: **MPICH** from Argonne National Labs, and **LAM** from the University of Notre Dame.

- **PVM**

- library lain yang mendukung program parallel pada clusters dan ditargetkan khususnya pada kumpulan heterogen komputer UNIX/Linux.



8.2.5 Libraries u/ komputasi Paralel (cont'd)

- **Higher-Level Libraries**

- ScaLAPACK

- sebuah library routine aljabar linier paralel yang beroperasi pada cluster yang menggunakan PVM atau MPI.
 - Memerlukan instalasi routine aljabar linier **LAPACK** dan library **BLACS** u/ komunikasi di dalam program aljabar linier.



8.2.6 Compiler dan Tools yang terkait

- **Compiler**
 - **GNU compilers**
 - **The Portland Group** → compilers u/ dialek Fortran termasuk FORTRAN77, High Performance Fortran, dan Fortran90. The Portland Group menyediakan compilers baik u/ PCs running Linux maupun supercomputers spt. Cray-T3E.
 - **Absoft Corporation** → Fortran compilers u/ Linux, Windows NT, dan Power Macintosh systems.
- **Tools terkait**
 - Various automated parallelization misal BERT77 → versi lite digunakan u/ cluster linux



8.2.7 Merakit Cluster

- **Server vs Regular Nodes**
 - Karena interkoneksi yang digunakan tampaknya tidak langsung terhubung langsung ke jaringan luar yang lebih besar, maka satu dari node tampaknya memerlukan network card tambahan u/ menghubungkan cluster jika jaringan luar yang lebih besar
- **Hardware**
 - Add Interconnect Hardware and Test
 - Place PCs on Shelving
 - Connect Cables
 - Test the Shelf of PCs
- **Software**
 - Install and Configure the Operating System
 - Install Software
 - Authorize Users



8.2.8 Aplikasi

- Bidang Computational Science → Grand Challenge Problem, mis.
 - Modeling dan simulasi Iklim/Cuaca
 - Molecular Dynamics Simulation in material physics and chemistry
 - Monte Carlo Simulation in Nuclear Physics
- Bidang Elektronika, mis.
 - Desain VLSI
- Bidang Penerbangan
 - Flight Simulator
- Dll.



8.3 Cluster WIN-NT (Wolfpack)



8.3.1 Motivasi

- Memperluas Windows NT dgn memasukkan konsep dan fitur cluster
- Mudah digunakan, produk market massak
- Availabilitas
- Manageabilitas
- Skalabilitas
- Mendukung → file/print, web Server termasuk aplikasi pendukungnya sbg fitur dasar
- API u/ produk-2 “cluster aware”



8.3.2 Aplikasi Target

- Database servers
- E-mail, Groupware, and Productivity Applications
- Transaction Processing Servers
- Enterprise Applications (i.e.; SAP, CA/ Uni-center)
- Internet Web Servers
- File and Print Servers

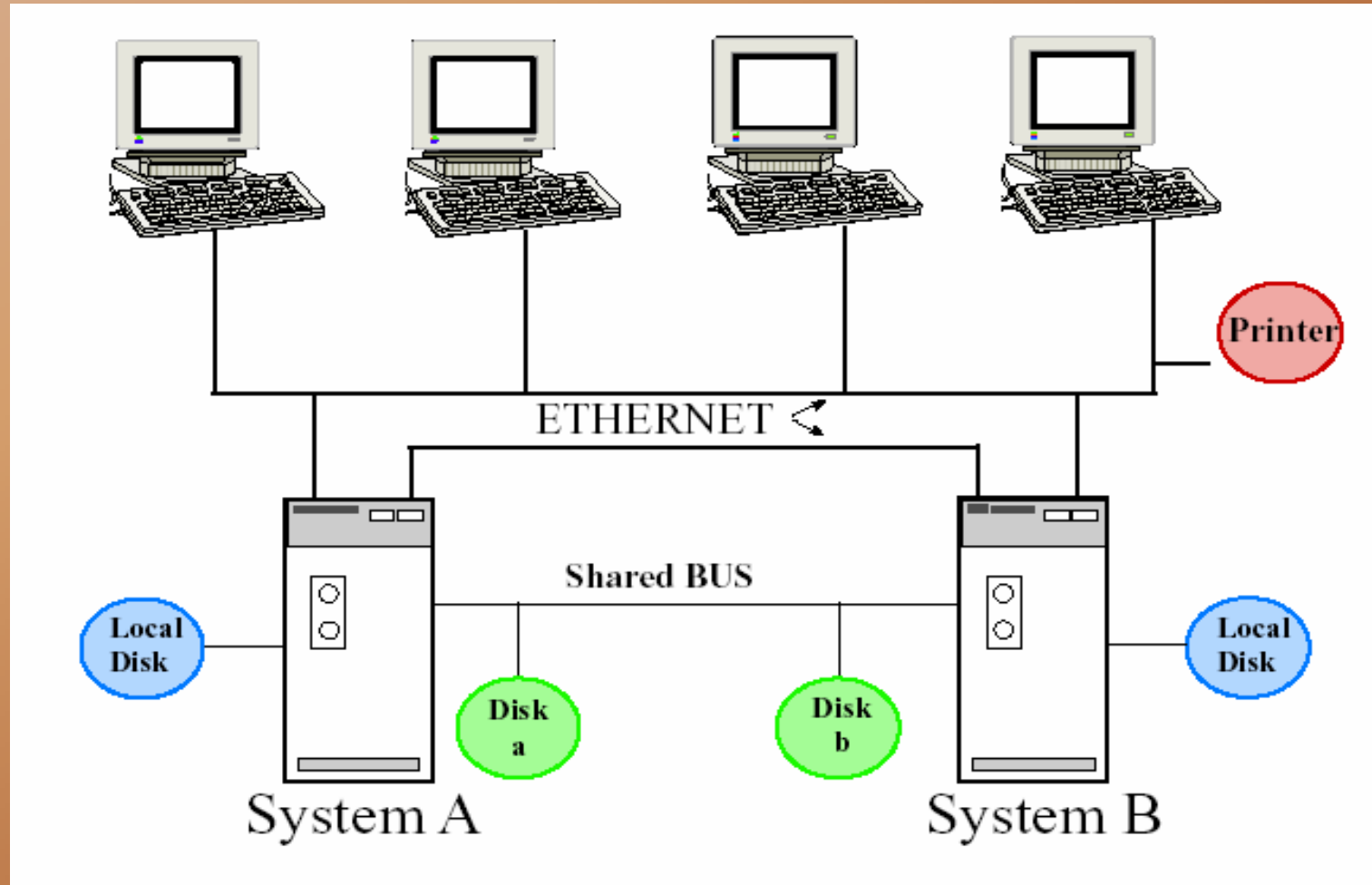


8.3.3 Model Cluster (tahap awal)

- Dua nodes (desain u/ N-node)
 - Keduanya aktif server (bukan pasif server)
- Failover protection u/ aplikasi-aplikasi server
- Aplikasi-2 secara eksplisit teregister sebagai pendukung
- Model storage dual-access
 - Kedua node dapat mengakses storage, satu server u/ suatu waktu (one node “owns” it)



Konfigurasi model

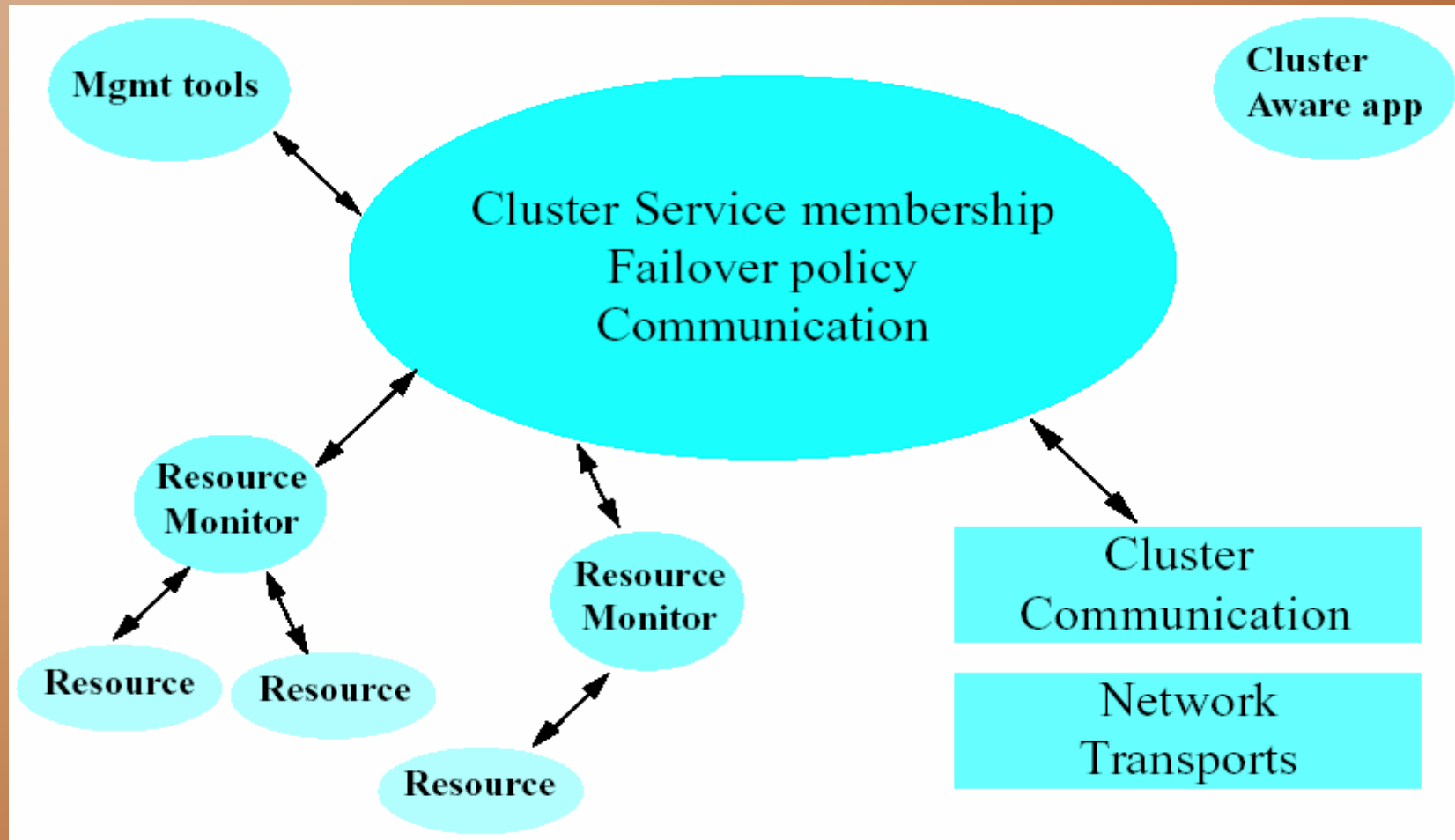


8.3.4 Desain dan Komponen

- Memisahkan layanan ketimbang modifikasi NT Kernel
 - Tidak mengganggu sistem dasar (base system)
 - Lebih u/ didesain dan didebug
 - Slight performance dan biaya ruang (space cost)
- Shared nothing
 - Konfigurasi HW disederhanakan Simplified hardware configuration
 - No shared file system
- Tools administratif yang sangat powerful
- Aplikasi-2 harus dapat di transaksikan



Model & Komponen (cont'd)



8.3.5 Resources

- Komponen yg menyediakan layanan dari clients ke sebuah client
- Lingkungan server environment spt:
 - physical disks,
 - processes, databases,
 - IP addresses, etc. ...
- Dimanage oleh cluster sbg obyek-2 Opaque
- Online on satu node pada suatu waktu
- Dapat berpindah satu sistem pd cluster ke sistem lain dalam cluster



Kebergantungan Resources

- Resources bisa tergantung pada resources yg lain
- Suatu resource di-online –kan jika beberapa resources bergantung padanya
- Sebaliknya di-offline-kan, jika tidak resources lain bergantung padanya
- Seluruh “dependent resources” akan failover bersama dan berada dalam grup yang sama



Resources Properties

- Resource Type
- Poll intervals
 - Looks Alive
 - IsAlive
- Private resource data
 - Unique identifier
 - Hardware binding
- Group Membership
 - Possible Nodes
 - Restart Policy
 - Dependencies

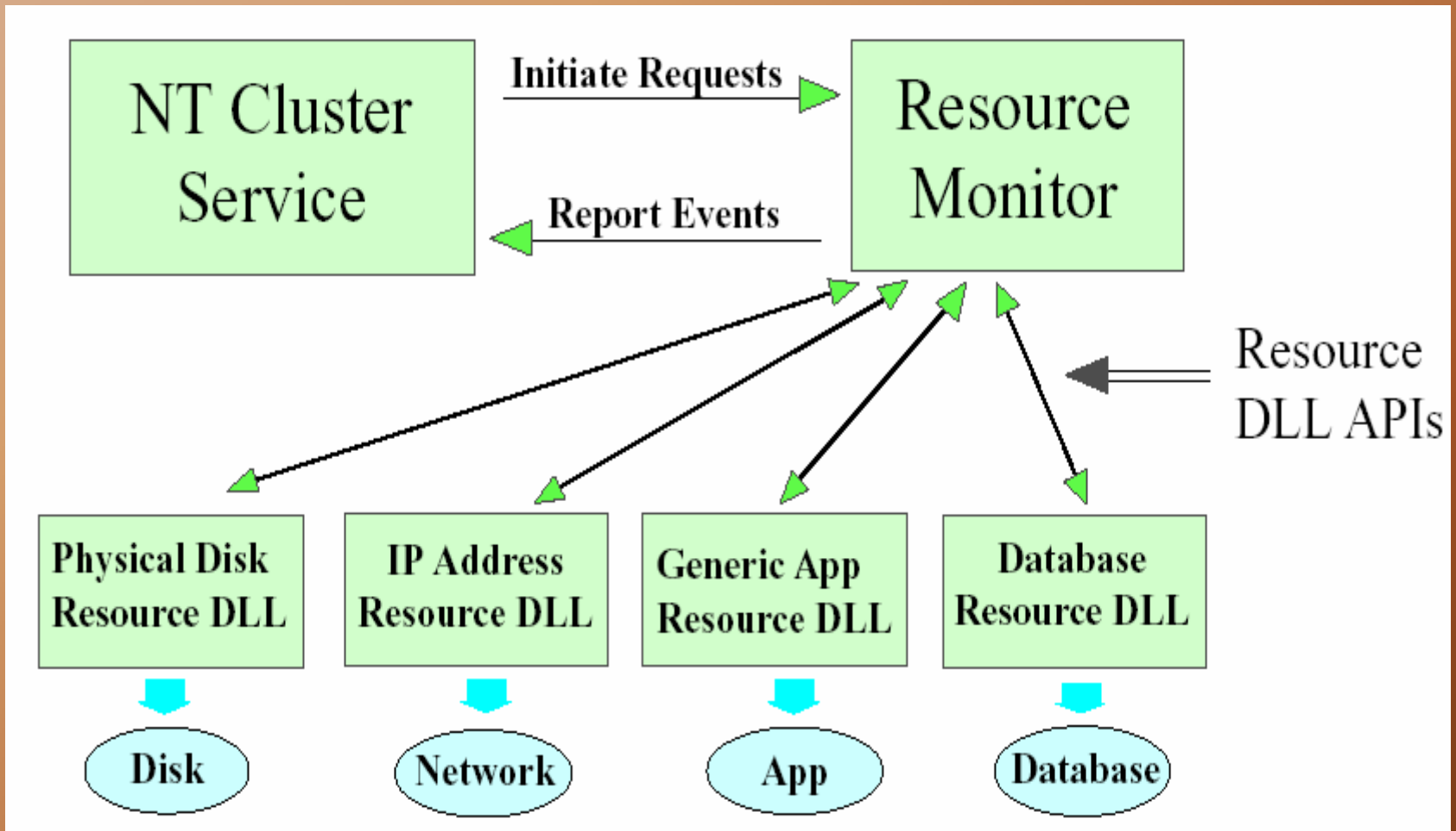


Resources DLLs

- Run pd proses monitor resource
- Kode spesifik resource yg mengizinkan SW-cluster u/ manage sebuah resource
- Resource DLLs disediakan oleh Microsoft sbg fitur dasar (base features)
- Penulisan yang mudah u/ meng-create aplikasi atau devais terkait/mendukung cluster



Resources DLL API's

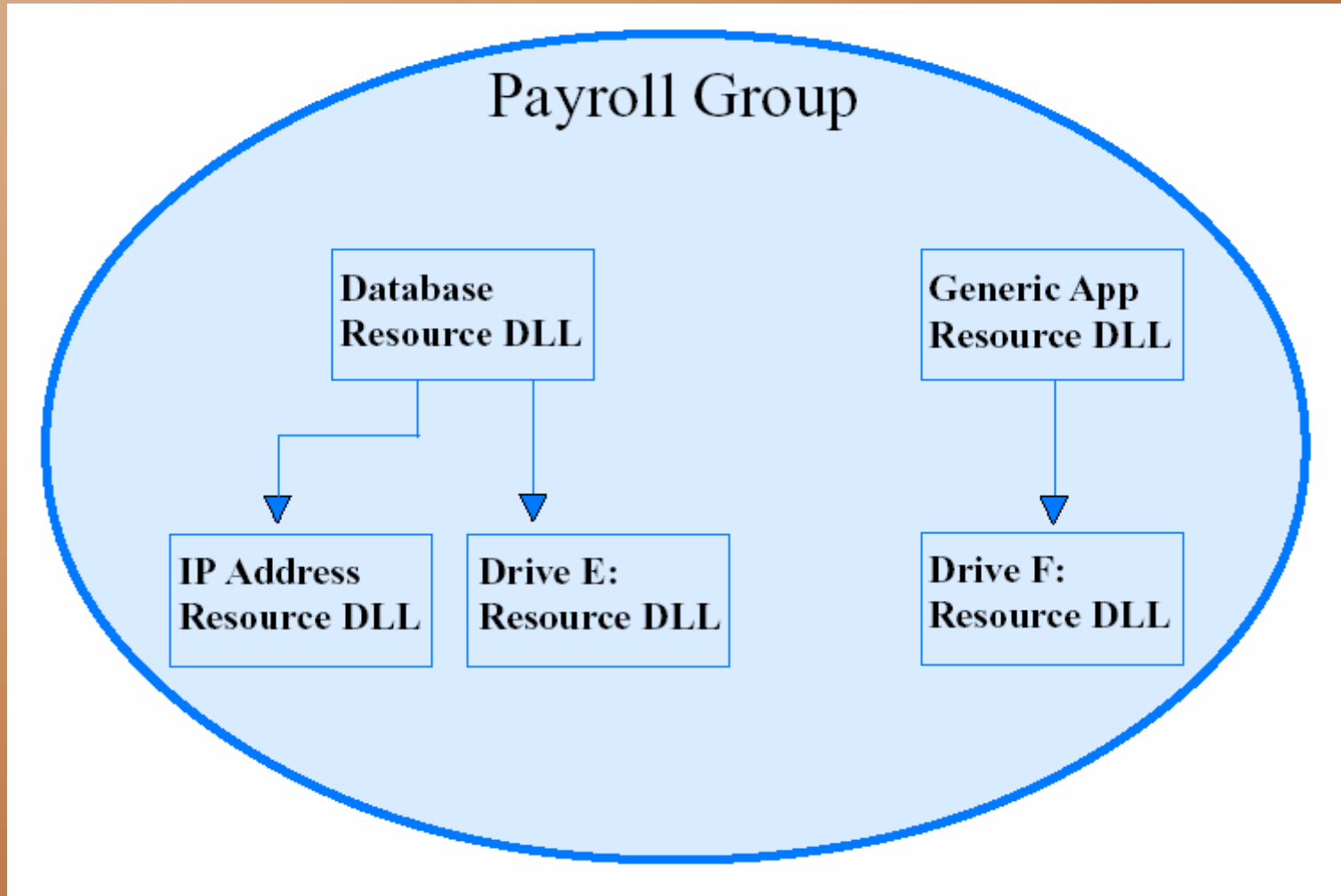


Resources DLL API's: Fungsi-Fungsi

- ◆ Startup
- ◆ Open
- ◆ Close
- ◆ Online
- ◆ Offline
- ◆ Terminate
- ◆ LooksAlive
- ◆ IsAlive
- ◆ Arbitrate
- ◆ Release
- ◆ ResourceControl
- ◆ ResourceTypeControl
- ◆ Create
- ◆ Delete



Contoh Grup

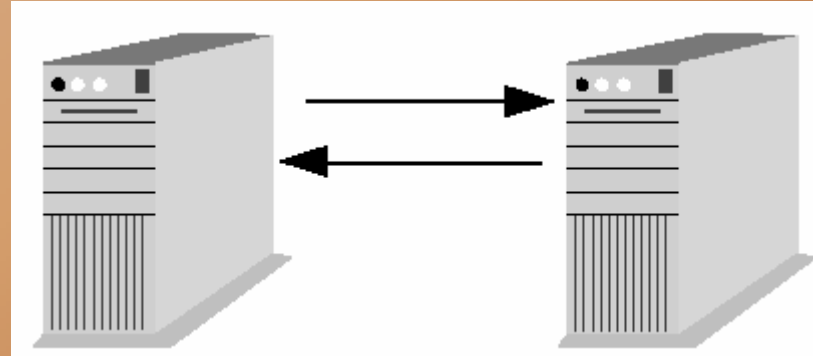


8.3.6 Interkoneksi

- Komunikasi Node-to-Node
- Komunikasi layanan cluster ke resources
- Komunikasi aplikasi ke layanan cluster

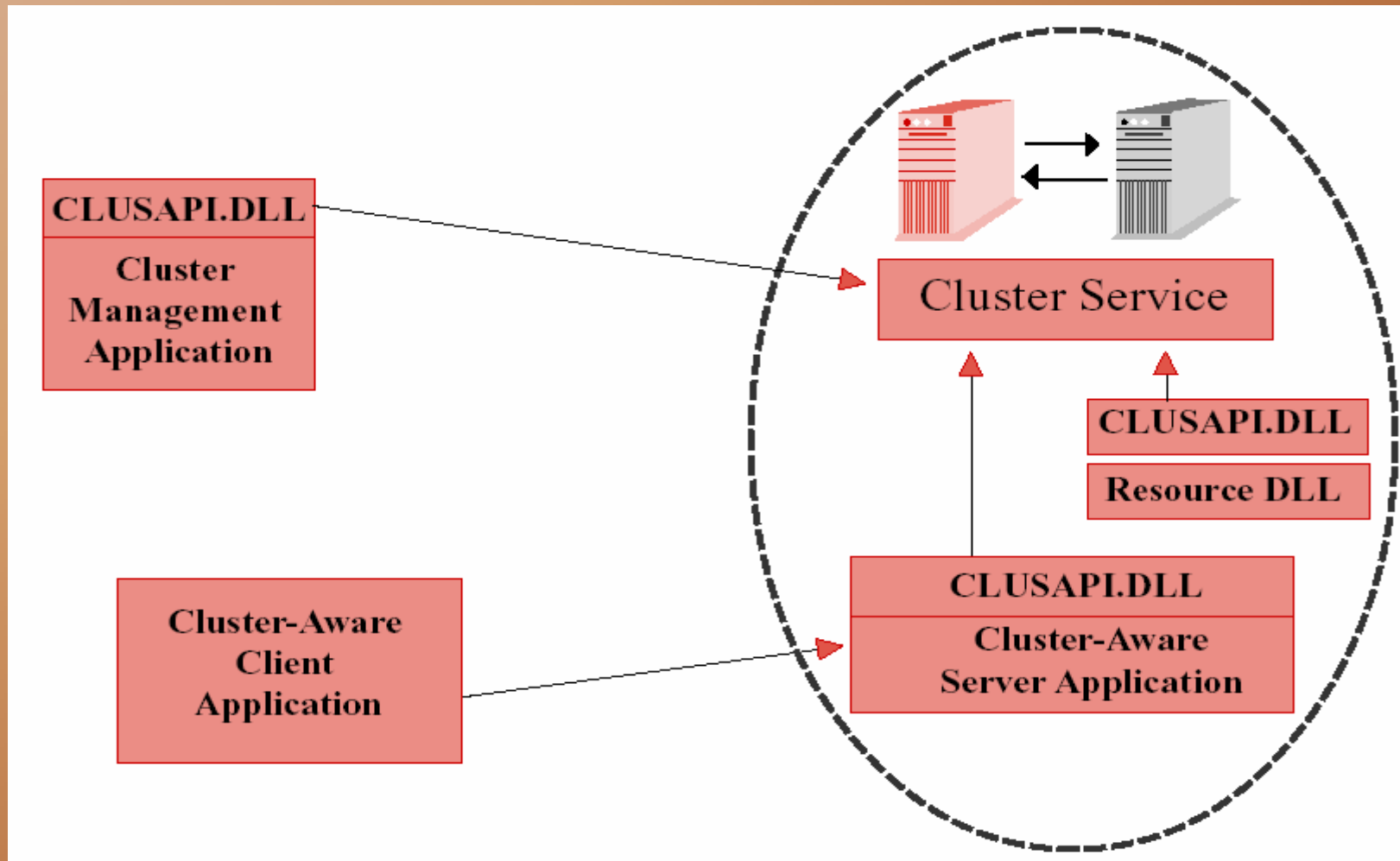


Node-to-node



- ◆ Layanan cluster berkomunikasi satu sama lain each other
- ◆ Notifikasi keepalive
- ◆ Keadaan resource mengubah broadcasts atomik
- ◆ Perintah intra-cluster

Aplikasi ke layanan cluster



8.3.7 Tools Manajemen Cluster

- Administrator cluster
 - Memonitor dan manage cluster
- Modifikasi-s minor thp tools yang ada
 - Monitor kinerja
 - § Menambah kemampuan u/ memperhatikan keseluruhan cluster
 - Administrator disk
 - § Meningkatkan pemahaman akan “shared disks”
 - Event logger
 - § “Broadcast events” ke seluruh nodes



8.3.8 Manajemen Aplikasi

- Banyak aplikasi berfungsi tanpa suatu modifikasi
 - Dukungan layanan-2 dan aplikasi-2 generik
 - Tools mengkonfigurasinya ke Wolfpack
- Dukungan yang mudah dari aplikasi lain applications
 - Wizard u/ meng-create suatu “resource DLL”
 - Wizard u/ meng-create ekstensi-2 administrator



CLUSAPI

- ◆ Manajemen cluster
- ◆ Manajemen node cluster
- ◆ Manajemen resource cluster
- ◆ Database konfigurasi cluster



Manajemen Cluster (cont'd)

- ◆ Mendapatkan dan berkomunikasi dgn cluster
- ◆ Properti-2 query/set
- ◆ Enumerate obyek-2 cluster
 - Nodes
 - Groups
 - Resources and Resource Types
- ◆ Notifikasi “event cluster”
 - Node state & property changes
 - Group state & property changes



8.4 Cluster Linux (Beowulf)



8.4.1 Motivasi

- Sistem komputer parallel → sangat mahal
- Digunakan terbatas → dept. pertahanan dan lembaga riset
- Penggunaan dan pemrogramannya cukup sulit → perlu skill khusus dan memahami pengetahuan mengenai arsitektur unik dari setiap mesin

SOLUSI masalah di atas → clustering

Pioner/pelopor clustering → ilmuwan NASA yang terlibat Beowulf Project



8.4.1 Motivasi (cont'd)

- Beowulf project (<http://www.beowulf.org/>):
 - Dimulai 1994 di NASA's Goddard Space Flight Center,
 - Menggunakan Linux dan SW open-source lainnya yang running pada PC tua dan tentunya murah.
 - Mendobrak pintu bagi komputasi cluster performansi tinggi dan biaya murah.
 - Disamping, standards dan tools tlh dikembangkan u/ sistem komputer paralel memori terdistribusi sedemikian, sehingga memudahkan programmer u/ mengembangkan aplikasi komputer yang “scalable dan “portable”
 - Pile of PC (POP)
- Karena murah → banyak universitas, lembaga riset, busines mengembangkan/membangun PC-clustering



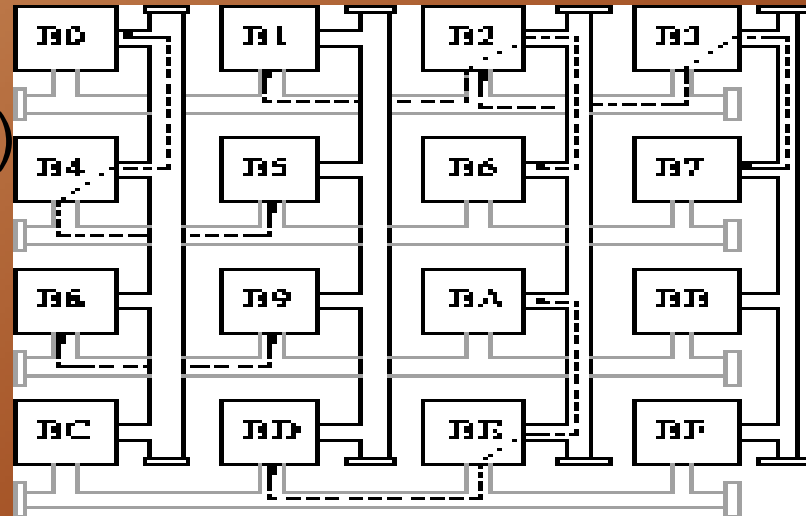
8.4.2 Aplikasi Target

- Komputasi Paralel → High performance paralel computing (scientific computation/computational science)



8.4.3 Model dan Arsitektur Cluster

- 16 Node dgn OS-Linux
 - Intel DX4 microprocessors at 100MHz
 - SiS471 chipset
 - 256K asynchronous cache
 - 16MB 60ns RAM
 - 540MB IDE disk
 - dual 10Mbps NICs (Network Interface Cards)
- Topology jaringan
(software routed network)



8.4.4 Sistem Software

- Berbasis Linux → Message Passing
 - Teknik yang berbeda dapat digunakan u/ pada Beowulf cluster atau platform komputer paralel lainnya termasuk
 - **Threads** atau **Inter-Process Communication (IPC)** pd sebuah node tunggal dengan MP, atau a single node with multiple processors, atau
 - **TCP sockets**,
 - **Remote Procedure Calls (RPCs)**, atau
 - Penukar message yang kurang sophisticated “files visible” pada multiple nodes.
 - Tetapi strategi yang mudah dan terbaik adalah menggunakan “software libraries” yang khusus didesain pada komputer paralel. Paling populer → PVM (Parallel Virtual Machine) dan MPI (Message Passing Interface).



8.4.5 Menggunakan MPI

- Langkah 1: download dan install library message passing Ipada seluruh node.
- Langkah 2: setelah instalasi software, mesin dipopulasi dgn list node yang tersedia.
- Langkah 3: Check documentation u/ mendapatkan dimana file ini harus diletakkan (reside)
- Berikutnya, kita harus familiar dgn sintaks dan semnatik dari penggunaan “MPI calls”.
- Metode paling umum dari programming aplikasi parallel disebut: Single Program Multiple Data (SPMD).
 - Jika program yang berbeda dapt ditulis u/ masing-2 yang bekerja single problem secara parallel, SPMD codes lebih mudah u/ ditulis dan dipelihara. Hanya contoh SPMD code akan dipresentasikan disini.



Contoh Program dgn MPI

Program 1 → "Hello World!" program → sbg ilustrasi MPI calls dasar yang penting u/ men-startup dan mengakhiri sebuah MPI program.

Program 1: hello.c

```
#include <stdio.h>
#include "mpi.h"
void main(int argc, char **argv)
{ int me, nprocs, namelen;
  char processor_name[MPI_MAX_PROCESSOR_NAME];
  MPI_Init(&argc, &argv);

  MPI_Comm_size(MPI_COMM_WORLD, &nprocs);
  MPI_Comm_rank(MPI_COMM_WORLD, &me);
  MPI_Get_processor_name(processor_name, &namelen);

  printf("Hello World! I'm process %d of %d on %s\n",
me, nprocs, processor_name);
  MPI_Finalize();
}
```



Keterangan program

- `MPI_Init()` harus di-called dan handed argument command line sehingga lingkungan men-setup secara benar u/ program agar run dgn proses parallel.
- `MPI_Comm_size()` → jumlah processes, yang secara subsequent disimpan di `nprocs`, pada grup komunikator `MPI_COMM_WORLD`.
 - `MPI_COMM_WORLD` → sebuah komunikator khusus yang menyatakan communicator seluruh processes tersedia ketika inialisasi.
- `MPI_Comm_rank()` menyediakan jumlah rank atau process (range dari 0 sampai `nprocs-1`) dari “calling process”.
 - Rank secara subsequen disipan pada `me`.
- `MPI_Get_processor_name()` menyediakan hostname dari node (bukan processor individual) yang digunakan, disimpan pada `processor_name`, panjang hostname disimpan di `namelen`

