High Performance Communication Subsystem for Clustering Standard High-Volume Servers Using Gigabit Ethernet

Wenzhang Zhu, David Lee, Cho-Li Wang

Department of Computer Science and Information Systems

The University of Hong Kong

Clustering Standard High-Volume Servers Using Gigabit Ethernet

- Large-scale application requires lots of computation and communication
  - Scientific computation: climate prediction, simulations
  - Commercial computation: large database systems
  - The Internet applications: parallel search engine, web servers, Internet VR, etc.
- **The only solution to handle the exponential growth of the computational demand is to build a system that can grow with it. Two suggested components:**
  - **SMP server:** SMP becomes the main building blocks of clusters (reduce the number of nodes; but reduce messaging overheads with shared memory)
  - **Gigabit Ethernet:** high performance, simplicity and low price make clustering on Gigabit Ethernet a cost efficient solution

Standard High-Volume Servers

- Coined by **Andy Grove**, Intel's CEO, 1994 speech at UniForum.
- Open server specification efforts in input/output, clustering, packaging and manageability (Intel's Enterprise Server Group)
- Enabled by killer microprocessor and symmetric multiprocessor system design
  - x86-based SMPs: Dell, Compaq, IBM, HP, ···
    - 2-, 4, 8-,··· processors
    - Standard, redundant power and cooling

Earliest SHV

- **Simply PCs**
  - Stocked with enough memory and disk to provide shared file storage and print spooling for networks of other personal computers.
- **Applications:**
  - Provide e-mail services for PC networks and to act as network gateways to legacy minicomputer and mainframe systems.

Issues on HVS Communication Subsystem

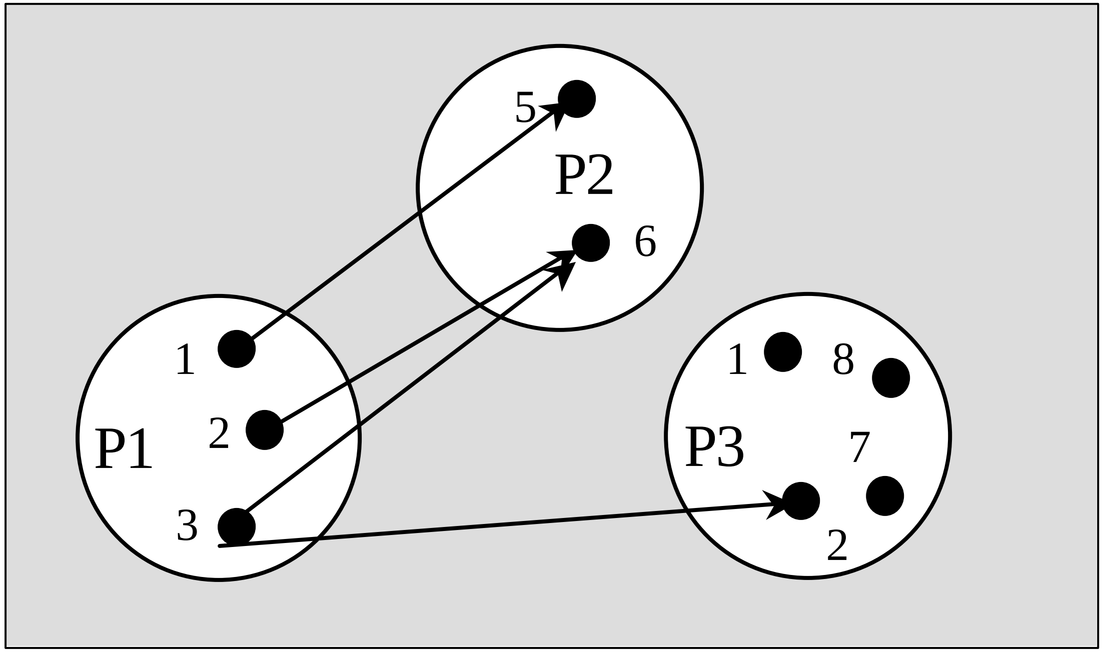
- Interface with good programmability
- Low resource consumption on the server
- High availability communication channel between HVS
- Multi-protocol support

Directed Point (DP) Model

- Depict the communication pattern using Directed Point Graph (DPG)

  - ***Directed Point graph*** (DPG) = (N, EP, NID, P, E)
    - ***N*** : Node set
    - ***EP*** : Endpoint set
    - ***P*** : Process set
    - ***NID*** : Node Identification function
    - ***E*** : Edge set

Example: some communication patterns in cluster



The DP program

P1

```
fd1=dp_open(1)
fd2=dp_open(2)
fd3=dp_open(3)
dp_target(fd1,1,5)
dp_target(fd2,1,6)
dp_target(fd3,1,6)
dp_target(fd3,1,2)
dp_read(fdx,⋯)
dp_write(fdx,⋯)
```

P2

```
fd1=dp_open(5)
fd2=dp_open(6)
dp_read(fdi,⋯)
dp_write(fdi,⋯)
```

P3

```
fd1=dp_open(1)
fd2=dp_open(2)
fd3=dp_open(7)
fd4=dp_open(8)
dp_read(fdi,⋯)
dp_write(fdi,⋯)
```

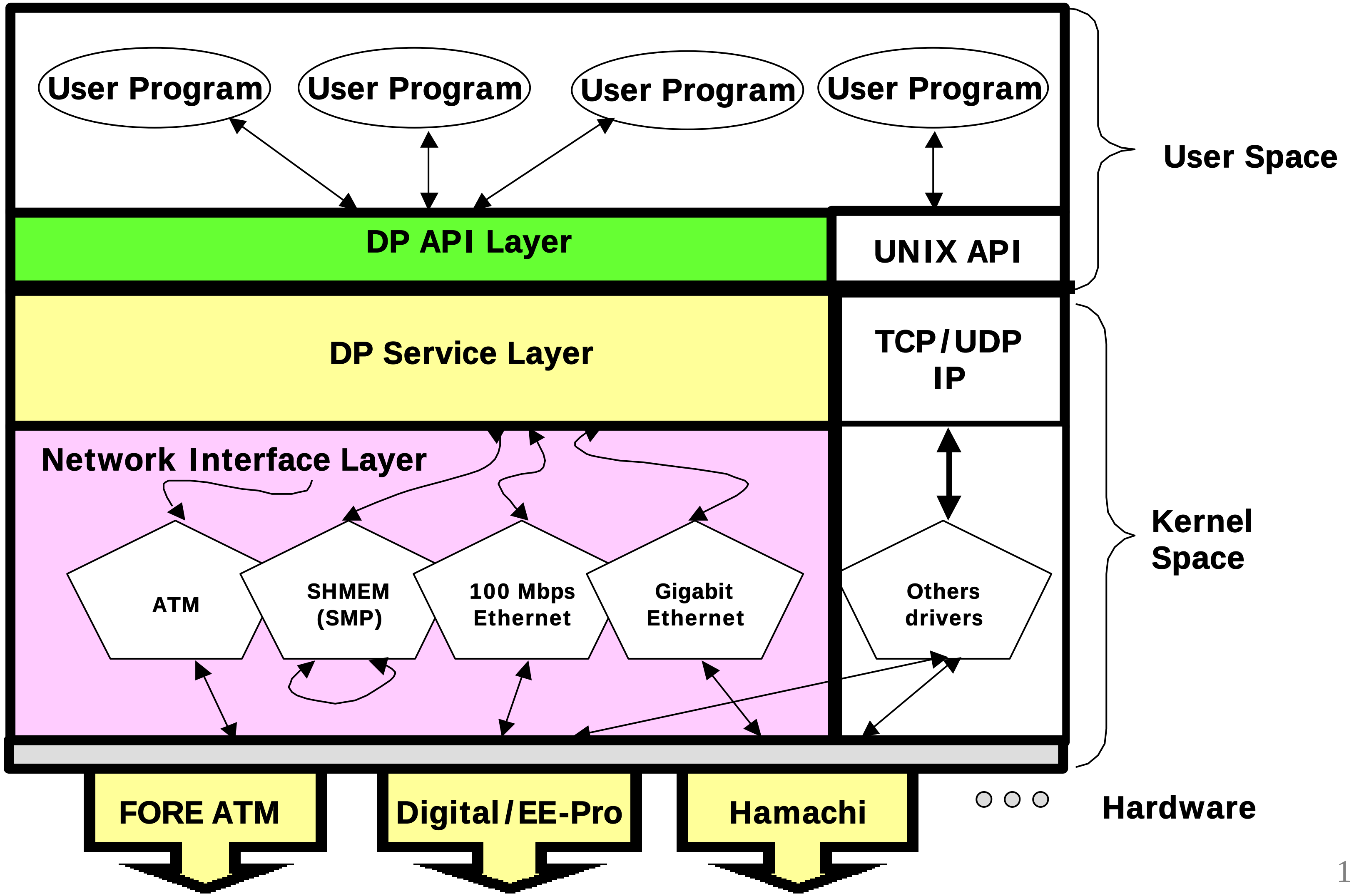
dp_open(local_dpid[pid]) /* create an endpoint */

dp_target(fd, remote_nid[pid], remote_dpid[pid])
/* make a connection with a remote endpoint */

Main Features of DP-II

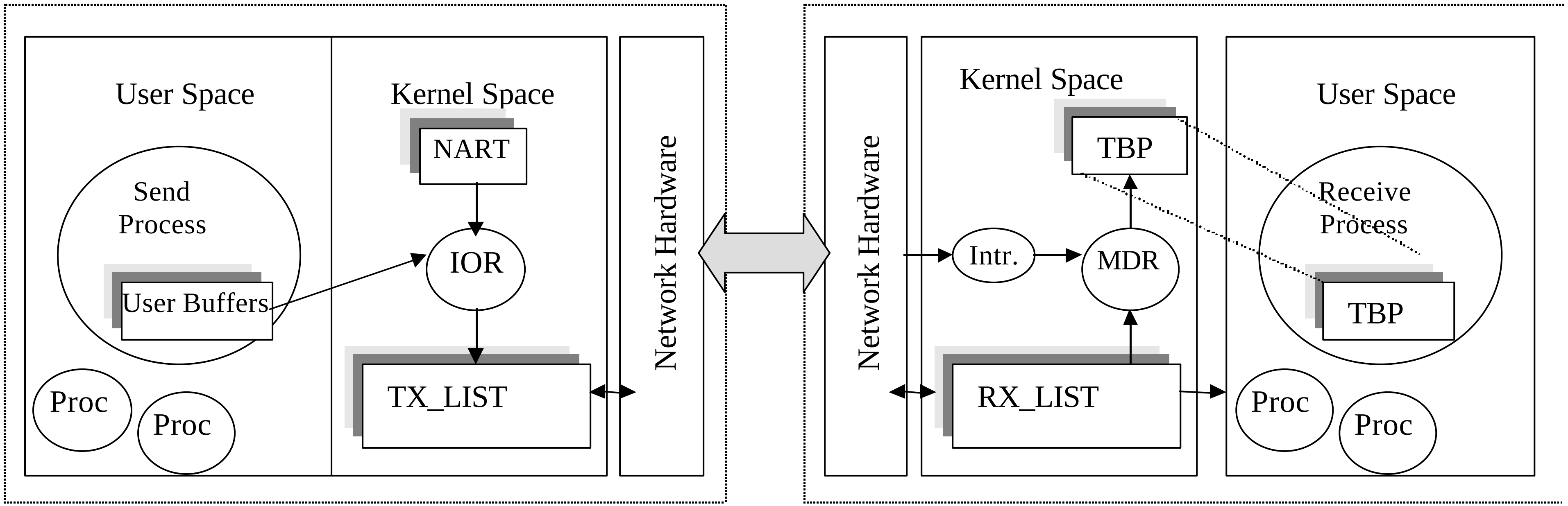
- Kernel Level Communication System
- UNIX I/O API
- Modular Design

DP-II Architecture

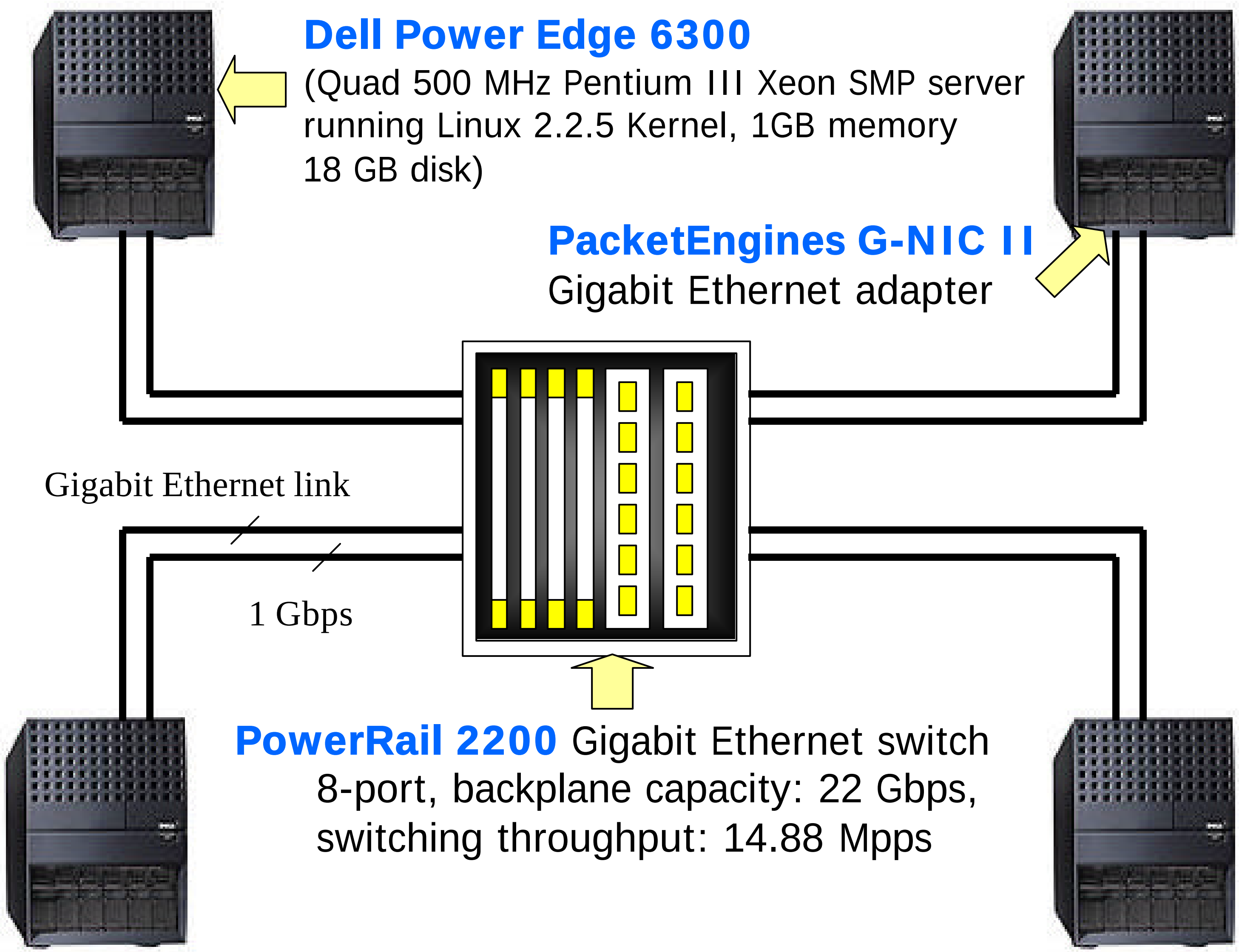


Light-Weight Messaging Techniques

- **Directed message**: use NID and DPID for multiplexing incoming packets.
- **Token Buffer Pool (TBP):** dedicated fixed-size buffer for a communication endpoint; accessible by both user and kernel threads
- **Lightweight messaging call** : fast path to enter kernel (x86 *call gate)*



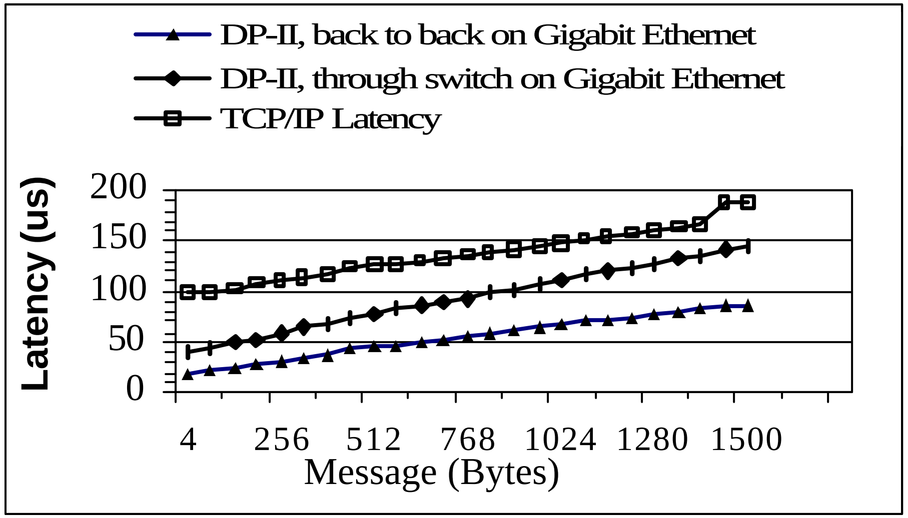
The HKU HVS Cluster



DP-II Performance

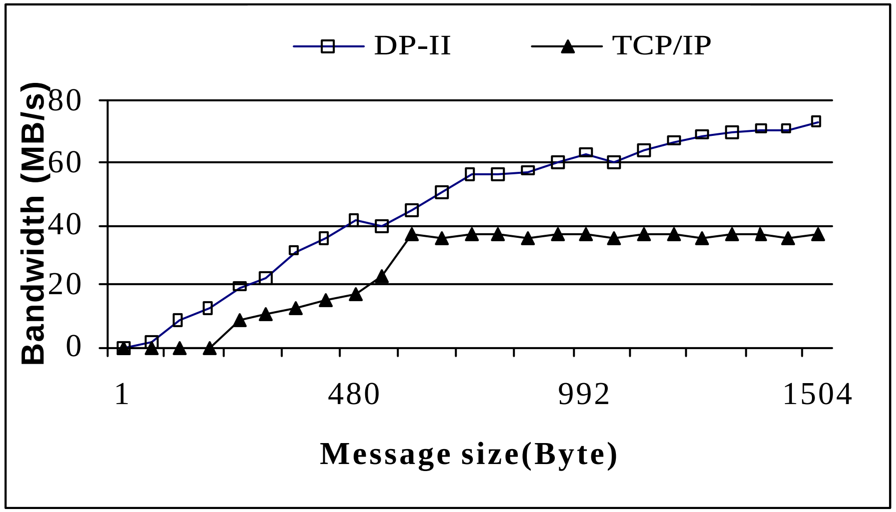
Single-trip Latency Test

(Min: 18.32 usec)



Bandwidth Test

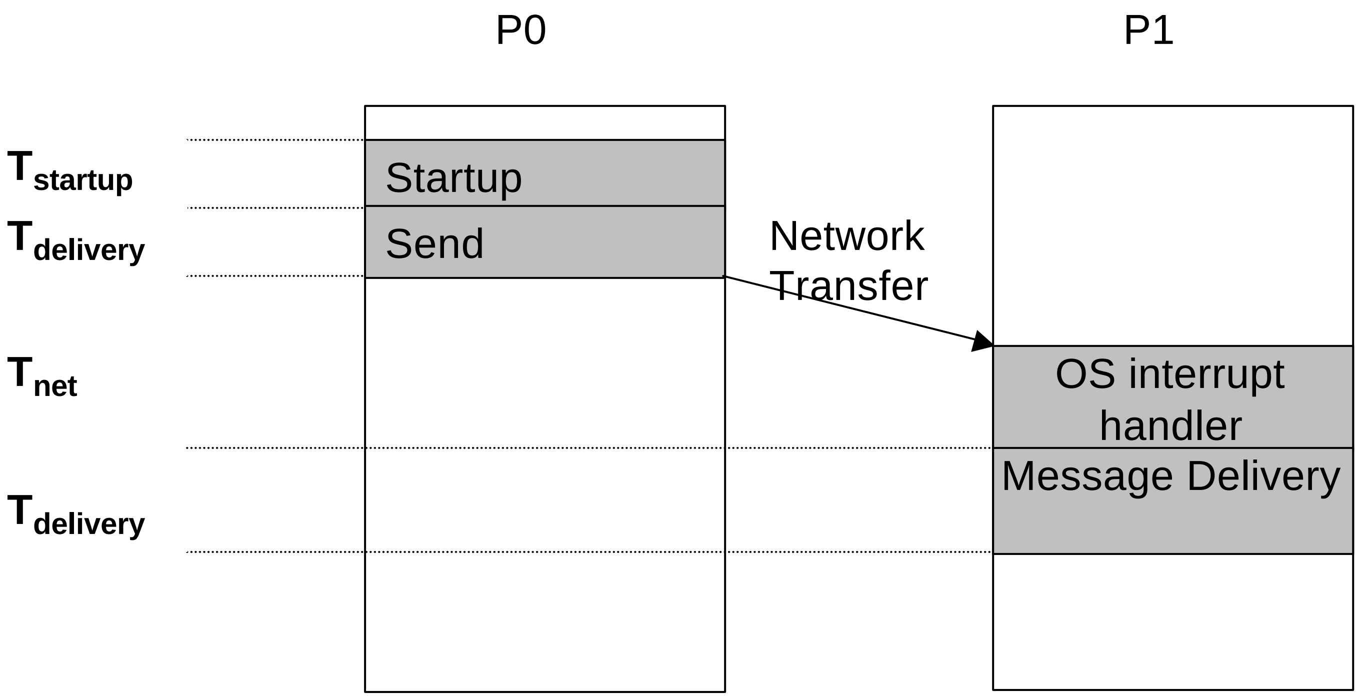
(Max: 72.8 MB/s)



Performance Analysis Model

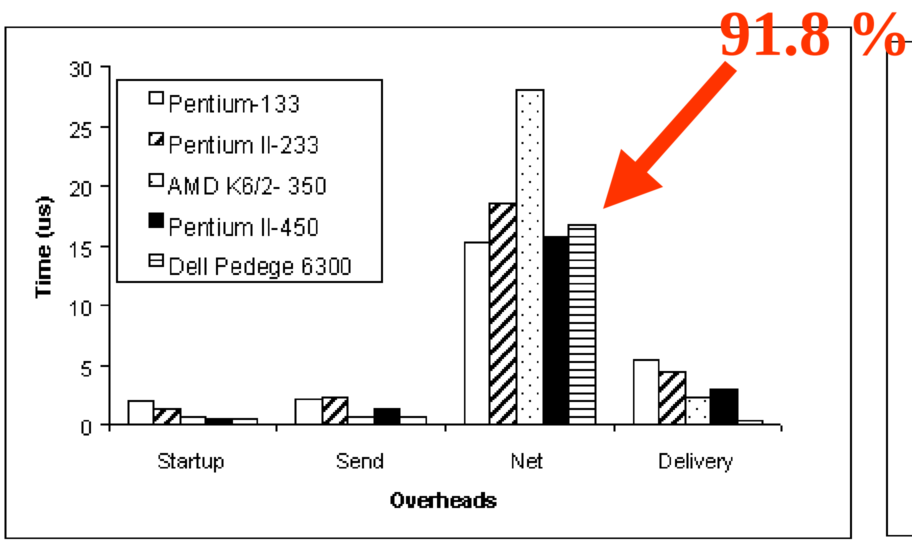
- latency breakdown

$$L(l) = T_{startup} + T_{send}(l) + T_{net}(l) + T_{delivery}(l)$$



Performance Analysis

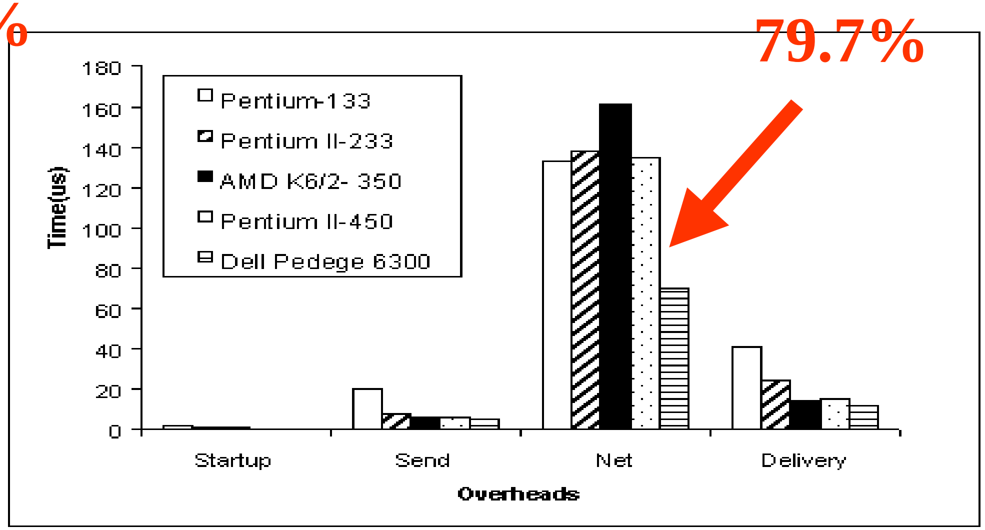
Total: 18.32 microseconds



Single-trip latency breakdown on sending a 1-byte message

(On Gigabit Ethernet, the $T_{startup}$, T_{send}, T_{net}, and $T_{delivery}$ are 0.44, 0.7, 16.82, and 0.36 us respectively)

Total: 87.84 microseconds



Single-trip latency breakdown on sending a 1500-byte message

($T_{startup}$, T_{send}, T_{net}, and $T_{delivery}$ are 0.44, 5.23, 69.96, and 12.21 us respectively.)

T_{net} : Major delay is contributed by the host PCI and the Hamachi NIC.

Performance Comparison

- **RWCP GigaE PM [3]**
  - **48.3 us** round-trip latency and **56.7 MB/s** on Essential Gigabit Ethernet NIC Pentium II 400 MHz.
- **RWCP GigaE PM II [2]**
  - on Packet Engines G-NIC II for connecting Compaq XP-1000 (Alpha 21264 at 500 MHz.) **44.6 us** round trip time. **98.2 MB/s** bandwidth.
- **HKU DP-II**
  - on Packet Engines G-NIC II : **36.64** us round-trip latency and **72.8 MB/s** on 4-way 500 MHz PIII Xeon.

Conclusions

- Processors, interconnect today exhibit extraordinary performance, making clustering the primary route to the extremes of performance. Clustering HVS using Gigabit Ethernet provides a cost efficient solution for deep parallel computing.
- The DP model is simple and easy to program the code.
- Major delay is contributed by the host PCI and the Hamachi NIC. The 66 MHz 64-bit PCI will help.
- Reducing memory copy overhead is essential when data is communicated in Gigabit speed.
- Efficient buffer management has become more critical then clustering standard PCs in 100 Mpbs Ethernet.

Future Work: HKU JESSICA Project