

Understanding Linux Storage Analysis and Tuning

- Barton@VelocitySoftware.com
- [HTTP://VelocitySoftware.com](http://VelocitySoftware.com)

“If you can’t Measure it,
I am Just Not Interested™”

Objectives:

- Understand z/VM Storage Requirements
- Understand Linux Storage Requirements
- Know how/where to measure
- Understand Demand Paging
- Determine Requirements
- Understand Measurements
- Suggest tuning measures

PGMBK is page table for virtual storage

PGMBK storage per referenced 1MB segment:

- Two 4k page PGMBK per 1mb segment (8mb/gb)
- 2048 pages/gb (100mb virtual requires 800mb real)
- (1gb Linux server: 8mb PGMBKs)

Locates all user pages in

- ~~Expanded Storage (pre z/VM 6.3)~~
- DASD Paging (and IBR list)
- Main Storage

A pageable PGMBK is eligible for page-out when it maps no virtual pages into real storage

The problem: What pages to page out?

Inactive storage? Linux Storage is not idle

- Extra storage used to cache data and programs
- Linux servers are not idle
- Linux applications poll at 200 times per second
- Which servers are actually doing work if all are “active”
- What pages can be legitimately paged out of real storage?

Determining pages for page out:

- **Active server?** Can not know if server is working or **polling**
- Take least recently used, non modified, non referenced
- Fast page-in (page recovery) very important

Linux Storage management is worst case to virtualize

- “Round robin” storage allocation keeps all storage active
- Oldest unreferenced page
 - Most likely to be paged out
 - Most likely to be next used by Linux
- All storage is used to buffer data, programs
- Small “available list”

The page most likely to be needed by Linux:

- Is most likely to have been paged out

(The answer is to minimize Linux free storage)

Storage Requirements

- System functions require storage
- Work requires programs and data
- More data in storage improves response time

Overcommit (sharing)

- “Expensive” storage is shared in virtual environment
- Storage often used once (initialization), not needed after
- Unreferenced pages of virtual machine can be paged out
- Idle applications and data can be paged out
- **Overcommit (sharing) is a metric for capacity planning,**
- **(Overcommit may mean lack of tuning, extra large virtual machines???)**

To “share” requires paging out:

- Inactive storage
- Inactive applications
- Initialization pages
- Inactive servers

Linux Storage management is worst case to virtualize

- “Round robin” keeps all storage active
- Oldest unreferenced page
 - Most likely to be paged out
 - Most likely to be next used by Linux
- All storage is used to buffer data, programs
- Small “available list”

The problem: What pages to page out?

Inactive storage? Linux Storage is not idle

- Extra storage used to cache data and programs

Inactive servers? Linux servers are not idle

- Linux applications poll at 200 times per second
- Which servers are actually doing work if all are “active”
- What pages can legitimately be paged out of real storage?

The page most likely to be needed by Linux:

- **Is most likely to have been paged out**

Problem: Impossible to choose the correct page to page out

Strategy / Best practices in past if overcommit high

- Need high speed page recovery

Expanded Storage was used for “30 second test case”

- Pages migrated to disk after 30 seconds
- **Minimum 20% of storage reconfigured to Expanded Storage**
- Page-in from expanded storage was synchronous and FAST
- Pages migratable to disk after 30 seconds unreferenced

“New” strategy is IBR (z/VM 6.3).

- **VERY LIMITED. 5% is the max**
- **2% is the default, Go the max!**

Limit virtual machine sizes

- PGMBKs – cost 8mb (PTRM address space) per virtual GB
- PTRM address space is pageable

Limit the amount of main storage used by MDC:

- SET MDCACHE STORAGE **minM** maxM

High Level, UCD

- Standard Linux system storage at a high level - ESAUCD2

Linux system storage

- Linux system storage details - ESALNXR

Linux process storage

- By process

Preview, Linux Storage

- Storage overview (ESAUCD2)
- Storage Details (ESALNXR)
- Process Storage (ESALNXP)

Report: **ESAUCD2** **LINUX UCD Memory Analysis Report** Velocity Sof

```

-----
Node/      <-----Storage Sizes (in MegaBytes)-----
Time/      <--Real Storage--> <-----SWAP Storage-----> Total <-----Storage in Us
Date       Total  Avail Used  Total Avail Used  MIN  Avail CMM  Buffer Cache 0
-----
18:30:00
*** Nodes *****
lxsugar   999.4   7.6 991.8 154.9 151.3   3.6 15.6 158.9      0   85.7 648.1 2
mail      8112.8 2318 5795   0     0     0 15.6 2318      0 639.8 907.9
mongo01   3849.8 983.3 2866 371.9 309.6 62.3 15.6 1293      0 150.6 1130
opensuse  15846 160.1 15686 8192 8192 0.3 15.6 8352      0 1524.5 8392
REDHAT6X  996.8 13.8 983.0 495.8 380.4 115.5 15.6 394.2     0 114.7 724.1 1
redhat7   994.0 411.5 582.4 1124 1124 0 15.6 1535      0 1.1 472.6 1
rhel64v   996.1 66.3 929.8 2047 2034 12.5 15.6 2101      0 103.3 39.6 7
rhel7v    2002.3 101.2 1901 2064 766.0 1298 15.6 867.2     0 0 253.0
sles11v3  868.8 88.0 780.8 2046 1406 639.7 15.6 1494      0 3.3 27.7 7
sles11x3  493.2 132.8 360.4 867.9 867.9 0 15.6 1001      0 141.6 149.5
  
```

Preview - Linux Storage details

Report: **ESALNXR** **LINUX RAM/Storage Analysis Report** Velocity Sof
 Monitor initialized: 04/15/21 at 00:00:00 on 8562 serial 040F78 First record

```

-----
Node/      <-----Memory in megabytes-----> <-Kernel (MB)-> <-Buffers (MB)
          <---Cache---><---Anonymous---> Stack<-Slab-->
Time      Total Free Size Actv Swap Total Actv Inact Size Size SRec Size Dirty B
-----
18:30:00
mongo01   3850  983 1130  939 26.9  1464 1333 201.3  3.5 57.3 46.3  151  0.7
opensuse 15846  160 8392 4346  0.1 915.4  426 512.0  6.2  54  477 1525  0.0
REDHAT6X  930.4 13.0  676  308  2.5  41.8  62.0 154.7  2.7 51.5 41.0  107  0.0
redhat7   994.0 412  473  328  0  40.8  40.9  56.0  2.6 46.8 26.9  1.1  0
rhel64v   996.1 66.2 39.6 74.6  1.2  14.0  1.1  13.9  1.8  101 42.9  103  0.0
rhel7v    2002  101  253  105 10.0  1437 1142 407.7  4.0  112 67.7  0  0.0
sles11v3  868.8 88.0 27.7 17.4 51.6  106.0 44.6  69.6  2.6 35.6  8.5  3.3  0.0
sles11x4  492.8  102  235  160  0  26.8  26.8  0.7  1.4 31.2 23.2 78.1  0.0
sles12    3374  124 2259 1557  2.7 534.0  483 459.6 30.8  153 51.8  110  0.1
sles12v   995.6  101  440  206  8.1 339.2  162 230.1  2.0 73.9 51.2  0.0  0.0
sles12x3  820.9  182  334  377  0  38.5 38.7  42.2  2.5 88.9 70.5  154  0.0
  
```

Linux admins oversize

Linux data shows

- Real storage
- Available storage
- Swap storage
- “cache”

Some Swapping is “good”

- If not swapping,
- reduce vm size
- Use CMM to reduce

Watch for opportunities

- HIGH available
- No swap

Report: **ESAUCD2** **LINUX UCD Memory Analysis** Velocity Software Corpo
 Monitor initialized: 10/03/14 at 07:22:27 on 2 First record analyzed:

```

-----
Node/      <-----Storage Size (MB)----->
Time/      <--Real Storage--> <-----SWAP Storage--Storage in Use----->
Date       Total  Avail Used  Total Avail Used  Buffer Cache Ovrhd Shared
-----
07:24:00
ORAap042  8041.5  475.9  7566  1130  1130  0.1  183.5  1512  5870  0
ORAap044   13069  7131  5939  6888  6888  0  233.0  3913  1793  0
ORAap046  8041.5  2091  5951  1130  1130  0.1  260.9  3423  2267  0
ORAap048  8041.5  2291  5751  1130  1130  0  224.8  3347  2179  0
ORAap050  8041.5  529.3  7512  1130  1130  0.1  186.9  1577  5749  0
ORAap052   10046  642.8  9403  8172  8172  0  226.5  3958  5218  0
ORAap054  8041.5  1235  6807  3036  2878  158.3  139.9  319.3  6348  0
ORAap056  8041.5  818.5  7223  5604  5592  12.2  156.4  968.3  6098  0
ORA1101b   12062  64.0  11997  4942  4758  183.6  727.5  10024  1246  0
ORA1201a   12062  218.9  11843  4942  4438  503.7  152.4  7170  4520  0
ORA1202a   12062  1668  10394  4942  4399  543.3  137.3  6435  3822  0
ORA1203a   12062  94.0  11968  4942  4443  498.5  168.6  7582  4216  0
ORA1204a   12062  90.9  11971  4942  3754  1188  70.9  8088  3811  0
ORA1403a   12062  462.1  11599  4942  4420  521.8  180.6  6783  4636  0
ORA1404a   12062  439.3  11622  4942  4442  499.9  103.4  6853  4666  0
ORA1405a   12062  442.5  11619  4942  4471  471.1  127.0  6593  4899  0
WAS2a016   2502.6  89.6  2413  1130  1106  24.2  203.0  243.0  1967  48.0
WAS2a020   2502.6  29.9  2473  1130  1106  24.1  254.3  238.8  1980  47.9
WAS2a024   5520.4  2635  2885  1130  1130  0  776.4  613.3  1496  50.3
WAS2a054   2502.6  22.0  2481  1130  1106  23.4  247.9  274.1  1959  48.5
WAS2a058   2502.6  22.4  2480  1130  1106  23.5  244.5  254.9  1981  48.5
WAS2a062   6528.3  3687  2841  1130  1130  0  762.0  591.8  1487  50.3
WAS2a114   2502.6  17.7  2485  1130  1106  23.6  219.6  267.6  1998  48.4
WAS2a118   2502.6  17.6  2485  1130  1106  23.6  260.5  264.1  1960  48.2
  
```

Preview - Linux Process Storage details

Report: **ESALNXP** LINUX HOST Process Statistics Report Velocity Software Co
 Monitor initialized: 04/15/21:00 on 8562 serial 040F78 First record analyze

```

-----
node/      <-Process Ident-> N<-----CPU Percents-----> <-----Storage
Name      ID   PPID   GRP  V Tot  sys user syst usrt  Size RSS Peak Swap Data
-----
18:30:00
mongo01      0     0     0  14.8 1.18 13.2 0.03 0.31 7248 1544 113K  727  78K
  mongod    10889     1 10887  5.75 0.60  5.15   0   0 2653 1307  40K  429  37K
   java    51013   8515   8515  4.94 0.31  4.62   0   0 1665  155  16K   0  14K
   java    51596   8515   8515  3.61 0.20  3.41   0   0 1743  186 8985   0 8053
opensuse      0     0     0  10.0 8.75  1.26 0.00 0.01  33K 5900 537K   0  38K
  gsd-colo  1909   1791   1776  1.13 0.00  1.13   0   0  706   84  11K   0 1773
 VBoxHead 24298 24280 24298  8.61 8.61   0   0   0 5824 4237  87K   0 2089
REDHAT6X      0     0     0  0.72 0.34  0.27 0.07 0.05  16K 1205 227K  641  14K
rhel7v        0     0     0  2.46 0.41  1.69 0.25 0.11  43K 1643 676K  20K 252K
   java    2028     1  1321  1.22 0.04  1.18   0   0 3848  865  58K 2054  55K
sles11v3      0     0     0  0.65 0.19  0.46   0   0 6526  117 105K 9009  27K
sles12        0     0     0  4.60 0.72  3.84 0.03 0.02  76K 5518 1.0M 2918 178K
  ora_mmon  2596     1  2596  3.61 0.32  3.29   0   0  896  403  11K 16.3 1155
sles12v      0     0     0  0.52 0.16  0.32 0.01 0.03  15K  379 239K  10K 144K
  
```

Sizing Objective: Best performance at lowest cost

- Minimize swap
- Minimize paging

First, evaluate existing z/VM storage

- Resident
- Paged out
- VDISK resident

```
Screen: ESAUSPG Velocity Software - VSIVC1 ESAMON 5.14
1 of 2 User Storage Analysis CLASS * USE
```

Time	UserID /Class	<-Storage Occupancy (MB)-->			Paged Out	<-Page I/O-->	
		<---Main Storage---> Total	>2GB	<2GB		Writes	Reads
19:10:00	MONGO8PR	933.89	680.07	253.82	452.43	3	5
	MONGO8S1	936.31	704.77	231.54	341.56	8	58
	MONGO8S2	468.02	348.38	119.64	528.86	57	114
	MONGO804	1610.4	1208.2	402.22	995.58	26	29
	MONG505A	159.35	115.25	44.10	348.14	4	8

(Re)-Sizing Memory for “mongodb” Server

Compare z/VM storage to Linux:

- Objective: Reduce mongo8s2 total storage – 468MB, 0 VDISK
- All servers undersized in RAM and SWAP
- Oops, took 600mb away

```
Screen: ESAUSPG Velocity Software - VSIVC1 ESAMON 5.14
      <-Storage Occupancy (MB)-->
UserID <---Main Storage---> Paged <MegaB Resident>
Time /Class Total >2GB <2GB Out VirtDisk AddSpce
-----
```

Time	/Class	Total	>2GB	<2GB	Out	VirtDisk	AddSpce
19:10:00	MONGO8PR	933.89	680.07	253.82	452.43	68	0
	MONGO8S1	936.31	704.77	231.54	341.56	3	0
	MONGO8S2	468.02	348.38	119.64	528.86	0	0
	MONGO804	1610.4	1208.2	402.22	995.58	0	0

```
Screen: ESAUCD2 Velocity Software - VSIVC1 ESAMON 5.
      Node/ <Real Storage (MB)> <--SWAP Storage (MB)
Time Group Total Avail Used Total Avail Used
```

Time	Group	Total	Avail	Used	Total	Avail	Used
19:10:00	MONGO8PR	974.9	218.3	756.6	371.9	163.0	208.9
	MONGO8S1	974.9	17.6	957.3	371.9	232.9	138.9
	MONGO8S2	974.9	257.6	717.3	371.9	186.3	185.6 ← CMM TARGET
	MONGO804	1982.9	463.7	1519.2	371.9	259.1	112.8

Sizing Objective: Best performance at lowest cost

- VDISK resident + userid resident: dropped 60mb
- Page space dropped 200mb
- Server has 600 MB Available, ready to go

Screen: **ESAUCD2** Velocity Software - VSIVC1 ESAMON 5.

Node/ Time	<Real Storage (MB)> Group	Total	Avail	<--SWAP Used	Storage (MB) Total	Avail	Used	
20:21:00	MONGO8S2	974.9	613.3	361.6	371.9	56.3	315.5	
20:20:00	MONGO8S2	974.9	628.5	346.4	371.9	55.6	316.3	
20:19:00	MONGO8S2	974.9	630.3	344.6	371.9	55.1	316.8	
20:18:00	MONGO8S2	974.9	634.6	340.4	371.9	53.8	318.0	←0 mb cmm
20:17:00	MONGO8S2	974.9	20.1	954.8	371.9	53.3	318.5	←600mb cmm
20:15:00	MONGO8S2	974.9	334.1	640.8	371.9	186.3	185.6	
20:14:00	MONGO8S2	974.9	325.8	649.1	371.9	186.3	185.6	
20:13:00	MONGO8S2	974.9	334.9	640.0	371.9	186.3	185.6	

Sizing Objective: Best performance at lowest cost

- Buffer and cache were the real targets

```
Screen: ESAUCD2 Velocity Software - VSIVC1 ESAMON 5.140 06/02 19:
2 of 3 LINUX UCD Memory Analysis Report CLASS * NODE MONGO8 85
```

Time	Node/ Group	<Real Storage (MB)>			<-----Storage in Use (MB)----->				
		Total	Avail	Used	CMM	Buffer	Cache	Ovrhd	Shared
20:24:00	MONGO8S2	974.9	610.8	364.2	0	1.0	108.8	254.4	8.1
20:23:00	MONGO8S2	974.9	611.7	363.2	0	0.9	108.0	254.3	8.1
20:22:00	MONGO8S2	974.9	612.5	362.4	0	0.8	107.2	254.4	8.1
20:21:00	MONGO8S2	974.9	613.3	361.6	0	0.7	106.5	254.4	8.1
20:20:00	MONGO8S2	974.9	628.5	346.4	0	0.6	92.3	253.6	8.0
20:19:00	MONGO8S2	974.9	630.3	344.6	0	0.4	90.3	253.9	8.0
20:18:00	MONGO8S2	974.9	634.6	340.4	0	0.3	86.6	253.4	8.0
20:17:00	MONGO8S2	974.9	20.1	954.8	620.0	0.2	81.8	872.8	8.0
20:15:00	MONGO8S2	974.9	334.1	640.8	0	64.0	270.1	306.7	10.1
20:14:00	MONGO8S2	974.9	325.8	649.1	9.8	64.0	269.7	315.4	10.1
20:13:00	MONGO8S2	974.9	334.9	640.0	0	64.0	269.6	306.4	10.1
20:12:00	MONGO8S2	974.9	334.4	640.5	1.0	64.0	269.5	307.0	10.1

Sizing Objective: Best performance at lowest cost

- VDISK resident + userid resident: dropped 60mb
- But wait. VDISK inactive.
- Final savings 170mb

Screen: **ES AUSPG** Velocity Software - VSIVC1 ESAMON 5.140 0
 2 of 2 User Storage Analysis CLASS * USER M

Time	UserID /Class	<-Storage Occupancy (MB)-->				<Address Spaces>	
		<---Main Storage---> Total	>2GB	<2GB	Paged Out	<MegaB Resident> VirtDisk	AddSpce
21:23:00	MONGO8S2	353.74	265.55	88.20	323.64	3	0
21:22:00	MONGO8S2	353.54	265.38	88.16	323.66	7	0
21:21:00	MONGO8S2	353.25	265.20	88.05	323.66	7	0
20:20:00	MONGO8S2	309.19	230.91	78.28	322.95	177	0
20:19:00	MONGO8S2	307.59	229.54	78.04	322.95	177	0
20:18:00	MONGO8S2	306.66	228.67	77.99	322.95	177	0
20:17:00	MONGO8S2	294.79	219.50	75.29	322.95	177	0
20:16:00	MONGO8S2	260.25	193.63	66.63	323.15	138	0
20:15:00	MONGO8S2	526.03	392.79	133.24	526.49	0	0
20:14:00	MONGO8S2	525.27	392.05	133.21	526.49	0	0
20:13:00	MONGO8S2	533.95	399.55	134.39	526.55	0	0

Full storage map available (ESASTR1)

- User storage: ESAUSPG
- Linux storage: ESAUCD2
- CMM can be your friend