

Linux Cluster Architecture

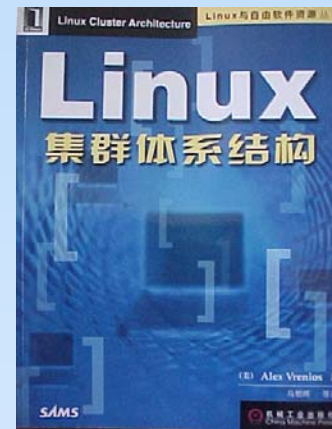
by

Alex Vrenios

(Shameless Plug)



English



Chinese

Slide # 1

Copyright © 2003 Alexander Vrenios

Linux Cluster Architecture

Overview:

- Why would anyone want to build a cluster system?
- Computer Architecture Review: UPs through Clusters
- Gathering the PC computer hardware (on the cheap!)
- Connecting the node computers into a local area network
- Local and remote software development file structure
- Configuring relevant Linux OS files for internetworking
- Subtasks and sockets for local or network programming
- The design of our master-slave cluster server software
- Internal and external performance monitoring and tuning

Linux Cluster Architecture

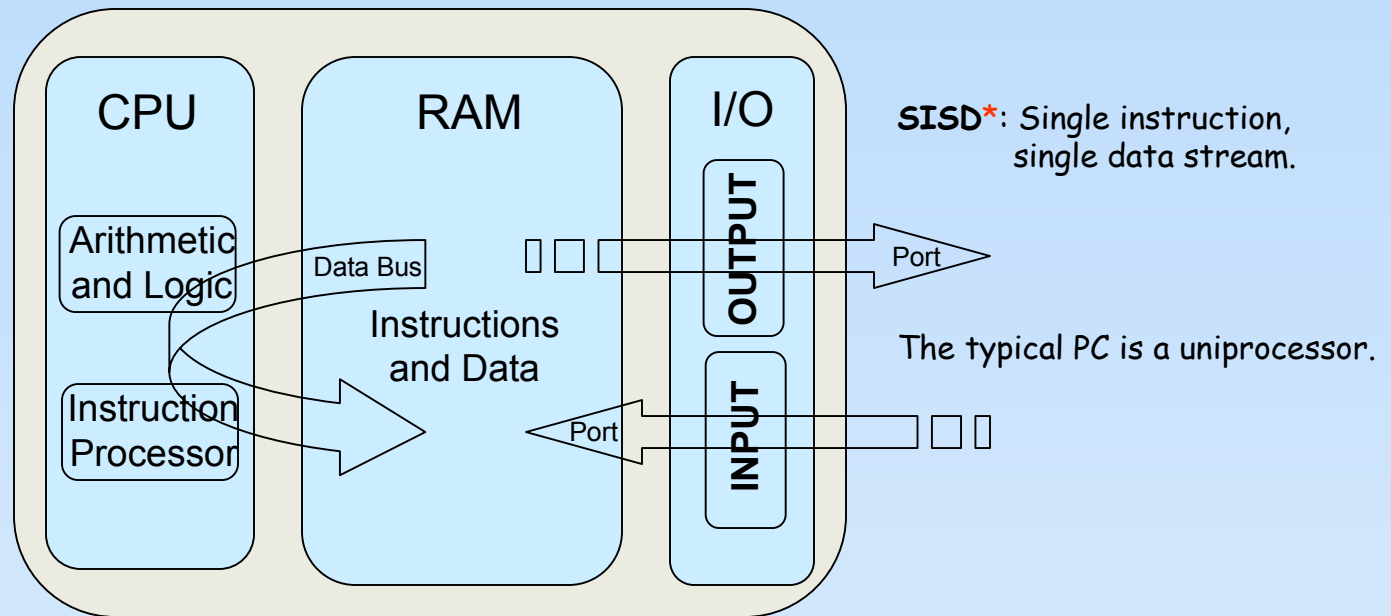
Why would anyone want to build a cluster system?

- Hobbyists:
It's a new and interesting pathway to experience! (And how many of your friends have a cluster server anyway?)
- Professionals:
Sophisticated systems are often developed in parallel, meaning the hardware won't be ready when you want to test your software. Having a test bed will get you past the hardware independent bugs, and put you in a position to polish your product when the platform is finally ready.
- Managers:
This is all bleeding edge stuff; you'll want to prepare for the issues your people might face and the questions they might ask. Experience gives you the insight you will need.
- Academics:
Analyze data from a live system, when you can, simulation can be over-simplified and its results can be misleading.

Linux Cluster Architecture

Computer Architecture Review:

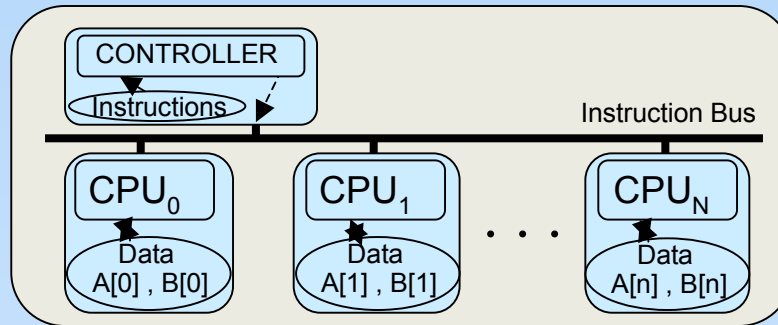
- Uniprocessor or UP



* Flynn proposed this taxonomy - some other configurations follow...

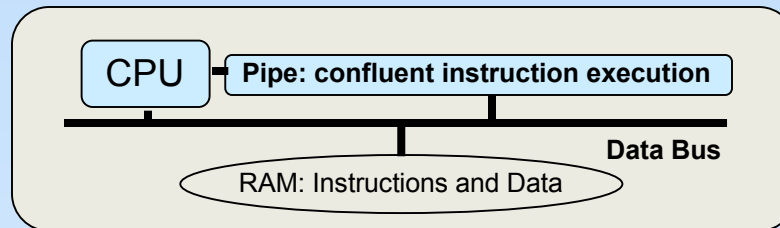
Linux Cluster Architecture

- **Array or Vector Processor**



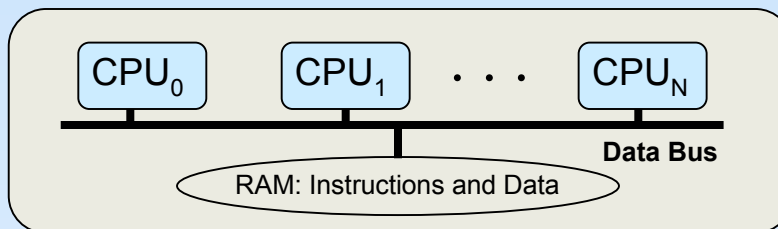
SIMD: Single Instruction, multiple data stream.
(ILLIAC IV, IBM 390, DSPs, etc.)

- **Pipeline Processor**



MISD: Multiple instruction, single data stream?
(Some say there is no MISD.)

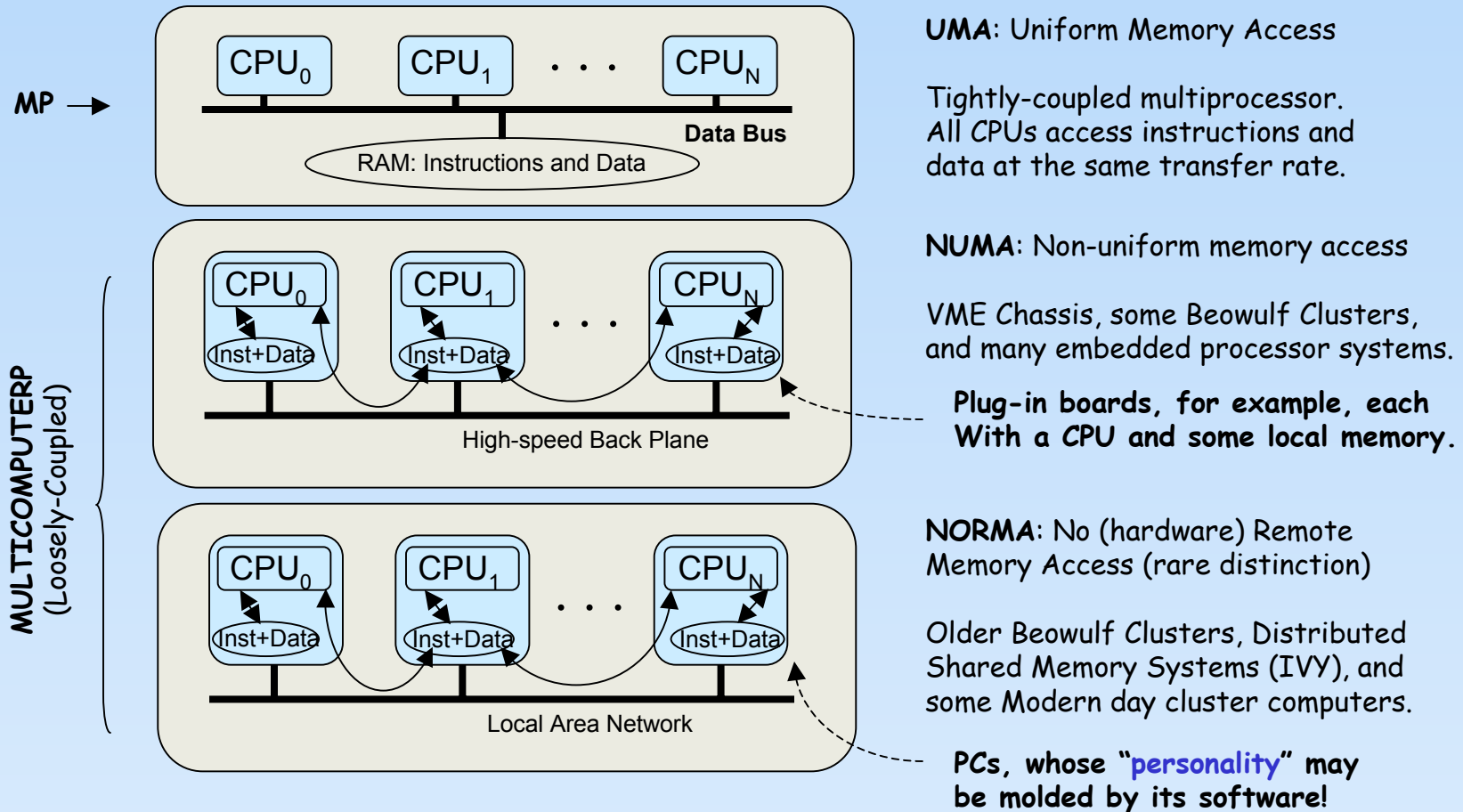
- **Multiprocessor or MP**



MIMD: Multiple instruction, multiple data streams.
Mainframe, Workstation, etc.
(Mostly for the very wealthy!)

Linux Cluster Architecture

- The MIMD is so interesting that gets its own taxonomy:



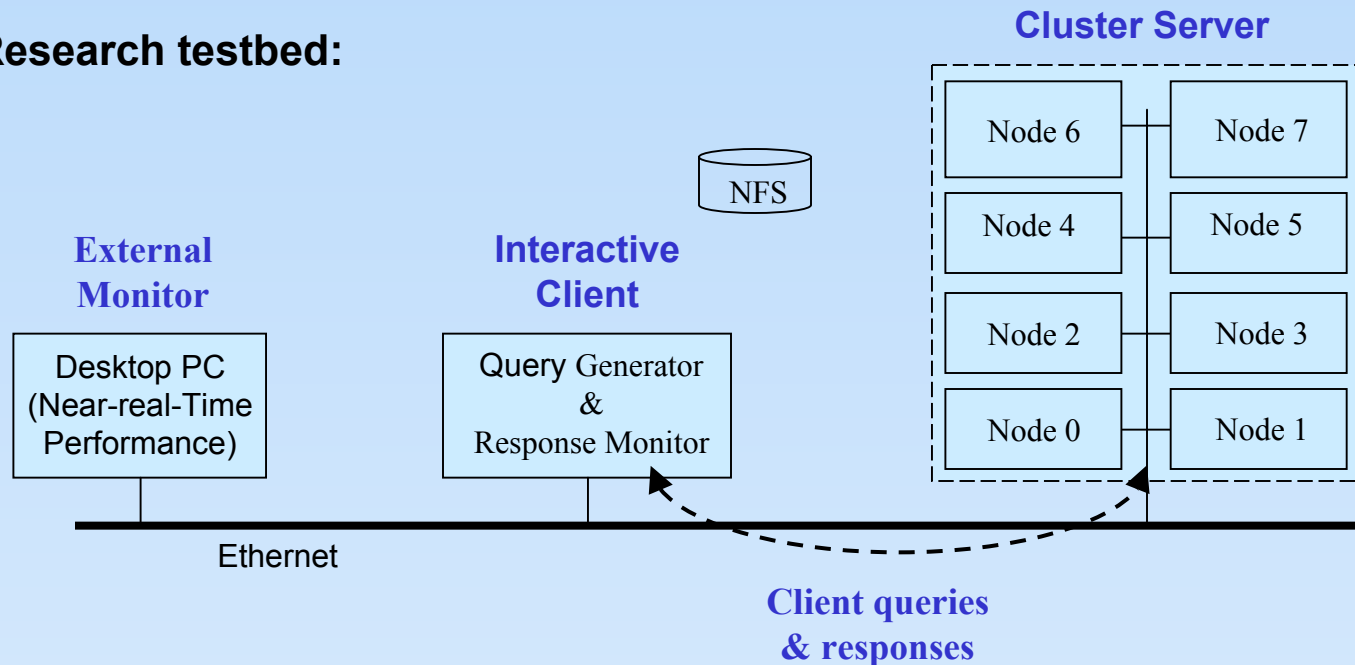
Slide # 6

Copyright © 2003 Alexander Vrenios

Linux Cluster Architecture

Network Block Diagram:

Research testbed:



Linux Cluster Architecture

CHAOS: Cheap Array of Outmoded Systems:

- Finding and networking N personal computers:

4-way KV Switches
(may be optional)

Network Activity
Monitoring

Surge Protectors



Development System



Books!

and NFS File Server

Linux Cluster Architecture

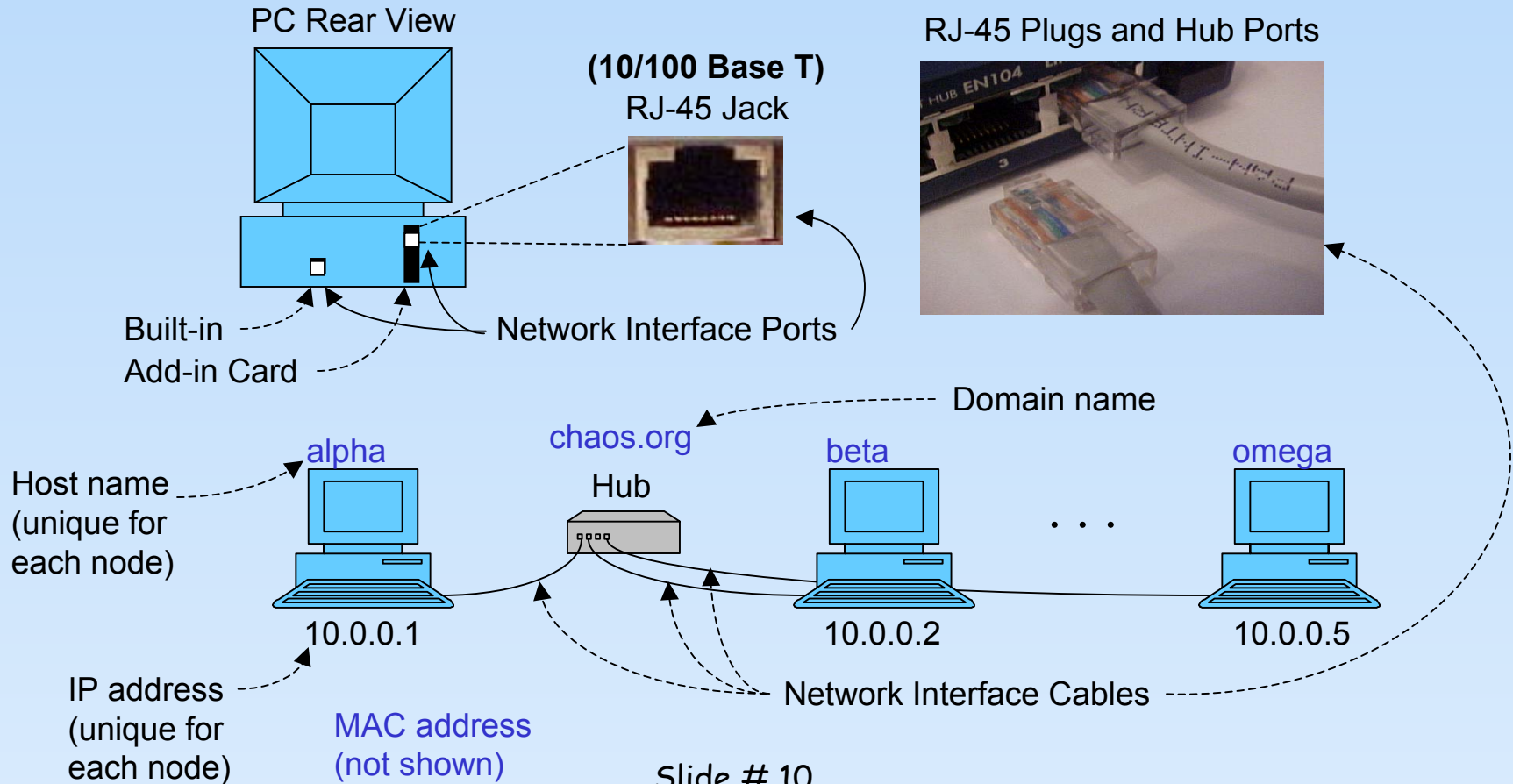
Gathering PC Computer Hardware:

- Small computer stores (Renaissance Computer, e.g.)
- Newspaper and club and organization newsletter ads
- Family, friends and neighbors (closets, garage sales)
- Large corporations? (hospitals, Am Exp, Mot, etc.)
- Computer salvage outlets:



Linux Cluster Architecture

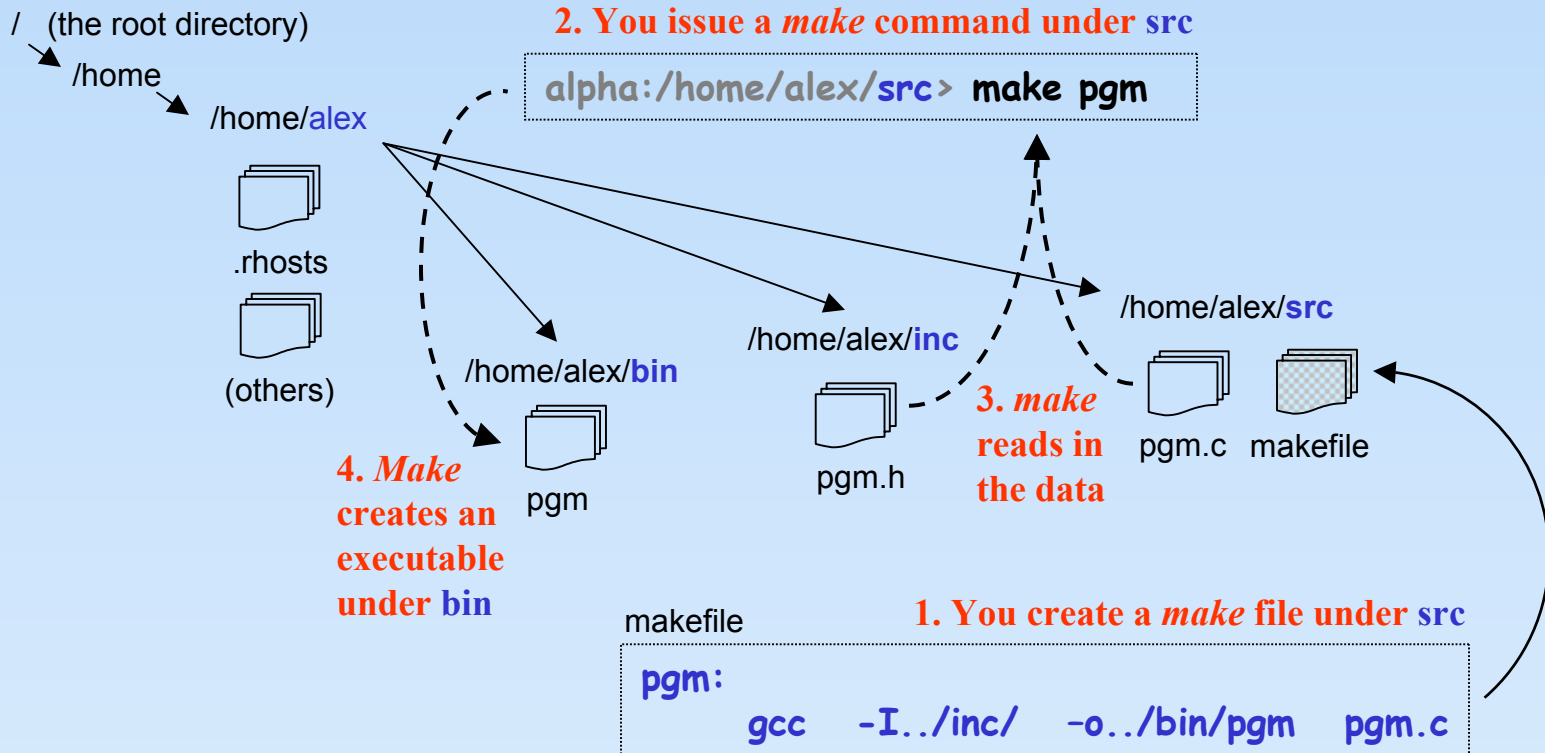
Connecting the Node PCs into a LAN:



Linux Cluster Architecture

Local User File Structure:

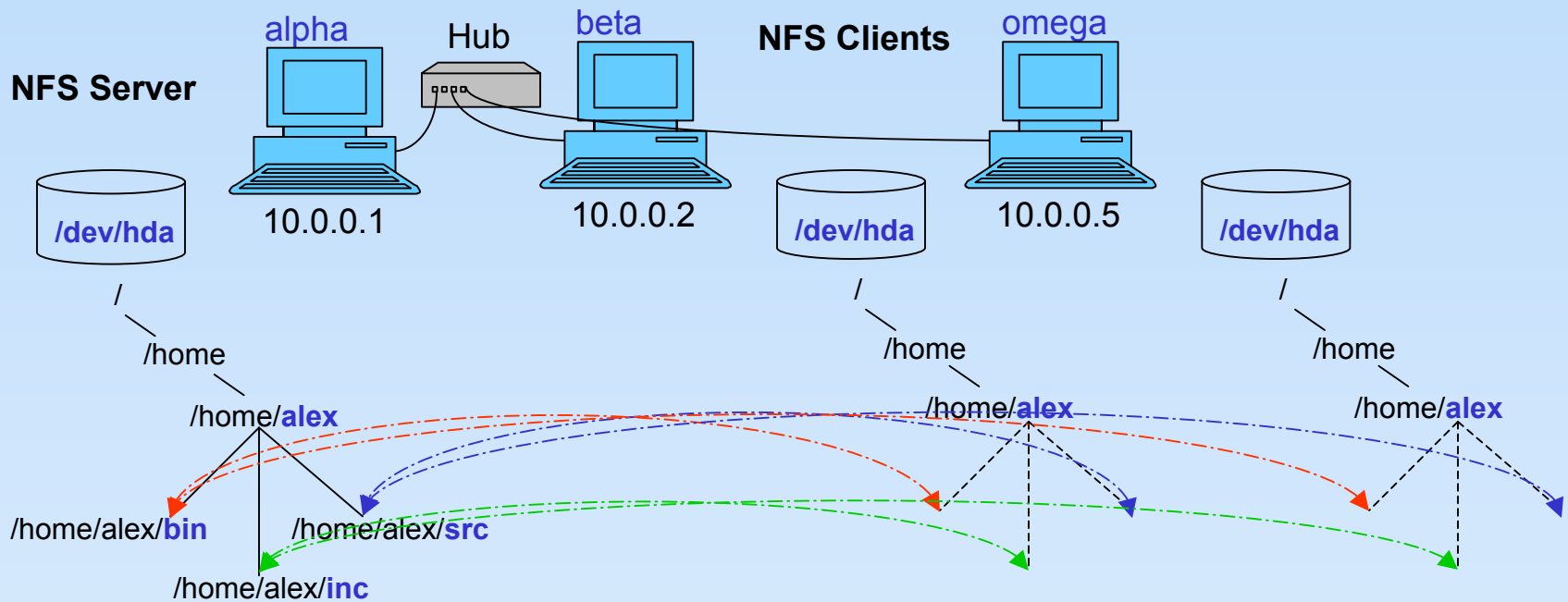
One way to organize your files for software development:



Linux Cluster Architecture

Remote User File Structure:

Network File System (NFS): the *illusion of locality* via remote-mount points



Linux Cluster Architecture

Linux OS Networking Files:

- First, file `/etc/hosts` belongs on all the network nodes:

127.0.0.1	localhost	localhost.chaos.org
10.0.0.1	alpha	alpha.chaos.org
...		
10.0.0.5	omega	omega.chaos.org

- Next, file `/etc/exports` on 10.0.0.1, the NFS server named alpha:

Server → `/home` (rw)

- Finally, file `/etc/fstab` on every node except the server named alpha:

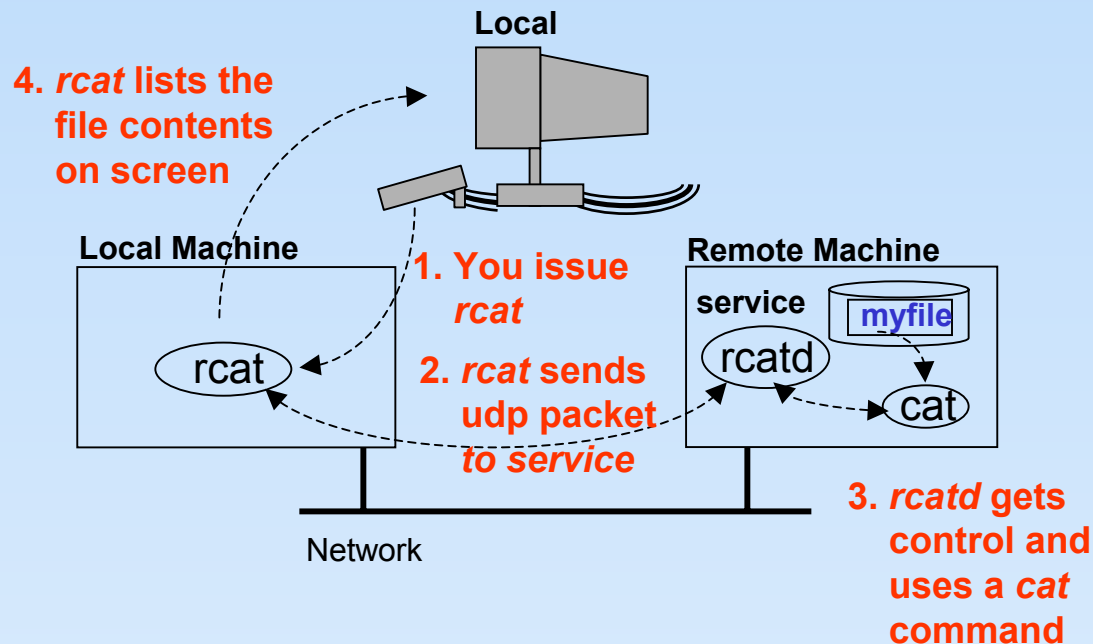
Clients →	<code>/dev/hda1</code>	<code>swap</code>	<code>swap</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
	<code>/dev/hda2</code>	<code>/</code>	<code>ext2</code>	<code>defaults</code>	<code>1</code>	<code>1</code>
	<code>10.0.0.1:/home</code>	<code>/home</code>	<code>nfs</code>	<code>rw</code>	<code>0</code>	<code>0</code>
	<code>/dev/fd0</code>	<code>/mnt/floppy</code>	<code>ext2</code>	<code>noauto</code>	<code>0</code>	<code>0</code>
	<code>none</code>	<code>/proc</code>	<code>proc</code>	<code>defaults</code>	<code>0</code>	<code>0</code>

Linux Cluster Architecture

UDP or TCP
socket

Internetworking Services: Operation

The *illusion of locality* via internetworking services



Slide # 14

Copyright © 2003 Alexander Vrenios

Linux Cluster Architecture

Internetworking Services: Configuration (using inetd)

- Add a line to file /etc/services on each remote-server node:

→ `rcatd 5000/udp # remote-cat UDP service on port 5000`

Refers to
entry in
services



- Add a line to file /etc/inetd.conf on each remote-server node:

→ `rcatd dgram udp wait alex /home/alex/bin/rcatd`

- Reconfiguration: `omega:/root> killall -HUP inetd`

Sequence of events:

1. Client process sends a UDP packet to server's port 5000
2. Daemon (inetd) starts process at /home/alex/bin/rcatd
3. Service reads incoming UDP packet data from "keyboard"

Linux Cluster Architecture

Internetworking Services: Configuration (using xinetd)

- File `/etc/xinetd.d/rcatd` on each (xinetd) remote-server node:

```
service rcatd
{
    port                = 5000
    socket_type         = dgram
    protocol            = udp
    wait                = yes
    user                = alex
    server              = /home/alex/bin/rcatd
    only_from           = 10.0.0.0
    disable             = no
}
```

Refers to name of service

Means 10.0.0.*

- Reconfiguration: `omega:/root> /etc/rc.d/init.d/xinetd restart`

Linux Cluster Architecture

Distributed Systems C-Language Skills:

Subtasking on a single processor
(details are in the book)



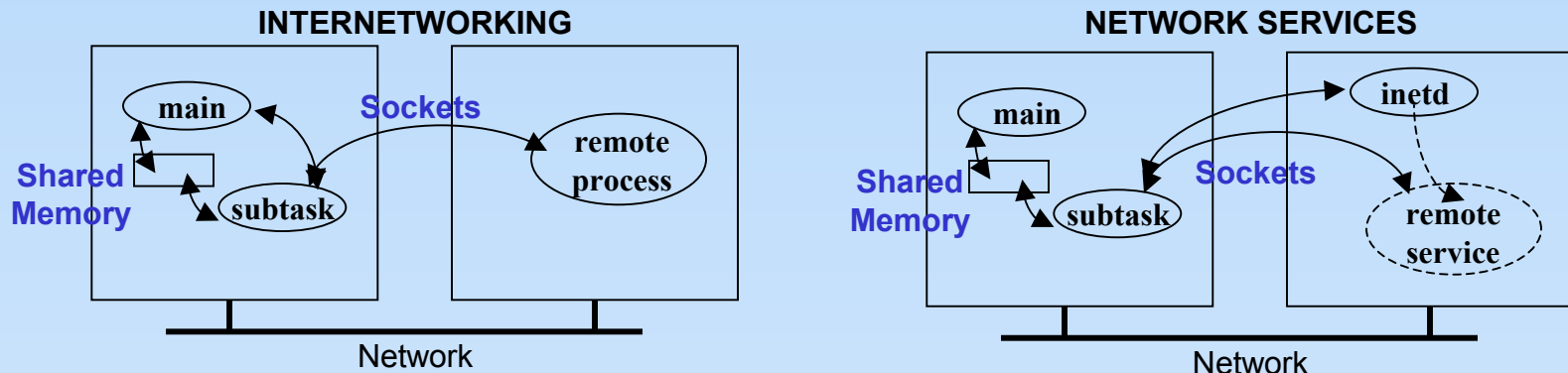
- Shared memory and semaphores
- Signal header/structure
- Handler initialization
- Signal processing and handler re-initialization

Slide # 17

Copyright © 2003 Alexander Vrenios

Linux Cluster Architecture

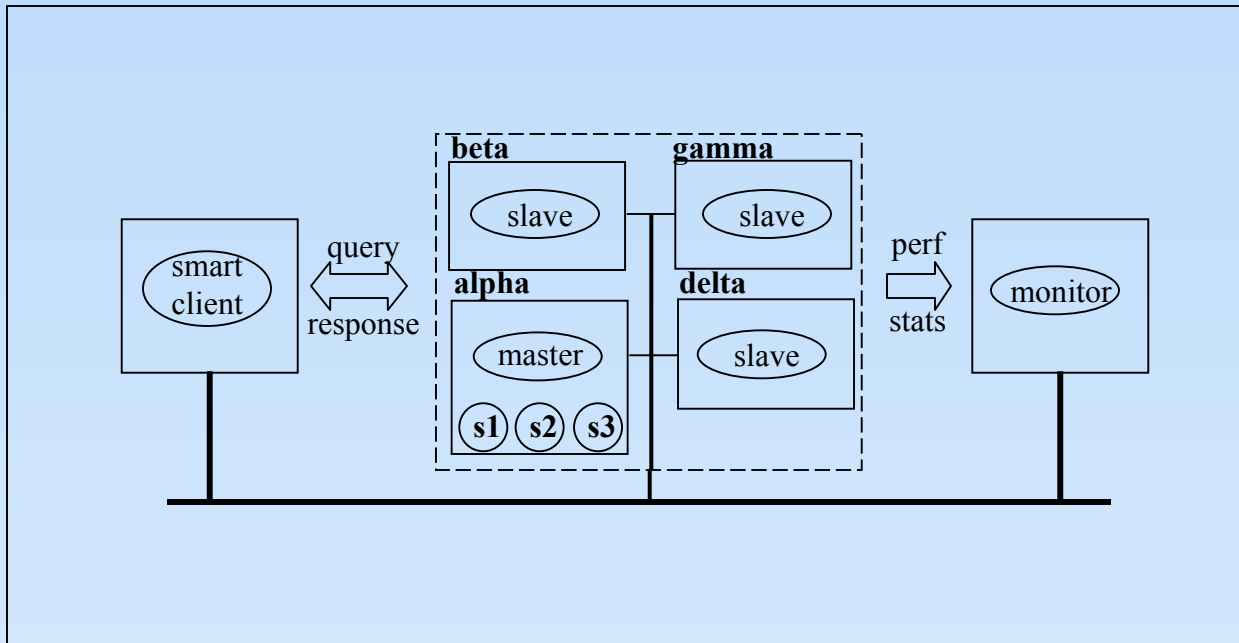
Distributed Systems C-Language Skills:



- Internetworking headers/structures
- Socket initialization
- Remote service "discovery"
- Reliable communications (topic discussions, anyway)

Linux Cluster Architecture

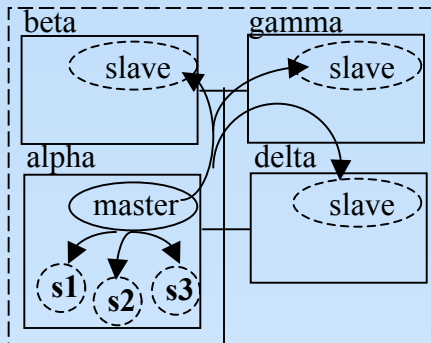
Master-Slave Cluster Server - Overview:



Linux Cluster Architecture

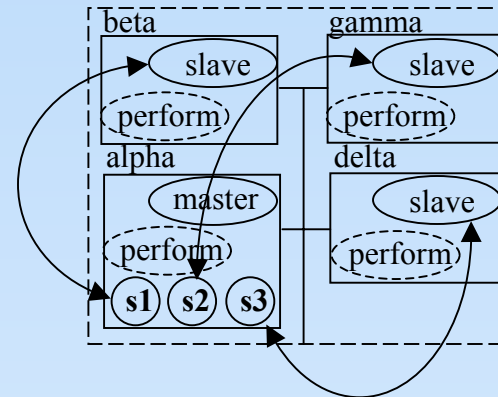
Master-Slave Cluster Server - Initialization:

Broadcast starts *slave* tasks...



master starts local subtask, one for each registering remote slave.

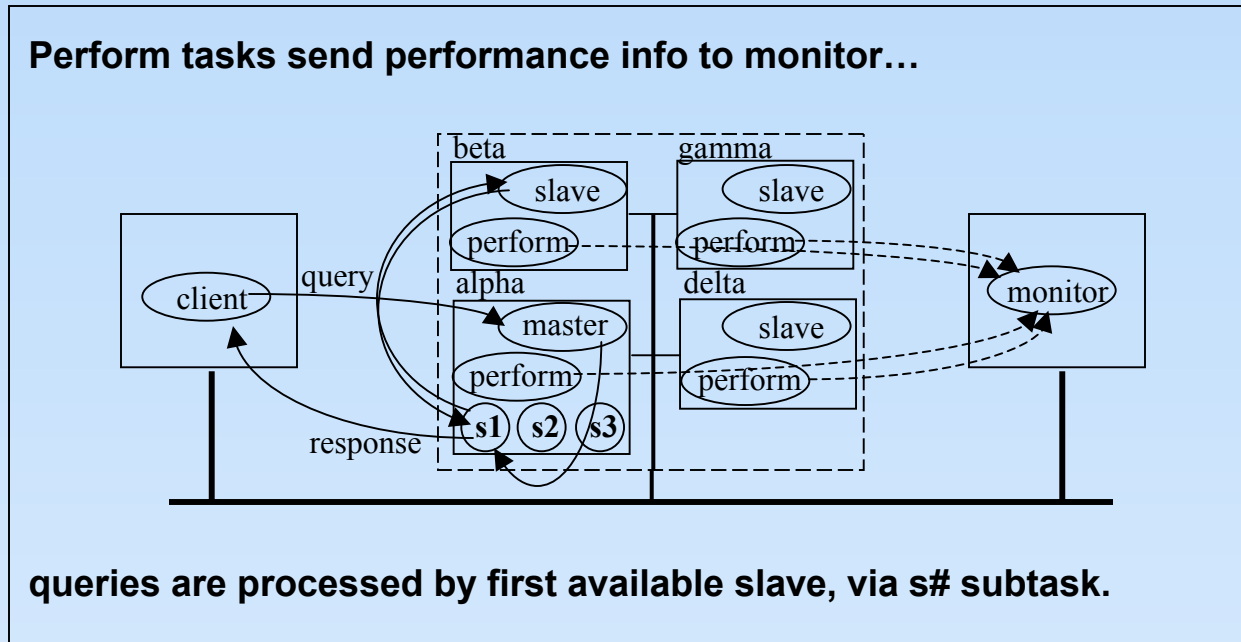
Local subtasks contact slaves...



all tasks start a *perform* subtask.

Linux Cluster Architecture

Master-Slave Cluster Server - Operation:



Linux Cluster Architecture

Performance Statistics - Pseudo-file Text*:

- CPU Utilization in /proc/stat - Running Jiffy Counts in each State

```
cpu 1256 0 1566 565277
```

Idle (since boot)
system
nice
user

- Disk Reads and Writes in /proc/stat - Running I/O Counts

```
disk_rio 1270 0 0 0  
disk_wio 1337 0 0 0
```

I/O count (since boot)

* Note that the exact meaning of proc file content can be OS release dependent: see [man proc](#)

Linux Cluster Architecture

Performance Statistics - More Pseudo-file Text:

- Memory Utilization in /proc/meminfo - Current Values

```
Mem: 14942208 13713408 1228800 ...
```

Free (right now)
Used
Total

- Packets Sent and Received in /proc/net/dev - Running I/O Counts

```
lo: 80 0 0 0 0 80 0 0 0 0  
eth0: 115 0 0 0 0 68 0 0 0 0
```

Transmitted (since boot)
Received

Linux Cluster Architecture

Internal Performance Statistics - Display:

NEAR REAL-TIME CLUSTER PERFORMANCE STATISTICS

```

                                10Base2
+----ALPHA-----+           |           +----BETA-----+
|  Cpu    Mem  |           |           |  Cpu    Mem  |
|   7%   94% |Rcvd 0       |    21 Rcvd|   28%   40% |
|  Rio    Wio  +-----+-----+   Rio    Wio  |
|   1     0  |Sent 12      |    1 Sent|   0     1  |
+----10.0.0.1---+           |           +----10.0.0.2---+
                                |
+----GAMMA-----+           |           +----DELTA-----+
|  Cpu    Mem  |           |           |  Cpu    Mem  |
|   2%   75% |Rcvd 2       |    0 Rcvd|   5%   56% |
|  Rio    Wio  +-----+-----+   Rio    Wio  |
|   4     0  |Sent 0       |   10 Sent|   3     0  |
+----10.0.0.3---+           |           +----10.0.0.4---+
                                |
                                chaos.org

```

- Overall Network Loading -
23 Pkts/sec

Slide # 24

Copyright © 2003 Alexander Vrenios

Linux Cluster Architecture

Performance Monitoring - External (displayed):

- Resource utilization reporting via a smart client process:

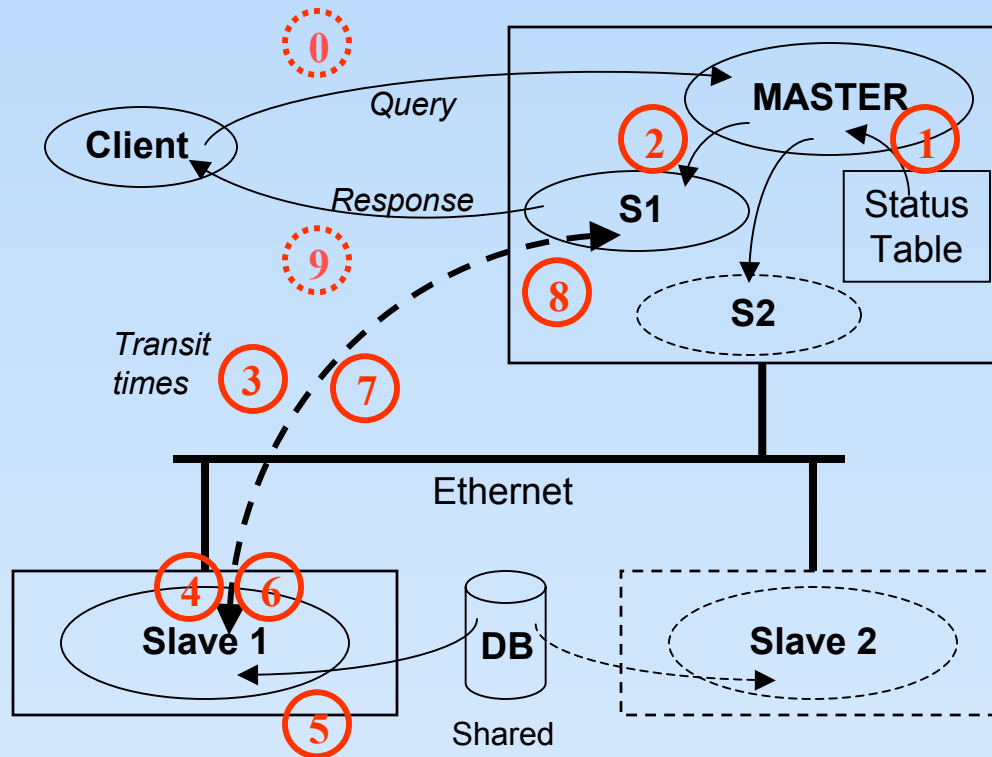
RESPONSE		OBSERVATIONS
TIME (msec)		10 20 30 40 50
1	10	
11	20	
21	30	*****
31	40	*****
41	50	**
51	60	
61	70	
71	80	
81	90	
91	100	

50 Total Observations

Average = 30 milliseconds ...what if you're not happy with this level of performance?

Linux Cluster Architecture

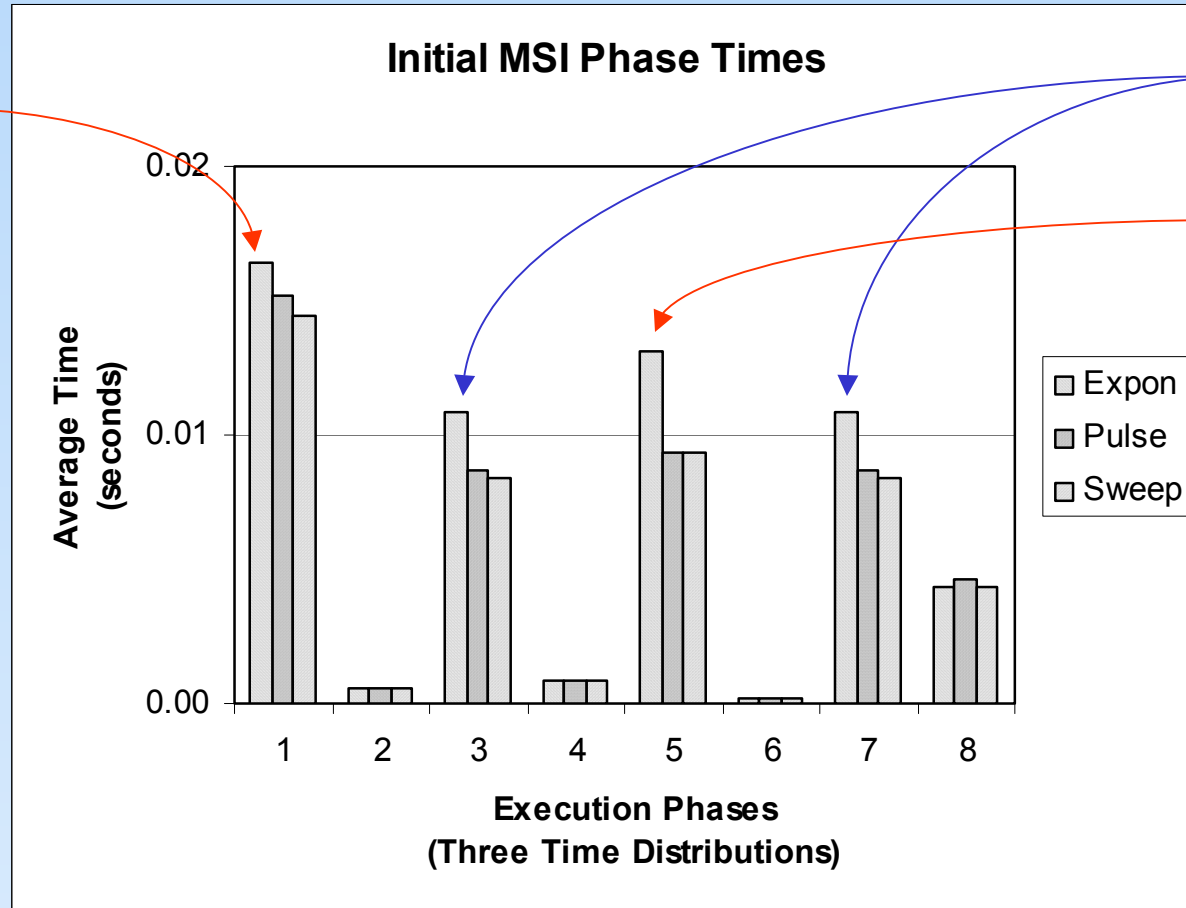
Performance Tuning - Defining Execution Phases:



Linux Cluster Architecture

Performance Tuning - Execution Phase Times:

Found a bug!

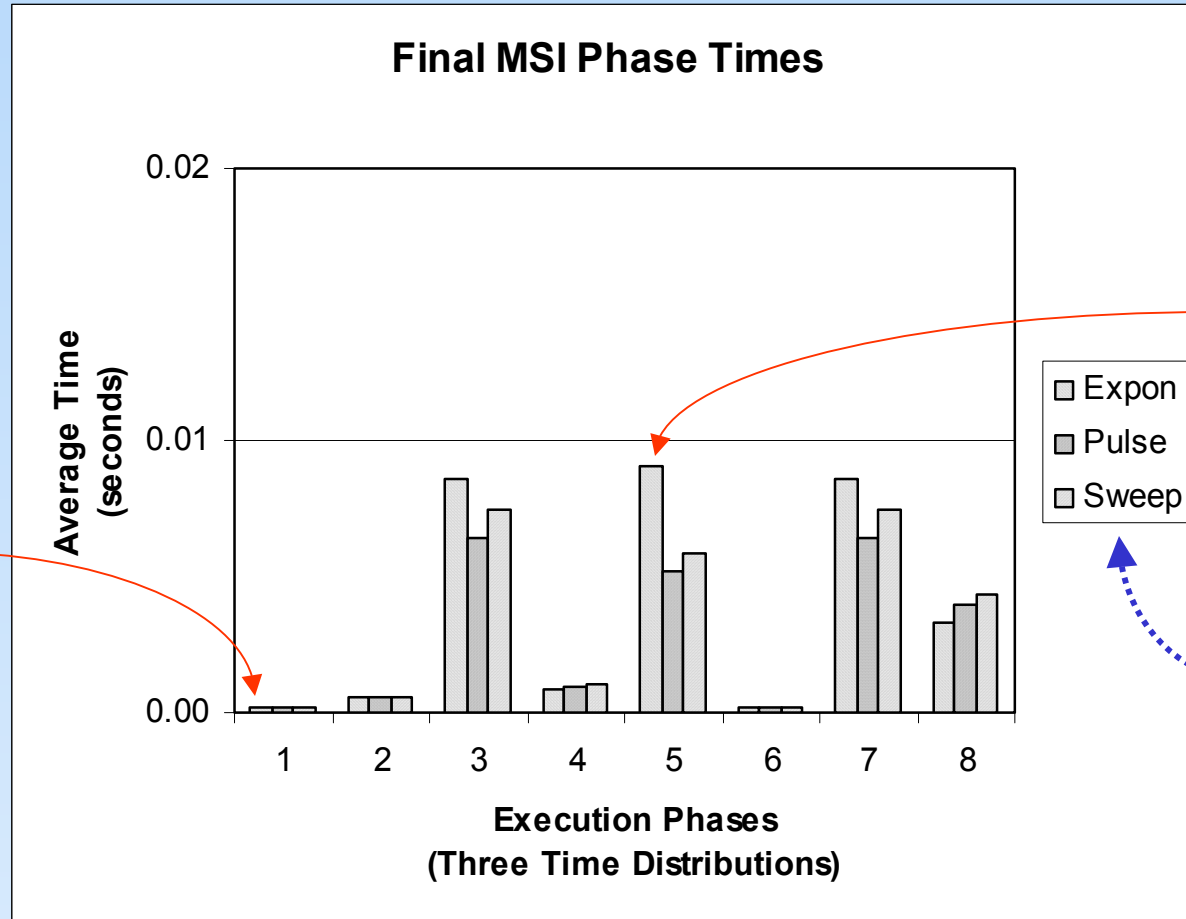


Not a SW issue.

Leave the file open?

Linux Cluster Architecture

Performance Tuning - Final Times:



Dramatic reduction!

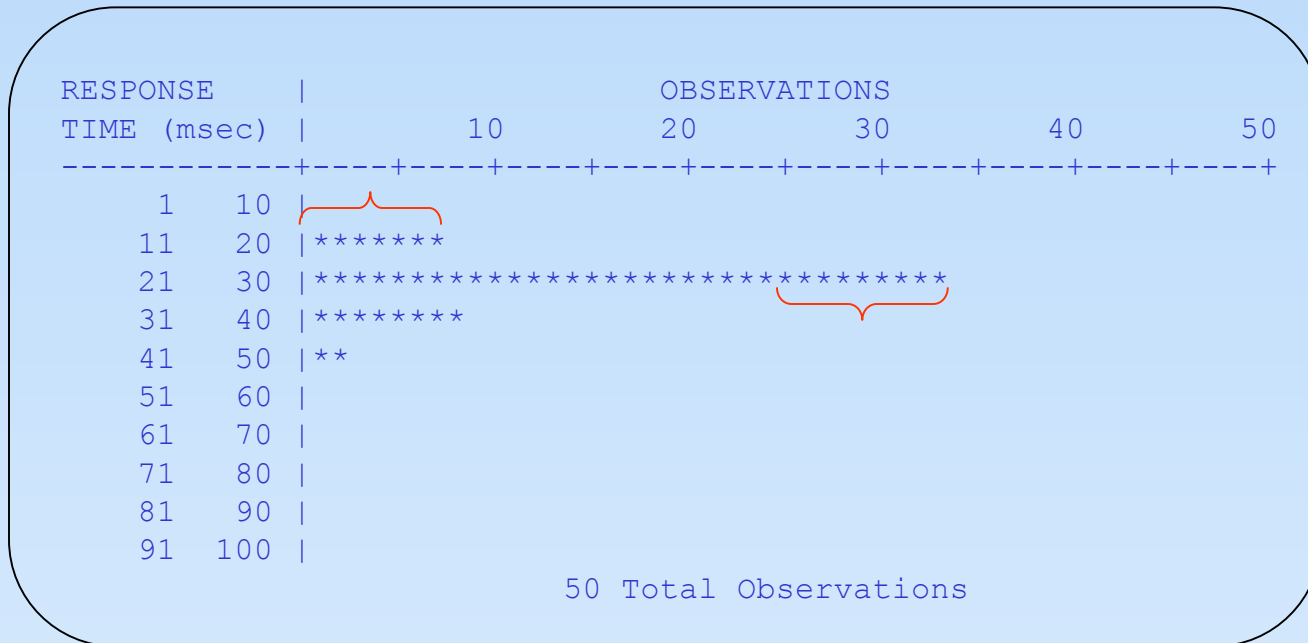
About a 10% improvement (not too bad)

See book for further details on statistical distributions.

Linux Cluster Architecture

Performance Tuning:

- The proof is in the pudding!



Average = 25 milliseconds = 17% improvement!

Linux Cluster Architecture

Further Details are in the Book:

- Download all the C source code for free:

<http://www.sampublishing.com>

- Search on "Linux Cluster Architecture" or "Vrenios"
- Click on the "Downloads" link in the book description
 1. Individual chapter examples are in zip files
 2. A complete user-files environment is in a tar.gz file

- Book Signings:

Sep 8th

Borders Chandler, Sunday @ 2pm

Sep 15th

Borders Arrowhead, Sunday @ 2pm

Oct 25th

Barnes & Noble Arrowhead, Friday @ 7pm

Linux Cluster Architecture

Further Reading:

Distributed Operating Systems, Andrew S. Tanenbaum
(of **MINIX** and **AMOEB**A fame!), Prentice Hall, 1995

Unix Distributed Programming, Chris Brown, Prentice Hall, 1994

Advanced Programming in the UNIX Environment,
W. Richard Stevens, Addison-Wesley, 1992

Linux Programming White Papers, Rushling, et al, CoriolisOpen, 1999

-> Further details about the early development of my cluster are in:

"CHAOS: A Cheap Array of Outmoded Systems," Alex Vrenios,
LinuxGazette.com, October 1998

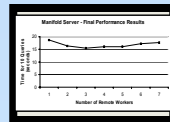
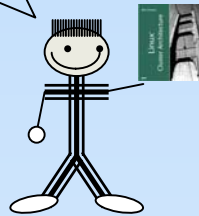
"CHAOS Part 2," LinuxGazette.com, Alex Vrenios, December 1998

Linux Cluster Architecture

You've been a terrific audience!

Any questions?

Hurry out and buy this book!



Or this one...



Slide # 32

Copyright © 2003 Alexander Vrenios