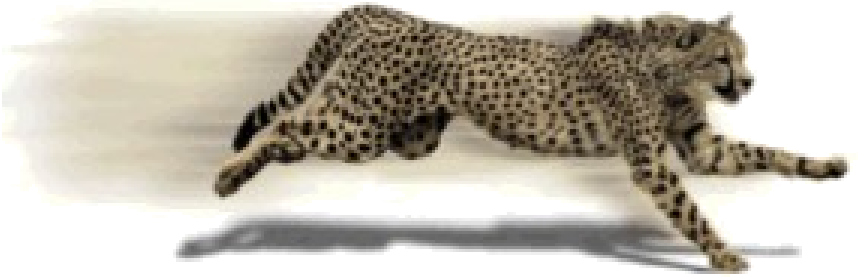


# Penguins on a Pin Head

## Experiences with tuning Linux on a P/390

Rob van der Heij
Velocity Software, Inc

*rvdheij@velocitysoftware.com*

http://velocitysoftware.com/

# Agenda

- The Size of a Pinhead
- Linux on a P/390
- Performance Monitor
- Get your expectations right
- Reducing the Idle Load
- Sneak Preview of unionfs
- Experiments
- Conclusion

# What is the Pin Head

NEW ORLEANS, May 22, 1995 . . . IBM today announced at the IBM Technical Interchange conference the first IBM PC server that can run both PC and mainframe-based applications. The new server, called the IBM PC Server 500 S/390, will enable application developers to write and test mainframe applications right on their PC, either in a standalone or LAN-based environment. That will save customers money by allowing developers to create new mainframe applications without sharing time on the mainframe. It can also reduce costs in such areas as network management and systems training.

A complete System/390 processor on a single PCI card.

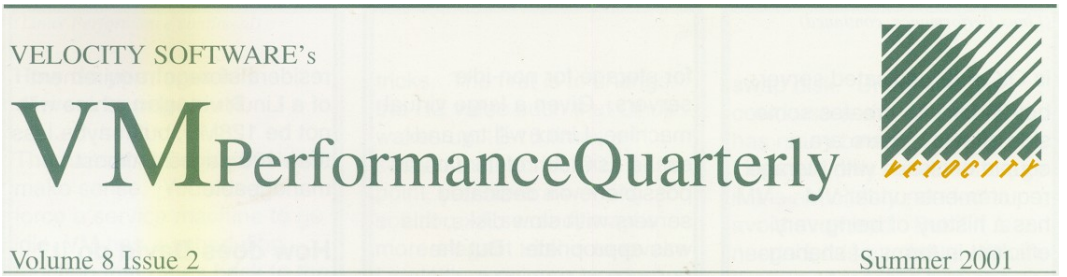
# Size of a Pin Head

*Two computing environments in one cost-effective solution*

- PC Server 330
  - Pentium Pro 200 MHz
  - 64 MB Memory
  - OS/2 with P/390 drivers for I/O
- S/390 CPU
  - 3 – 4 MIPS (no IEEE)
  - 128 MB ECC Memory
  - 256 MB Expanded Memory

Two orders smaller than a typical z9 system

# Running Linux on a P/390




Why would you?

- P/390 is easier to get than a z9
- Technical Challenge
- Demonstrate techniques
- Train your tuning skills
- Easy to spot your mistakes (they get amplified)

**How many idle users can we support now?**

I have a bet with Rob Van der Heij that we can run 100 Linux servers on a 128MB P390. Results of this bet to be posted...

# Running Linux on a P/390

## Working with one hand on your back

- Old VM release - z/VM 3.1 (31-bit)
  - No Guest LAN, no QDIO, no LCS
  - No VM63282
- Limited to Linux 2.4 kernel
  - SLES8 SP4 with some tweaking
  - No 2.6 kernel memory management fixes

## But at least a good Performance Monitor !

# ESALPS - Linux Performance Suite

| | |
|---|---|
| ESAMON | Real Time Monitor |
| ESAMAP | Performance Reporting |
| ESAWEB | Fast Web Server |
| ESATCP | Linux Performance Monitor |
| | |
| zMON | Entry Level Real Time Monitor |
| zTUNE | Performance Services Subscription |

# Reboot of a Linux Server

```
Screen: ESAUSR2  rvdheij.com                          ESAMON V3.5  01/25 09:31-09:56
1 of 3  User Resource Utilization                     CLASS * USER LNX00A0 P390 00000

                  <---CPU time--> <--------Main Storage (pages)--------->
         UserID   <(seconds)> T:V <Resident>  Lock <-----WSS----->
Time     /Class   Total  Virt Rat Total Activ  -ed Total Actv  Avg Resrvd
-------- -------- ----- ----- --- ----- ----- ---- ----- ---- ---- ------
09:56:00 LNX00A00  0.22  0.20 1.1 10797 10797    0 12835  12K  12K      0
09:55:00 LNX00A00  1.67  1.59 1.1 10797 10797    0 12835  12K  12K      0
09:54:00 LNX00A00 56.51 55.42 1.0 10797 10797    0 12835  12K  12K      0
09:53:00 LNX00A00 57.91 57.40 1.0 10703 10703    0 12835  12K  12K      0
09:52:00 LNX00A00 49.88 48.97 1.0 10703 10703    0 12835  12K  12K      0
09:51:00 LNX00A00 35.91 35.28 1.0 10703 10703    0 12835  12K  12K      0
09:50:00 LNX00A00 27.98 27.46 1.0 10703 10703    0 12835  12K  12K      0
09:49:00 LNX00A00 50.26 48.64 1.0 10703 10703    1 10642  10K  10K      0
09:48:00 LNX00A00 48.23 45.99 1.0 10702 10702    1 10642  10K  10K      0
09:47:00 LNX00A00 21.10 14.66 1.4  3353  3353    0  3281 3281 3281      0
09:46:00 LNX00A00  0.19  0.17 1.1 22640 22640    0 22640  22K  22K      0
09:45:00 LNX00A00  0.27  0.20 1.3 22640 22640    0 22619  22K  22K      0
09:44:00 LNX00A00  1.83  1.60 1.1 22640 22640    0 22618  22K  22K      0
```

**Observations**

- **Reboot took about 8 minutes, and used 300 seconds CPU time**
- **Resident pages after reboot half of what it was before (~88 MB)**
  **Note: Whatever the change, reboot it may appear to fix things**

# Reboot of a Linux Server

```
Screen: ESASTR2  rvdheij.com                          ESAMON V3.5
1 of 2  Main Storage DPA Analysis

                          <----DPA--> <Avail List> Failed
         <---Users---->     Size Stor Size  Empty  Demand
Time       On Actv In Q    pages Load (pgs)  /Sec   Scans
-------- ---- ---- ---- ------ ---- ----- ------ ------
09:56:00   15    5  2.0  31459 0.54  7848   0.00      0
09:55:00   15    5  2.0  31461 0.54  7860   0.00      0
09:54:00   15    5  1.0  31458 0.54  7889   0.00      0
09:53:00   15    5  2.0  31460 0.54  8933   0.00      0
09:52:00   15    5  2.0  31453 0.54  9372   0.00      0
09:51:00   15    5  2.0  31460 0.54 10953   0.00      0
09:50:00   15    5  2.0  31458 0.53 11229   0.00      0
09:49:00   15    5  2.0  31453 0.47 11429   0.00      0
09:48:00   15    5  2.0  31453 0.46  9095   0.00      0
09:47:00   15    5  2.0  31451 0.23 19774   0.00      0
09:46:00   15    6  1.0  31454 0.85  1534   0.00      0
09:45:00   15    7  1.0  31454 0.85  1567   0.00      0
09:44:00   15    5  1.0  31456 0.85  1084   0.00      0
```

**Observations**

- **Maybe z/VM was not constrained and had no need to pages the server out?**
- **But this is a 128M Linux server... doesn't Linux use all memory you give it?**

# Reboot of a Linux Server

```
Screen: ESAUSR2  rvdheij.com                          ESAMON V3.5  01/25 09:31-09:55
3 of 3  User Resource Utilization                     CLASS * USER LNX00A0 P390 00000

                  <------Paging (pages)-------> <Spooling(pages)> Q'd
         UserID   <---Allocated---> <---I/O--->       <---I/O---> Pg+
Time     /Class   Total ExStg  Disk  Read Write Alloc  Read Write Spl
-------- -------- ----- ----- ----- ----- ----- ----- ----- ----- ---
09:56:00 LNX00A00     0     0     0     0     0    58     0     0   0
09:55:00 LNX00A00     0     0     0     0     0    58     0     0   0
09:54:00 LNX00A00     0     0     0     0     0    58     0     0   0
09:53:00 LNX00A00     0     0     0     0     0    58     0     0   0
09:52:00 LNX00A00     0     0     0     0     0    58     0     0   0
09:51:00 LNX00A00     0     0     0     0     0    58     0     0   0
09:50:00 LNX00A00     0     0     0     0     0    58     0     0   0
09:49:00 LNX00A00     0     0     0     0     0    58     0     0   0
09:48:00 LNX00A00     0     0     0     0     0    58     0     0   0
09:47:00 LNX00A00     1     0     1     1     1    58     0     0   0
09:46:00 LNX00A00 17255     0 17255     0     0    58     0     0   0
09:45:00 LNX00A00 17255     0 17255     0     0    58     0     0   0
09:44:00 LNX00A00 17255     0 17255    56     0    58     0     0   0
```

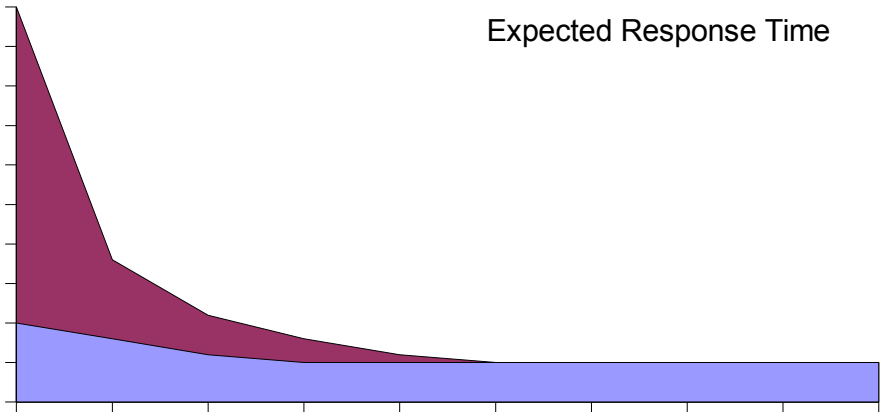
**Observation**

- **Large portion of the Linux server had been paged out**
  **Numbers don't add up because same page can be in memory and on disk**

# Get your Expectations Right

- Transaction must be repeatable
- Response time can be measured
- Shows characteristics we want to tune
  - Transaction latency (start-up delay)
  - System overhead

  - Apache web server
  - httperf measurement



# Get your Expectations Right

## Measurement of 250 web hits

- LINUX00B00 runs apache
  - 12.5 seconds for 250 hits = 50 ms per web hit
- LINUX00A00 runs httperf
  - 127 seconds = 500 ms per web hit (just to measure it?)

| Time | UserID /Class | <---CPU time--> <(seconds)> Total | Virt | T:V Rat |
|---|---|---|---|---|
| 11:29:00 | LNX00A00 | 0.55 | 0.51 | 1.1 |
| 11:28:00 | LNX00A00 | 0.75 | 0.69 | 1.1 |
| 11:27:00 | LNX00A00 | 5.23 | 4.52 | 1.2 |
| 11:26:00 | LNX00A00 | 43.93 | 26.21 | 1.7 |
| 11:25:00 | LNX00A00 | 37.92 | 22.36 | 1.7 |
| 11:24:00 | LNX00A00 | 40.48 | 23.96 | 1.7 |
| 11:23:00 | LNX00A00 | 2.67 | 1.91 | 1.4 |
| 11:22:00 | LNX00A00 | 0.55 | 0.51 | 1.1 |

| Time | UserID /Class | <---CPU time--> <(seconds)> Total | Virt | T:V Rat |
|---|---|---|---|---|
| 11:29:00 | LNX00B00 | 0.24 | 0.22 | 1.1 |
| 11:28:00 | LNX00B00 | 0.25 | 0.22 | 1.1 |
| 11:27:00 | LNX00B00 | 0.34 | 0.30 | 1.2 |
| 11:26:00 | LNX00B00 | 4.42 | 3.74 | 1.2 |
| 11:25:00 | LNX00B00 | 4.32 | 3.66 | 1.2 |
| 11:24:00 | LNX00B00 | 4.14 | 3.50 | 1.2 |
| 11:23:00 | LNX00B00 | 0.52 | 0.45 | 1.2 |
| 11:22:00 | LNX00B00 | 0.25 | 0.22 | 1.1 |

# Get your Expectations Right

Using a “remote” client to send the requests

| | |
|---|---|
| Apache | 43 mS |
| TCP/IP | 50 mS |

- Whatever configuration of virtual machines, we should not expect more than 10 hits/sec.
- Issues with a lot of virtual machines on z/VM
  - Idle load of all other virtual machines
  - z/VM overhead due to paging
  - Linux overhead in a constrained environment

# Reducing Idle Load

```
Screen: ESAUSR2  rvdheij.com                          ESAMON V3.5  03/06 01:55-
1 of 3  User Resource Utilization                     CLASS * USER LNX00C0 P390

                  <---CPU time--> <--------Main Storage (pages)--------->
         UserID   <(seconds)> T:V <Resident>  Lock <-----WSS----->
Time     /Class   Total  Virt Rat Total Activ  -ed Total Actv  Avg Resrvd
-------- -------- ----- ----- --- ----- ----- ---- ----- ---- ---- ------
02:13:00 LNX00C01  3.21  1.86 1.7  6990  6990    0  6974 6974 6974      0  hz_timer
02:12:00 LNX00C01  3.24  1.88 1.7  6990  6990    0  6974 6974 6974      0
02:11:00 LNX00C01  1.17  1.01 1.2  6990  6990    0  6966 6966 6966      0
02:10:00 LNX00C01  0.40  0.36 1.1  6980  6980    0  6964 6964 6964      0
02:09:00 LNX00C01  0.35  0.32 1.1  6980  6980    0  6964 6964 6964      0
02:08:00 LNX00C01  0.35  0.32 1.1  6980  6980    0  6964 6964 6964      0
02:07:00 LNX00C01  0.35  0.33 1.1  6980  6980    0  6964 6964 6964      0
02:06:00 LNX00C01  0.35  0.32 1.1  6980  6980    0  6964 6964 6964      0
02:05:00 LNX00C01  0.35  0.32 1.1  6980  6980    0  6964 6964 6964      0
```

- An idle usage of 350 mS/min (0.6%) remains in this case

# Reducing Idle Load

## Counting timer ticks with TRACE EXT 1004

- Idle server with about 120 ticks per minute
- ESAMON confirms that Linux drops from queue

```
Screen: ESAUSRQ  rvdheij.com                          ESAMON V3.5
1 of 3  User Queue and Load Analysis                  CLASS * USER

                  <----------User Load------------->
         UserID   Logged  Non-           Disc- Total  Tran
Time     /Class       on  Idle  Active   conn  InQue  /min
-------- -------- ------ ----- ------- ------ -----  ----
12:48:00 LNX00C01      1     1       1      0  0.18   104
12:47:00 LNX00C01      1     1       1      0  0.35   112
12:46:00 LNX00C01      1     1       1      0  0.60    72
12:45:00 LNX00C01      1     1       1      0  0.25    86
12:44:00 LNX00C01      1     1       1      0  0.05   120
12:43:00 LNX00C01      1     1       1      0  0.07   120
```

# Reducing Idle Load

## Locate `schedule_timeout` in System.map

- `001234bc T schedule_timeout`
- Find a suitable place to put the trace

```
#cp trace i r 12350e.2 term run cmd d g2"#d 80.4;base1
```

```
276 ?        S      0:03 /usr/sbin/cron
287 ?        S      1:59 /usr/sbin/sshd
303 ?        S      0:09 /usr/sbin/httpd -f /etc/httpd/httpd.conf
311 ?        S      0:01 /usr/sbin/nscd
313 ?        S      0:01 /usr/sbin/nscd
314 ?        S      0:01 /usr/sbin/nscd
315 ?        S      0:00 /usr/sbin/nscd
316 ?        S      0:00 /usr/sbin/nscd
317 ?        S      0:00 /usr/sbin/nscd
318 ?        S      0:00 /usr/sbin/nscd
320 ?        S      0:00 /usr/sbin/httpd -f /etc/httpd/httpd.conf
```

| Count | Time | PID |
|---|---|---|
| 363 | 100 | 303 |
| 180 | 201 | 313 |
| 72 | 500 | 1 |
| 73 | 500 | 8 |
| 24 | 1501 | 311 |
| 24 | 1501 | 314 |
| 1 | 5201 | 276 |
| 2 | 5901 | 276 |
| 4 | 6001 | 276 |

# Reducing Idle Load

| Time | Node | Process/ Application name | ID | <---Processor Percent---> Total | <Process> sys | <Children> user | <Children> syst | <Children> usrt | <---CPU Seconds---> <Process> sys | <Process> user | <Children> syst | <Children> usrt | Proc Count | <-----Percent Process Status-- Actv | Run- ing | Sleep -ing | Zom bie | Disk Wait | Page Wait |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 09:03:00 | NEALE1 | *Totals* | 0 | 3.7 | 1.4 | 2.3 | 0.0 | 0.0 | 0.8 | 1.4 | 0.0 | 0.0 | 36.0 | 2.0 | 2.0 | 0.0 | 34 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | init | 1 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | syslogd | 962 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | klogd | 965 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | portmap | 974 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | sshd | 22218 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | xinetd | 1067 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | cron | 1073 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | nscd | 1076 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | mingetty | 1156 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 2.0 | 0.0 | 0.0 | 0.0 | 2.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | snmpd | 4213 | 0.3 | 0.2 | 0.1 | 0.0 | 0.0 | 0.1 | 0.1 | 0.0 | 0.0 | 1.0 | 1.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | events/0 | 4 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 10.0 | 0.0 | 0.0 | 0.0 | 10 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | httpd2-p | 6097 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 7.0 | 0.0 | 0.0 | 0.0 | 7.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | mono | 6106 | 3.4 | 1.2 | 2.2 | 0.0 | 0.0 | 0.7 | 1.4 | 0.0 | 0.0 | 1.0 | 1.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | migratio | 2 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | ksoftirq | 3 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | kswapd0 | 100 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | kmcheck | 158 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | rpciod | 989 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | lockd | 990 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |
| 09:03:00 | NEALE1 | login | 1155 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.0 | 0.0 |

# Reducing the Idle Load

- Determine the cost of services by stopping them and measure the usage again

```
 80 mS  nscd
 10 mS  cron
 70 mS  apache
 10 mS  syslog
180 mS various
```

# Reducing Idle Load

```
Screen: ESAMAIN  rvdheij.com                        ESAMON V3.5  03/02 10:48-11:20
1 of 3  System Overview                             LIMIT 500            P390 00000

         <---Users----> Transact.      <Processor>  Cap- <--Storage (MB)->
         <-avg number->  per Avg.      Utilization  ture Fixed Active Stor
Time       On Actv In Q Sec. Time CPUs Total Virt. Ratio  User Resid. Load
*------- ---- ---- ---- ---- ---- ---- *---- ----- ----- ----- ------ ----
11:20:00   30   20  3.0 10.5 0.05    1  15.1  11.5   100     2    111  0.3
11:19:00   30   20  2.0 10.3 0.04    1  16.6  13.0   100     2    111  0.3
11:18:00   30   20  3.0 10.6 0.05    1  15.8  11.8   100     2    111  0.3
11:17:00   30   19  5.0  9.3 0.05    1  27.6  20.3   100     2    111  0.3
11:16:00   30   19 15.0  0.3 0.14    1 100.0  86.4   100     2    110  0.3
11:15:00   30   19 14.0 10.5 0.05    1  15.1  11.2   100     3    109  0.3
11:14:00   30   19  8.0 10.5 0.05    1  14.6  11.1   100     2    109  0.3
11:13:00   30   19  5.0 10.5 0.05    1  16.1  12.2   100     2    109  0.3
11:12:00   30   19  7.0 10.6 0.05    1  14.9  11.3   100     2    109  0.3
11:11:00   30   21  5.0 10.6 0.04    1  20.2  13.3   100     2    109  0.3
11:10:00   30   19  5.0 10.3 0.05    1  14.2  10.6    99     2    113  0.3
11:09:00   30   18  4.0 10.5 0.05    1  12.5   9.6   100     2    112  0.3
11:08:00   30   18  5.0 10.6 0.05    1  13.1  10.0   100     2    112  0.3
11:07:00   30   18  5.0 10.5 0.04    1  12.6   9.6   100     2    112  0.3
11:06:00   30   18  3.0 10.6 0.05    1  13.4  10.1   100     2    112  0.3
11:05:00   30   18  3.0 11.0 0.04    1  12.7   9.7   100     2    111  0.3
11:04:00   30   18  5.0 10.5 0.05    1  13.2  10.2   100     2    111  0.3
11:03:00   30   18  6.0 10.3 0.05    1  12.6   9.4   100     2    111  0.3
11:02:00   30   18  6.0  9.9 0.05    1  24.1  18.5   100     2    111  0.3
11:01:00   30   20 14.0  0.5 4.06    1 100.0  88.5   100     2    111  0.3
11:00:00   30   19 12.0  8.7 0.30    1  30.4  22.7   100     3    113  0.3
10:59:00   30   18 16.0 10.5 0.05    1  13.6   9.8   100     2    113  0.3
```

# Sneak Preview of UnionFS

## Stackable file system

http://www.fsl.cs.sunysb.edu/project-unionfs.html

- Appears to merge the contents of directories while retaining the physical location
  - Provide R/W layer over a R/O device
  - COW disks implement this at the block level
  - unionfs operates on file-level

```
mount -t unionfs none /data -o dirs=/private/data=rw:/shared/data=ro
```

write /private/data
read /private/data, otherwise /shared/data

# Sneak Preview of UnionFS

- Modified Linux startup to merge 3 layers
  - R/W Private data
  - R/O Server specific configuration files
  - R/O Full root file system
- Allows for easy site-wide configuration changes
- No free lunch
  - Tricky to change R/O disks afterwards
  - One-time performance hit when copying file
  - Some functional problems

# Experiment #1

- With 40 servers using shared kernel in NSS
  - Idle servers plus infrastructure use 45% of CPU
  - Almost no paging

| | <---Users----> | | | Transact. | | | <Processor> | | Cap- | <--Storage (MB)-> | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | <-avg number-> | | | per Avg. | | | Utilization | | ture | Fixed | Active | Stor |
| Time | On | Actv | In Q | Sec. | Time | CPUs | Total | Virt. | Ratio | User | Resid. | Load |
| 11:11:00 | 56 | 47 | 24.0 | 46.9 | 0.11 | 1 | 44.6 | 34.9 | 99 | 3 | 99 | 0.3 |
| 11:10:00 | 56 | 47 | 23.0 | 47.7 | 0.09 | 1 | 44.0 | 34.4 | 99 | 3 | 99 | 0.3 |
| 11:09:00 | 56 | 47 | 29.0 | 48.3 | 0.10 | 1 | 45.9 | 35.6 | 99 | 3 | 98 | 0.2 |
| 11:08:00 | 56 | 46 | 25.0 | 48.5 | 0.10 | 1 | 43.5 | 34.1 | 100 | 3 | 97 | 0.2 |
| 11:07:00 | 56 | 46 | 33.0 | 47.2 | 0.11 | 1 | 44.6 | 34.8 | 100 | 3 | 97 | 0.2 |
| 11:06:00 | 56 | 45 | 24.0 | 47.6 | 0.10 | 1 | 41.3 | 32.5 | 100 | 3 | 96 | 0.2 |
| 11:05:00 | 56 | 46 | 23.0 | 47.5 | 0.11 | 1 | 44.2 | 34.7 | 99 | 3 | 96 | 0.2 |
| 11:04:00 | 56 | 46 | 25.0 | 48.3 | 0.10 | 1 | 43.5 | 34.0 | 99 | 3 | 96 | 0.2 |
| 11:03:00 | 56 | 46 | 24.0 | 47.8 | 0.10 | 1 | 44.0 | 34.6 | 100 | 3 | 96 | 0.2 |
| 11:02:00 | 56 | 46 | 23.0 | 48.8 | 0.09 | 1 | 43.8 | 34.2 | 100 | 3 | 96 | 0.2 |
| 11:01:00 | 56 | 46 | 27.0 | 49.3 | 0.10 | 1 | 50.7 | 40.5 | 100 | 3 | 95 | 0.2 |

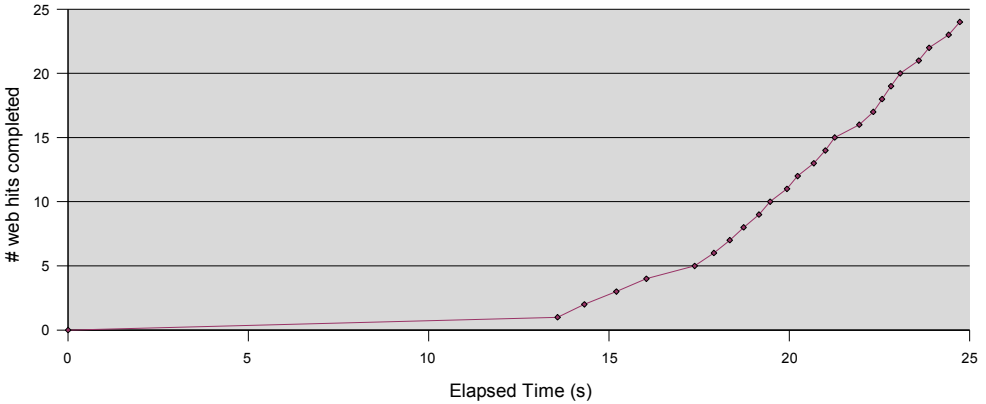
# Experiment #1

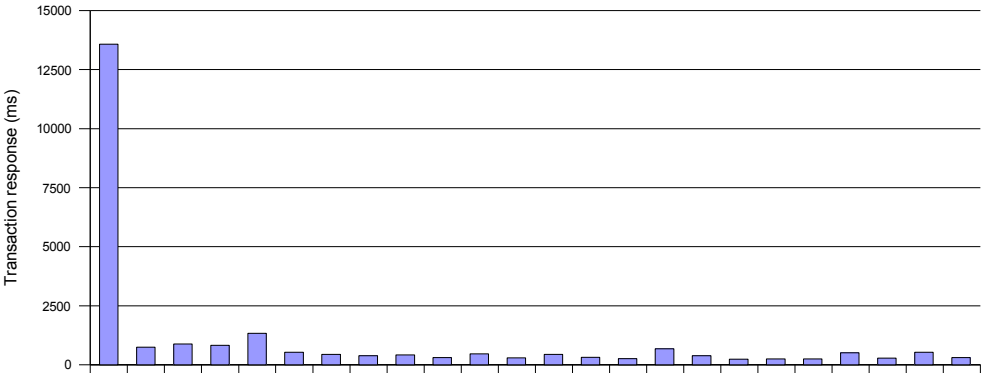
- Grouped in classes of 10 servers each
  - Idle server at 350 mS/min
  - Average working set 540 pages (2.1 MB)

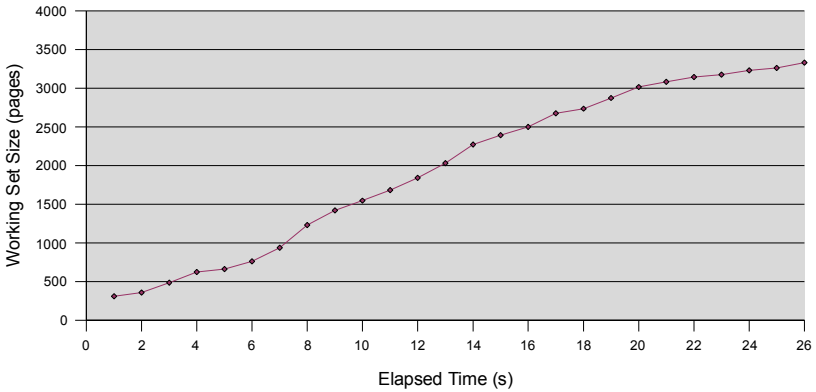
| | | <---CPU time--> | | | <--------Main Storage (pages)---------> | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | UserID | <(seconds)> | | T:V | <Resident> | | Lock | <-----WSS-----> | | | |
| Time | /Class | Total | Virt | Rat | Total | Activ | -ed | Total | Actv | Avg | Resrvd |
| 11:11:00 | *Keys | 1.51 | 1.13 | 1.3 | 487 | 486 | 20 | 518 | 467 | 156 | 0 |
| | *Servers | 6.51 | 6.13 | 1.1 | 1948 | 1789 | 6 | 4497 | 1764 | 176 | 0 |
| | *TheUsrs | 0.66 | 0.65 | 1.0 | 475 | 474 | 1 | 577 | 474 | 158 | 0 |
| | LNX | 14.20 | 13.04 | 1.1 | 22571 | 22571 | 0 | 22694 | 21K | 540 | 0 |
| | LNX0 | 3.22 | 2.96 | 1.1 | 4167 | 4167 | 0 | 4984 | 3896 | 390 | 0 |
| | LNX1 | 3.71 | 3.40 | 1.1 | 5195 | 5195 | 0 | 5031 | 5031 | 503 | 0 |
| | LNX2 | 3.64 | 3.35 | 1.1 | 6766 | 6766 | 0 | 6496 | 6496 | 650 | 0 |
| | LNX3 | 3.63 | 3.33 | 1.1 | 6443 | 6443 | 0 | 6183 | 6183 | 618 | 0 |
| | System: | 22.88 | 20.95 | 1.1 | 25481 | 25320 | 27 | 28286 | 24K | 434 | 0 |

# Experiment #1

- Measurements show average latency in 1st transaction of 12.5 s
- WSS grows from 500 to 3000 pages



# Experiment #1

- Most of the latency is due to paging
  - 2500 pages in 15 seconds = 170 pg/s
  - More than what CP keeps in back-pocket
  - Peak at 200 pg/s to DASD

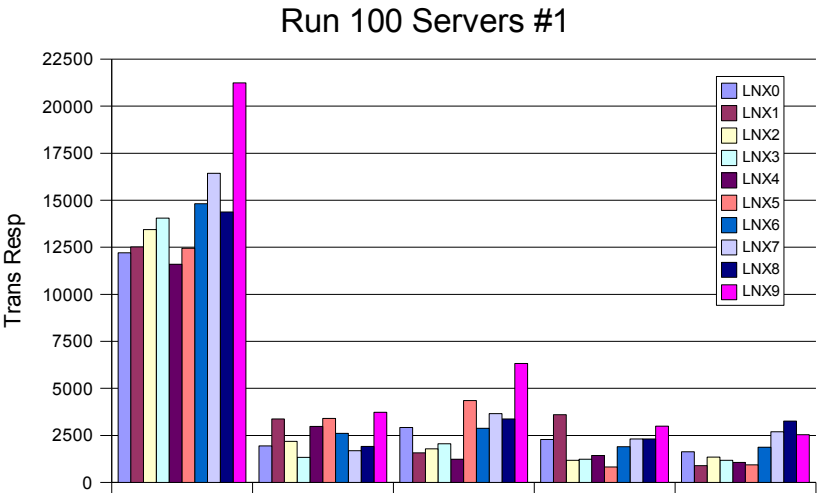
| | <-------Expanded Storage---------------> | | | | | | <-----Paging-----> | | |
|---|---|---|---|---|---|---|---|---|---|
| | (MB) | <------pages/second------> | | | | Page | <-pages/sec-> | | Serv |
| Time | Avail | Alloc | Relse | PGIN | PGOUT | Age | Read | Write | Time |
| 15:50:00 | 256 | 40 | 57 | 37 | 23 | 823.0 | 3.5 | 0.1 | 4.3 |
| 15:49:00 | 256 | 48 | 75 | 59 | 38 | 561.0 | 9.9 | 0.1 | 4.9 |
| 15:48:00 | 256 | 169 | 177 | 142 | 158 | 421.0 | 84.0 | 15.5 | 5.9 |
| 15:47:00 | 256 | 189 | 190 | 114 | 188 | 580.0 | 103.0 | 89.0 | 13.2 |
| 15:46:00 | 256 | 144 | 113 | 74 | 134 | 864.0 | 100.1 | 29.1 | 10.1 |
| 15:45:00 | 256 | 90 | 83 | 82 | 88 | 2761 | 33.3 | 0.1 | 3.3 |
| 15:44:00 | 256 | 14 | 14 | 14 | 14 | 7625 | 5.7 | 0.0 | 1.8 |

# Linux Footprint Reduction

- Kernel in NSS
  - Savings are relatively small (2 MB)
  - Kernel pages are good to have around
- Linux binaries in DCSS using xip file system
  - Especially useful for “memory mapped” files
    - Programs and shared libraries
  - Space is limited, so need to select the files
  - Mount into root file system with the –bind option

# Experiment #2: Using xip

- Test with 100 servers
  - 30M virtual memory
  - 24M VDISK for swap
  - 256M DCSS
- Avg latency 12 sec
- Latency again can be explained by paging
  - WSS grows from 150 to 1000 pages
  - 1300 -2300 page reads per server

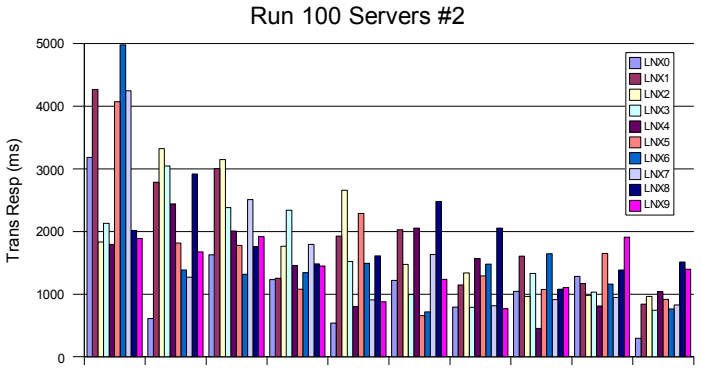
# Experiment #2: Using xip

- Latency of 12 - 20 seconds is rather heavy
- Problem is caused by “new” virtual servers
  - Data has been cached during boot process
  - Running the real application makes Linux re-use pages that still had initial data

# Experiment #2: Using xip

The 2nd run performs much better

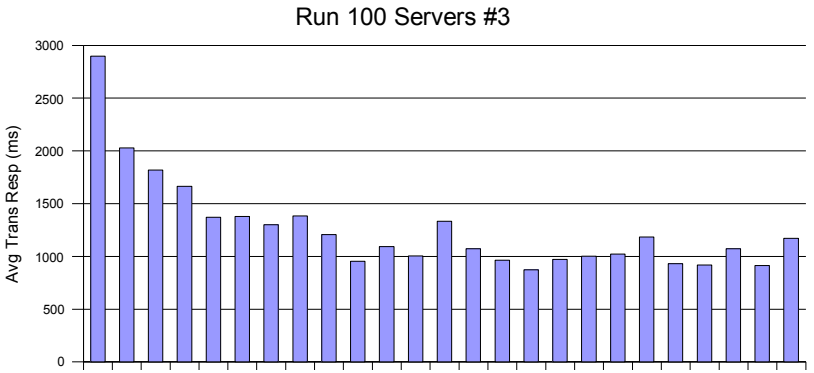
- Virtual machines have settled better
- Overall average latency under 2 seconds
  - Difference between groups has disappeared
  - Still a worst case scenario (oldest server next)




# Experiment #2: Using xip

- Subsequent runs produce very similar pattern
  - Virtual machines have settled
  - WSS of 250 pages with very minimal paging

```
          <---CPU time--> <---Main Storage (pages)---
UserID    <(seconds)> T:V <Resident>  <-----WSS----->
/Class    Total  Virt Rat Total Activ Total Actv  Avg
-------- ----- ----- --- ----- ----- ----- ---- ----
LNX      19.74 17.80 1.1 25494 25494 24253  24K  243
LNX0      1.97  1.78 1.1  2278  2278  2071 2071  207
LNX1      1.98  1.78 1.1  2221  2221  2039 2039  204
LNX2      1.95  1.76 1.1  2303  2303  2225 2225  223
LNX3      1.97  1.77 1.1  2309  2309  2191 2191  219
LNX4      2.01  1.81 1.1  3435  3435  3304 3304  330
LNX5      1.94  1.75 1.1  2863  2863  2797 2797  280
LNX6      1.98  1.79 1.1  2624  2624  2522 2522  252
LNX7      2.02  1.82 1.1  2409  2409  2335 2335  234
LNX8      1.96  1.77 1.1  2062  2062  1941 1941  194
LNX9      1.98  1.79 1.1  2990  2990  2828 2828  283
```



# Experiment #2: Using xip

- DCSS is large so it can hold a lot of code
  - Segment of 256M at 256M boundary
    - Mapping the segment in Linux takes 6M
  - Only 332 pages are being used (~ 0.5%)
- Kernel in NSS
  - 547 pages R/O of which 224 resident

```
Filename=SUS8SEG1  Filetype=DCSS  Class=A  Spoolid=0046
Time loaded=1 01:08   Size=256M
Pages:  Main=325  Xstore=6  Dasd=7307  Locked=0
Paging:
  Xstore:  Reads=66615      Writes=73763      Migrates=7142
  Dasd:    Reads= 8567      Writes= 7904
```

# Reducing Linux Footprint

## Options for reducing memory requirements

### Smaller virtual machines!

- During tests there was no swapping at all
- All used VDISK pages are paged out
- Make them small to cause some swapping

| Time | Owner | Space Name | <AddSpce> Cre- ates | Del- etes | DASD Page Slots | X-Store Blks |
|---|---|---|---|---|---|---|
| 01:19:00 | LNX00C00 | VDISK$LNX00C00$0200$00C0 | 0 | 0 | 56 | 0 |
| | LNX00C01 | VDISK$LNX00C01$0200$00C1 | 0 | 0 | 56 | 0 |
| | LNX00C02 | VDISK$LNX00C02$0200$00C2 | 0 | 0 | 56 | 0 |
| | LNX00C03 | VDISK$LNX00C03$0200$00C3 | 0 | 0 | 56 | 0 |
| | LNX00C04 | VDISK$LNX00C04$0200$00C4 | 0 | 0 | 56 | 0 |
| | LNX00C05 | VDISK$LNX00C05$0200$00C5 | 0 | 0 | 56 | 0 |
| | LNX00C06 | VDISK$LNX00C06$0200$00C6 | 0 | 0 | 56 | 0 |
| | LNX00C07 | VDISK$LNX00C07$0200$00C7 | 0 | 0 | 56 | 0 |

# Reducing Linux Footprint

- Squeeze with CMM rather than reboot
  - Avoids the “reboot helps” effect
  - No need to waste time on IPL
- Could chop off 10 MB before Linux starts to swap

| Time | UserID /Class | DASD I/O | DASD Block I/O | MDisk Cache Hits | Virt Disk I/O | Cache Hit Pct | <---Users----> <-avg number-> On | Actv | In Q | Transact. per Sec. | Avg. Time | CPUs | <Processor> Utilization Total | Virt. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 16:16:00 | LNX | 10 | 0 | 0 | 0 | 0.0 | 116 | 106 | 88.0 | 55.1 | 0.80 | 1 | 69.0 | 54.0 |
| 16:15:00 | LNX | 31 | 0 | 1 | 34 | 3.2 | 116 | 106 | 83.0 | 49.7 | 0.99 | 1 | 84.4 | 54.6 |
| 16:14:01 | LNX | 34 | 0 | 3 | 27 | 8.8 | 116 | 106 | 96.0 | 51.9 | 0.88 | 1 | 97.6 | 65.4 |
| 16:13:00 | LNX | 47 | 0 | 1 | 79 | 2.1 | 116 | 106 | 71.0 | 51.9 | 0.90 | 1 | 96.4 | 69.4 |
| 16:12:00 | LNX | 2 | 0 | 0 | 0 | 0.0 | 116 | 106 | 93.0 | 51.1 | 0.91 | 1 | 97.5 | 65.5 |
| 16:10:00 | LNX | 27 | 0 | 0 | 3 | 0.0 | 116 | 106 | 99.0 | 61.5 | 0.60 | 1 | 67.2 | 52.8 |
| 16:09:00 | LNX | 31 | 0 | 0 | 1 | 0.0 | 116 | 106 | 98.0 | 60.6 | 0.62 | 1 | 67.9 | 54.2 |

# Experiment #3

## 100 Linux servers running

### Test run is shown from 16:04 - 16:06

```
Report: ESASSUM      Subsystem Activity
Monitor initialized:                        on P390 serial 00000     First
-------------------------------------------------------------------------------
         <---Users----> Transactions <Processor> Storage (MB) <-Paging-->
         <-avg number->    Per  Avg. Utilization Fixed Active <pages/sec>
Time       On Actv In Q Minute  Resp Total Virt.  User Resid. XStore DASD
-------- ---- ---- ---- ------ ----- ----- ----- ----- ------ ------ ----
03/14/06
16:01:00  116  107 78.0 3350.3 0.756    70    56   7.1  103.1     28    0
16:02:00  116  106 88.0 3332.1 0.757    70    56   7.1  103.9     25    1
16:03:00  116  107 97.0 2960.1 0.974    79    54   7.1  104.7    158   46
16:04:00  116  107 93.0 2585.4 1.078   100    53   7.3  102.8    435  100
16:05:00  116  107 89.0 2295.9 1.131   100    49   7.1  100.3    567   62
16:06:00  116  106 90.0 2592.5 1.089   100    61   7.2   99.1    367   46
16:07:00  116  106 85.0 3380.6 0.742    77    62   7.1  102.0     41    5
16:08:00  116  106 77.0 3378.0 0.751    68    54   7.1  102.7     14    2
16:09:00  116  106 77.0 3394.8 0.735    68    54   7.1  104.9     27    8
16:10:00  116  106 98.0 3636.4 0.624    68    54   7.1  105.1     16    0
16:11:00  116  106 99.0 3688.6 0.600    67    53   7.1  105.3     24    1
```

# Experiment #3

- Assigned 10 servers per class LNX*
- Test was to “wipe” over all 100 servers
- This interval shows LNX1 being hit

```
Report: ESAUSR2      User Resource Utilization                      rvdheij.com
Monitor initialized:                        on P390 serial 00000     First record analyzed: 03/
-------------------------------------------------------------------------------------------
         <---CPU time--> <--------Main Storage (pages)--------> <------Paging (pages)------->
UserID   <(seconds)> T:V <Resident> Lock <-----WSS-----> <---Allocated---> <---I/O--->
/Class   Total  Virt Rat Totl Activ  -ed Totl Activ  Avg Resrvd Total ExStg  Disk  Read Write
-------- ----- ----- --- ---- ----- ---- ---- ----- ---- ------ ----- ----- ----- ----- -----

LNX      25.67 20.13 1.3  24K 24270    0  23K 22818  228      0  379K 49894  329K  2442  1170
LNX0      1.96  1.73 1.1 1499  1499    0 1404  1404  140      0 36203  1758 34445     0  1170
LNX1      3.27  2.59 1.3 3806  3806    0 3628  3628  363      0 36124  2941 33183   543     0
LNX2      2.53  1.69 1.5  966   966    0  985   985   99      0 36998  2726 34272     0     0
LNX3      2.55  2.08 1.2 4279  4279    0 3872  3872  387      0 37301  3021 34280   602     0
LNX4      2.56  2.06 1.2 2309  2309    0 2324  2324  232      0 38985  5618 33367   311     0
LNX5      1.99  1.75 1.1 1257  1257    0 1179  1179  118      0 38595  7782 30813     0     0
LNX6      2.64  2.10 1.3 2797  2797    0 2547  2547  255      0 38490  6436 32054   256     0
LNX7      2.26  1.91 1.2 1736  1736    0 1620  1620  162      0 38888  6785 32103     3     0
LNX8      2.81  2.06 1.4 2434  2434    0 2246  2246  225      0 38740  6746 31994   217     0
LNX9      3.10  2.16 1.4 3187  3187    0 3013  3013  301      0 38500  6081 32419   510     0
```

# Experiment #3

- Only minimal amount of VDISK is resident (very little swapping)

```
Report: ESAUSPG      User Storage Analysis                          rvdheij.com                    ESAMAP
Monitor initialized:                        on P390 serial 00000     First record analyzed: 03/14/06 16:0
-------------------------------------------------------------------------------------------------------
         <---Storage occupancy in pages---> <--Main Storage page Read/Write--> Pages  <Address Spaces->
UserID   <---Main Storage---> <--Paging---> <-Page Writes to:--> <Page Reads:> Moved  <pages Resident>
/Class   Total   >2gb   <2GB  Xstor   DASD   Xsto   Disk   Migr  Xstor  Disk   <2GB   VirtDisk AddSpce
-------- ------ ------ ------ ------ ------ ------ ------ ------ ------ ------ ------ -------- -------
LNX       24270      0  24270  49894 328930  16943   1170   1170  14363   2442      0     1020       0
LNX0       1499      0   1499   1758  34445   1133   1170   1170    697      0      0      239       0
LNX1       3806      0   3806   2941  33183   1882      0      0    788    543      0      346       0
LNX2        966      0    966   2726  34272   3715      0      0   3574      0      0        0       0
LNX3       4279      0   4279   3021  34280    527      0      0    459    602      0      126       0
LNX4       2309      0   2309   5618  33367   2138      0      0   1418    311      0       85       0
LNX5       1257      0   1257   7782  30813   2009      0      0    941      0      0        0       0
LNX6       2797      0   2797   6436  32054   1435      0      0   1177    256      0       83       0
LNX7       1736      0   1736   6785  32103   1867      0      0   1595      3      0        0       0
LNX8       2434      0   2434   6746  31994   1091      0      0   1591    217      0       18       0
LNX9       3187      0   3187   6081  32419   1146      0      0   2123    510      0      123       0
```

# Conclusion

One can run 100 Linux servers on a P/390 !

- With tuning they remain fairly responsive
  - Some tweaking was necessary
- It does not provide practical value
  - Some techniques apply to modern machines as well
- CPU speed is the biggest problem
- Lack of IEEE Floating Point hurts net-snmp
  - Unable to monitor Linux internal behavior
  - Like driving a car with your eyes closed