

Intro to z/VM Performance and Configuration Guidelines

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

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

Introducing Performance Analysis

- Basic concepts
- Where to start

z/VM CPU Analysis and Configuration

- Linux Guidelines

z/VM Storage Analysis and Configuration

- Linux on z Configuration Guidelines
- Virtual CPU
- Storage size

Note that zmap is used for all analysis examples

The “performance” Objective:

- Configure a system to avoid performance problems
- Run well at higher utilization

If there are problems (and there will be problems):

- Understand the problem and correct

Visibility is required. Black boxes are not managed

zVPS: Performance Management is a Process

Performance Analysis

- Understanding system, application performance
- Resolving current performance issues (z/VM, Linux, network)

Operational Alerts

- Supporting 100's/1000's of servers/containers in many locations
- Defining and automating operational support

Capacity Planning

- Providing input to the financial acquisition process

Accounting / Charge back

- Building a financial model for resource billing
- Who is consuming the resource?

Performance Management: Performance Analysis

Why Performance Analysis: Service Level Mgmt

- Diagnose problems real time (**ONE MINUTE GRANULARITY**....)
- PLATFORM SPECIFIC....
- Analyze all z/VM subsystems in detail, real time
 - (DASD, Cache, Storage, Paging, Processor, TCPIP)
- Analyze Linux
 - (applications, processes, processor, storage, swap)
- **Analyze z/OS**
 - Subsystems (disk, CPU), jobs, CICS, DB2
- **Historical view of same data important**
 - Why are things worse today than yesterday?
 - Did adding new workload affect overall throughput?
 - Know who/what is using resource and how to re-allocate

Enterprise Performance Management Requirements

Why Capacity Planning: Future Service Levels (15 minute granularity)

- How many more servers / workload can you support with existing z16(s)?
- What is capacity requirements for an application?
- **Avoid crisis *in advance***

Why Chargeback?

- Distributed chargeback model is by server (does NOT port to Z!)
- Shared chargeback model is by resource consumption
- **Encourages efficient/effective resource use**
- **Align IT to your business model**

Operational Requirements

- Operations will manage 100's (1000's) of servers
- Requires automated performance management
- **Alerts** for processes in loops, disks 90% full, missing processes
- **zALERT always needed (One Minute Granularity)**
- **zOPERATOR, if no enterprise monitor, or do it anyway**

Performance Management Additional Requirements

Correct data (Virtual Linux CPU, SMT - data wrong)

Capture ratios

- When numbers from different sources do not agree
- Which data is valid?

Performance management can NOT be the performance problem

Black boxes are not managed by definition

- Depend on “sales” to decide requirements?
- How to tune if no feedback?

- **Capacity Planning – Plan by “production”, “Test/Dev”**
- **Server Grouping**
 - Critical servers
 - Production vs Test/Dev
 - Group by Application (10-15 groups / classes)
 - Virtual Machines / Linux Nodes
- **Linux Instrumentation (visibility) really helps....**

Performance Problem “Opportunities”

The user says “Performance is bad, must be VM”

- From the “top” user who can’t see anything

So is it z/VM? To start, “visibility” is required.

“visibility” is what Velocity Software provides...

Must understand the metrics from:

- **z/VM Subsystems (intro)**
 - Processor, Storage, Paging, DASD I/O
- **Linux Subsystems (basics)**
 - Processor, RAM, file systems, network
- **Network (advanced)**
- **Applications (more advanced)**
 - Java, WAS, Oracle, MQ, DB2, postgres, gpfs

The Performance Analysis Flow Chart

Analysis starts with “Define the problem?”

- Describe the problem (what user(s) impacted, what time)

System Configuration

- Processor model, CPU type, SMT support
- Number of processors, storage size

Wait states for those impacted

Loads on the system and subsystems

Subsystem Analysis based on wait states

- DASD, Storage, Paging, Processor, Network

Know the configuration: ESAHDR

```
Monitor file created:                11/21/22  22:00:00

z/VM Version: 7                      Release 2.0 SLU 2201
TOD clock at last IPL:              09/26/22  06:08:25

System Identifier                    VMXXX3
Machine Model/Type                   Z15:8561/704
Multithreading Status(SMT): Enabled
  Core Thread count:                 2
  Enabled Count:                      2

System Sequence Code                 000000000002F81F
Processor 0 model/serial              8561-704 /xxxx1F Master
Processor 1 model/serial              8561-704 /xxxx1F
.....
Processor 28 model/serial             8561-704 /xxxx1F
Processor 29 model/serial             8561-704 /xxxx1F

CPU Cycles/ns:                       5200
CPU Cycles/ns (GP):                  5200
  Operating on IFL Processor(s)      ←-----
Totals by Processor type:
<-----CPU-----> <-Shared Processor busy>
Type Count Ded shared total assigned Ovhd Mgmt
-----
CP      4   0     4  35.2    33.4  1.0  1.9
IFL     60  48    12 263.0   258.9 4.0  4.1
ICF     2   1     1   2.9     0.6  0.0  2.3
ZIIP    1   0     1   1.1     1.1  0.0  0.0

Number of logical partitions defined: 23

Main Storage installed (MB):          2867199
Main Storage Generated (MB):          2867199
```



Common configuration problems

- IFLs? SMT
- Real Storage (2.7TB)
- Release significant
- Master processor significant

Know the configuration: ESAHDR

```
z/VM Version: 7                               Release 3.0 SLU 2201
TOD clock at termination                       01:00:00
Abend code of last termination
TOD clock at last IPL:                        06/18/23 09:59:24
System Operator:                              OPERATOR
Time zone adjustment from GMT:                1 hours

Machine Model/Type                            Z16:3931/400
Multithreading Status(SMT): Enabled
Core Thread count:                            2
Enabled Count:                                2

System Sequence Code                          000000000008C9C8
Processor 0 model/serial                      3931-400 /11C9C8 Master
Processor 1 model/serial                      3931-400 /11C9C8
Processor 6 model/serial                      3931-400 /11C9C8
...
Processor 7 model/serial                      3931-400 /11C9C8

CPU(GP) Capability Factor:                    3002
CPU Cycles/ns:                               5200
Operating on IFL Processor(s)

Totals by Processor type:
<-----CPU-----> <-Shared Processor busy>
Type Count Ded shared total assigned Ovhd Mgmt
-----
IFL      70  0    70 1655 1603.7 32.8 51.7
Number of logical partitions defined:        22

Main Storage installed (MB):                  151552
Main Storage Generated (MB):                  151552
```

(65)

Common configuration problems

- LinuxOne, IFL Only
- IFLs? SMT
- Real Storage (148GB)
- Release significant
- Master processor significant

CP Monitor provides user Wait states

- State Sampling – once per minute per user
- Hi-Frequency State Sampling – once per second per vCPU
 - (900 samples per vCPU per 15 minute period)

All virtual cpus have a “state” at all times

Determine if problem is for a user or system

Waits reported by server, class, top user

- System: What impacts the whole system?
- **User classes**: Does one class stand out? (database vs WAS?)
- Users: Is there something specific?
- Recognize “running” to wait comparison

Wait state (queue) analysis -> where to focus

- Running / CPU Wait -> CPU Subsystem
- Simulation wait (master processor) -> CPU Subsystem
- Page wait -> Paging/Storage subsystems
- Asynchronous I/O, SIO -> DASD subsystem
- ~~Eligible~~ – SRM Settings – has no value after 6.3

Normal idle wait states

- TCPIP, Linux: test idle
- Traditional servers: SVM (service machine wait)
- Traditional users: idle (not in queue)

Wait States: ESAXACT

Report: **ESAXACT** Transaction Delay Analysis Veloc
 Monitor initialized: 04/15/11 at 10:00:00 on 2097 serial 726xx First

-----<-----Percent non-dormant (Wait states)----->-----																	
UserID	<-Samples->		<-----Percent non-dormant (Wait states)----->														Pct
/Class	Total	In Q	Run	Sim	CPU	SIO	Pag	E- SVM	D- SVM	T- SVM	CF	Tst Idl	<Asynch>		Ldg	Elig	
11:00:00	1335	1011	4.0	0.2	0.6	0	0.5	0	0	0.1	0	91	0.1	.	.	0	
Hi-Freq:	116K	59208	4.2	0.0	1.9	0.0	0.3	0	7.9	0.1	0.0	89	0.4	0.1	0.2	0	
Key User Analysis																	
TCPIP	893	285	0.4	0	2.5	0	0	0	0	0	0	97	0	0	0	0	
User Class Analysis																	
*Servers	12502	822	0.7	0.1	1.0	0.2	0	0	17	4.5	0	93	0	0	0	0	
*SOA	35720	31695	7.0	0.0	2.2	0	0.3	0	0	0	0.1	88	0.6	0.0	0.1	0	
*ITM	36613	23570	1.1	0.0	1.7	0	0.3	0	0	0	0	91	0.1	0.2	0.4	0	
*TheUsrs	24111	480	0.2	0.8	1.3	0	0.6	0	26	5.2	0	91	0.2	0	0.2	0	
Top User Analysis																	
LN XUWA01	893	893	71	0	2.8	0	0.1	0	0	0	0	24	1.7	0.4	0	0	
LN XUWA03	1786	1786	28	0.2	5.5	0	1.2	0	0	0	0.6	57	7.2	0.1	0.1	0	
LN XUWA02	1786	1786	27	0.1	3.6	0	0.1	0	0	0	0.4	69	0.1	0	0.1	0	

Wait state (queue) analysis -> where to focus

- Running / CPU Wait -> CPU Subsystem
- Simulation wait (master processor) -> CPU Subsystem
- Page wait -> Paging/Storage subsystems

Wait State "RUN": ESAXACT

Report: **ESAXACT** Transaction Delay Analysis Veloc
 Monitor initialized: 04/15/11 at 10:00:00 on 2097 serial 726xx First

<-----Percent non-dormant (Wait states)----->																	
UserID	<-Samples->		<-----Percent non-dormant (Wait states)----->														Pct
/Class	Total	In Q	Run	Sim	CPU	SIO	Pag	E- SVM	D- SVM	T- SVM	CF	Tst Idl	<Asynch>		Ldg	Elig	
11:00:00	1335	1011	4.0	0.2	0.6	0	0.5	0	0	0.1	0	91	0.1	.	.	0	
Hi-Freq:	116K	59208	4.2	0.0	1.9	0.0	0.3	0	7.9	0.1	0.0	89	0.4	0.1	0.2	0	
Top User Analysis																	
LN XUWA01	893	893	71	0	2.8	0	0.1	0	0	0	0	24	1.7	0.4	0	0	
LN XUWA03	1786	1786	28	0.2	5.5	0	1.2	0	0	0	0.6	57	7.2	0.1	0.1	0	
LN XUWA02	1786	1786	27	0.1	3.6	0	0.1	0	0	0	0.4	69	0.1	0	0.1	0	

Run is "high" – Not waiting for anything

- Need more cycles in shorter amount of time
- Faster engine
- Split workload into multiple virtual CPUs

CPU Wait – Waiting for CPU

SIM Wait – Waiting for master processor CPU

Reasons for wait:

- Total IFLs on CEC Utilization
- LPAR Entitlement
- LPAR Utilization
- SMT Not Enabled for 2 threads
- Shares (usually only needed at high utilization)
- And.... Something else

Wait States: ESAXACT

```

Report: ESAXACT      Transaction Delay Analysis
Monitor initialized: 01/12/21 at 14:00:00 on 3906 serial 35B
-----
                                <-----Percent non-dormant (Wait states)
UserID  <-Samples->
/Class  Total   In Q  Run Sim CPU SIO Pag  E-   D-   T-   Tst
-----  -----  ---  ---  ---  ---  ---  ---  ---  ---  ---  ---
01/12/21
14:01:00    140    126    0  56    0    0    0    0    0    0  0.8  44
Hi-Freq: 11400    7780  1.4  40   20   10  0.9    0  1.6  0.1  4.3  21
***User Class Analysis***
LTCUST      342    342  3.8  35   27    0    0    0    0    0  4.1  29
LTDEVL      342    342  2.6  15   31   3.2  1.2    0    0    0   16  24
LTVPARS     855    842  1.5  26   23   12  0.6    0    0    0   14  21
zRTF        2280   2165  1.0  33   19   16  0.1    0    0    0   0.5  27
VPWSPapp    2451   2450  0.7  56   13   10  0.9    0    0    0   1.1  16
LTT4*       969    918  2.1  30   31   8.2  3.3    0    0    0   4.4  18
*TheUsrs    3477    541  2.4  40   28   0.4  0.6    0  5.3  0.9  7.8  19
***Top User Analysis***
VMMONIT      57     35  5.7  66   17  2.9    0    0    0  5.7  2.9
LTDEVLB     171    171  2.9  12   24    0  2.3    0    0    0   33  25
U1PC        114    114  4.4  25   50  1.8    0    0    0    0  19
VPWSP38     114    114  4.4  20   35  0.9   18    0    0    0    0  18

```

Wait state (queue) analysis -> focus on CPU!

- CPU Wait very high for system, for top users
- Sim wait VERY high

CPU Utiliaztion

Report: ESALPARS Logical Partition Summary V
Monitor initialized: 01/12/21 at 14:00:00 on 3906 serial 35B158 F

```
-----<-----Logical Partition-----> <-Assigned> Entitled
                Virt CPU <%Assigned> <---LPAR--> CPU Cnt
Time           Name      Nbr CPUs Type Total  Ovhd  Weight  Pct
-----
01/12/21
14:01:00      Totals:    00   25  CP  682.1  25.9   1001  100
                VM3      35    5  CP  362.2   0.3   156  15.6   4.99
```

```
Totals by Processor type:
<-----CPU-----> <-Shared Processor busy->
Type Count Ded shared Total Logical Ovhd Mgmt
-----
CP      32   0    32  716.9   656.3  25.9  34.8
```

Wait state (queue) analysis -> focus on CPU!

- CPU Wait very high for system, for top users

32 CPUs shared,

VM3 has access to 5, Entitled to 5, using 3.6

CPU Utiliaztion

```
Report: ESACPUU          CPU Utilization Report
Monitor initialized: 01/12/21 at 14:00:00 on 3906 serial 35B158
-----
          <-----Load----->          <-----CPU (percentages)---->
          <-Users-> Tran      CPU      Total  Emul  User   Sys  Idle
Time      Actv In Q  /sec  CPU  Type  util  time ovrhd ovrhd time
-----
01/12/21
14:01:00   95  126  2.0  0  CP    75.6  0.5  51.3  23.8  24.2
              1  CP    72.4  1.2  69.1   2.0  27.5
              2  CP    71.5  1.3  67.0   3.2  28.2
              3  CP    70.4  1.4  66.5   2.5  29.4
              4  CP    72.1  1.9  69.0   1.2  27.8
-----
System:                361.9   6.2  322.9   32.8  137.1
```

Wait state (queue) analysis -> focus on CPU!

- CPU Wait very high for system, for top users
- **Why is there CPU wait when not that busy?**
- Linux: Too many virtual cpus waiting for each other
- The secret command....

Wait States: ESAXACT

Report: ESAXACT Transaction Delay Analysis
 Monitor initialized: 11/06/23 at 17:00:00 on 3931 serial 11C9

```

-----
                                <-----Percent non-dormant (Wait states)-
UserID  <-Samples->
/Class  Total   In Q  Run Sim CPU SIO Pag  E-  D-  T-  Tst
                CF Idl
-----
11/06/23
17:01:00      36    27   22   0  7.4   0   0   0   0   0   0   70
Hi-Freq:  3480  1682   18  0.2  21  0.1   0   0   12  1.0   0   60
***Key User Analysis ***
TCPIP        60    13    0   0   0   0   0   0   0   0   0   100
***User Class Analysis***
Servers      420    19    0   0   0   0   0   0   0   0   0   100
Velocity     660    30    0   0   0  3.3   0   0   44  40   0   57
TheUsrs      780    13   23   0   0   0   0   0   16  31   0   46
RHEL         60    60   6.7   0  10   0   0   0   0   0   0   83
COREOS     1560  1560   19  0.3  22   0   0   0   0   0   0   59
***Top User Analysis***
LIMEDW2      480    480   42  0.2  20   0   0   0   0   0   0   38
LIMEDW1      480    480  8.3   0  26   0   0   0   0   0   0   66
LIMEDM1      240    240  15  1.3  36   0   0   0   0   0   0   47
LIMEDI1      360    360  5.3   0  8.9   0   0   0   0   0   0   86
LIMEDL1      60     60  6.7   0  10   0   0   0   0   0   0   83
  
```

Wait state (queue) analysis -> focus on CPU!

- CPU Wait very high for system, for top users
- Sim wait low

CPU Utiliaztion

```
Report: ESALPARS          Logical Partition Summary          V
Monitor initialized: 11/06/23 at 17:00:00 on 3931 serial 11C
-----
<-----Logical Partition----->
Time          Name          Nbr  Virt CPU  <%Assigned>  <-Thread->  Entitle
-----          -----          ---  ---  ---  ---  ---  ---
11/06/23
17:01:00      Totals:       00   91 IFL   1659   30.1
                RHOSVMD5     11    4 IFL   279.3   2.2   113.3   2   3.52

Totals by Processor type:
<-----CPU-----> <-Shared Processor busy->
Type Count Ded shared  Total  Logical Ovhd Mgmt
-----
IFL      70    0    70 1706.9  1628.8 30.1 48.0
```

Wait state (queue) analysis -> focus on CPU!

- CPU Wait very high for system, for top users

70 CPUs shared,

VMD5 has access to 4, Entitled to 3.5, using 2.8

CPU Utiliaztion

Report: ESACPUU CPU Utilization Report
Monitor initialized: 11/06/23 at 17:00:00 on 3931 serial 11C9C8

```
-----  
      <----Load---->      <-----CPU (percentages)-----  
      <-Users-> Tran      CPU      Total  Emul  User  Sys  Idle  
Time   Actv In Q  /sec  CPU  Type  util  time ovrhd ovrhd time  
-----  
11/06/23  
17:01:00   26 27.0  0.6  0  IFL   55.1  53.3  0.8  1.0  44.0  
              1  IFL   58.4  57.0  0.7  0.8  40.6  
              2  IFL   56.2  54.7  0.7  0.8  42.9  
              3  IFL   54.8  53.4  0.7  0.8  44.2  
              4  IFL   51.8  50.3  0.7  0.8  47.2  
              5  IFL   51.5  50.0  0.7  0.8  47.5  
              6  IFL   54.8  53.2  0.7  0.8  44.2  
              7  IFL   54.5  53.0  0.7  0.8  44.5  
-----  
System:                437.2  425.0  5.6  6.6  355.1
```

Wait state (queue) analysis -> focus on CPU!

- CPU Wait very high for system, for top users
- **Why is there CPU wait when not that busy?**
- Too many linux virtual cpus waiting for each other
- The secret command....

Configuration issues

- Linux guideline, minimize vcpu
- Linux guideline – SET SHARE REL 100/vcpu

Capacity Issues - visibility

Linux configuration - Number of vCPUs?

- Spin locks?
- Linux overhead of managing vCPUs?
- z/VM overhead of managing vCPUs?
- Very small time slices, each waits for it's turn
- Hardware cache pollution?

Linux: How many Virtual Processors?

Linux is multiprocessor capable, thus requiring LOCKs

Global SPIN lock is large issue

- One virtual processor acquires lock
- Other virtual processors attempt to spin
- On 390 – spin converted to Diagnose 44 (now 9C)

Problem easily detected

- High Diagnose -> Instruction Simulation -> SIE
- High TV ratio
- Guideline: Minimize virtual processors

CASE STUDIES>>>>

Linux vCPU – Why does it hurt?

Too many vCPUs hurt because:

- DIAG 9C overhead
- Linux balances across all vCPUs
- Pollutes Hardware cache
- vCPUs will wait for each other
- vCPUs will wait to be dispatched, but do little work

Customer Critsit:

- Customer has excess processor capacity
- Bad performance, “CPU WAIT” (ESAXACT)
- Totals by Processor type on box (25% utilization (ESALPARS))

Totals by Processor type:

```
<-----CPU-----> <-Shared Processor busy->
Type Count Ded shared Total Logical Ovhd Mgmt
-----
IFL      70    0    70 1706.9  1628.8 30.1 48.0c
```

LINUXM1:

- Polling at 15,000 times / second (each “poll” has cpu wait)
- Using 60% of one CPU
- $36 \text{ CPU seconds} / 15,000 = 2.4\text{ms} / \text{dispatch}$

Report: **ESUSR3**

```
-----  
UserID      <Dispatch>  
/Class      <Rate/Sec>  
            Disp  Waits  
-----  
17:01:00    45K  45271  
LIMEDM1     15K  14578  
LIMEDI1     6261  6261
```

Linux vCPU – Why does it hurt?

Report: **ESAUSCP** Virtual Machine VCPU Analysis

UserID	<---CPU time-->				<---Percent						
	CPUvadd	<-Percent->		<-SHARE-->	CPU	<-Samples->		Run	Sim	CPU	
	Cnt	TOT	Virt	Type	Value	TYPE	Total	In Q			
17:01:00	0	430.6	425.0	.	.	.	3480	1682	18	0.2	21
LINUXM1	4	61.53	59.64	REL	400	IFL	240	240	15	1.3	36
CPU-00		15.89	15.26	REL	100	IFL	60	60	17	1.7	35
CPU-01		13.77	13.41	REL	100	IFL	60	60	8.3	3.3	38
CPU-02		17.69	17.19	REL	100	IFL	60	60	15	0	32
CPU-03		14.18	13.78	REL	100	IFL	60	60	18	0	38
LIMEDI1	6	49.66	48.93	REL	600	IFL	360	360	5.3	0	8.9
CPU-00		8.85	8.67	REL	100	IFL	60	60	10	0	10
CPU-01		8.25	8.13	REL	100	IFL	60	60	6.7	0	6.7
CPU-02		7.24	7.11	REL	100	IFL	60	60	1.7	0	10
CPU-03		8.47	8.36	REL	100	IFL	60	60	1.7	0	8.3
CPU-04		10.13	10.04	REL	100	IFL	60	60	8.3	0	10
CPU-05		6.73	6.61	REL	100	IFL	60	60	3.3	0	8.3

Linux balances small tasks across vCPUs

Each vCPU waits and then does a small task

Having more vcpu just means the number of delays increases

Linux vCPU – Why does it hurt?

Report: ESAUSCP Virtual Machine VCPU re Corporate
 Monitor initialized: 11/07/24 at 13:02:24rial 00FBF8 alyzed: 11/07

UserID	<---CPU time-->				<---Percent			<Dispatch>			
	CPUvadd	<-Percent->	<-SHARE-->	CPU	Run	Sim	CPU	<Rate	MS/Disp		
	Cnt	TOT	Virt	Type	Value	TYPE		/Second			
13:04:00	0	4092	4020	.	.	.	24	0.3	16	193K	0.212
LX1140	18	1074	1060	REL	100	IFL	58	0.6	19	20502	0.524
CPU-00		68.33	64.83	REL	6	IFL	70	1.7	18	2453	0.279
CPU-01		69.89	69.17	REL	6	IFL	62	0	22	1156	0.605
CPU-02		48.68	48.16	REL	6	IFL	43	1.7	23	1042	0.467
CPU-03		56.03	55.48	REL	6	IFL	57	0	15	1136	0.493
CPU-04		51.00	50.40	REL	6	IFL	47	0	27	1363	0.374
CPU-05		54.53	54.00	REL	6	IFL	52	0	25	938	0.582
CPU-06		70.10	69.42	REL	6	IFL	65	0	15	1062	0.660
CPU-07		62.87	62.31	REL	6	IFL	63	0	18	904	0.695
CPU-08		64.73	64.18	REL	6	IFL	63	0	23	876	0.739
CPU-09		48.21	47.72	REL	6	IFL	53	1.7	23	923	0.522
CPU-10		53.52	52.95	REL	5	IFL	50	0	23	1045	0.512
CPU-11		51.09	50.54	REL	5	IFL	47	0	18	1092	0.468
CPU-12		62.50	61.87	REL	5	IFL	67	0	15	1088	0.574
CPU-13		76.12	75.45	REL	5	IFL	70	1.7	15	932	0.817
CPU-14		73.40	72.71	REL	5	IFL	75	0	10	1137	0.645
CPU-15		50.87	50.18	REL	5	IFL	53	0	15	1378	0.369
CPU-16		51.32	50.81	REL	5	IFL	50	1.7	17	909	0.565
CPU-17		60.47	59.79	REL	5	IFL	63	1.7	20	1066	0.567

New column MS/DISP
Server busy 60%
Share setting a problem
Very small dispatches
Each one waits

z/VM uses “deadline scheduler”

- Each vcpu gets “target deadline” based on share
- Total inque 3000
- $\text{Deadline} = 3000 / \text{share} * \text{DSPSLICE} / \text{nCPU}$
- Rel share of 6, deadline = 5 seconds / nCPU = 70ms
- **Any work that comes in with lower deadline goes first**
- So 70 ms target for sub millisecond work

Correct Chargeback, accounting for CPU

CPU: What are you measuring? Do you know?

- CPU “thread” numbers are traditional, measured by Linux
- **VSI Prorated** based on **HMC** data
 - Shows SMT is significantly better

```
Report: ESAUSP5           User SMT CPU Consumption Analysis
-----
```

UserID /Class	<-----CPU Percent Consumed		(Total)----->		<-TOTAL CPU-->			
	<Traditional> Total	<MT-Equivalent> Virt	<IBM Prorate> Total	<VSI Prorated> Virtual	<IBM Prorate> Total	<VSI Prorated> Virtual		
17:01:00	430.6	425.0	341.9	337.4	271.0	267.6	216.7	213.9
User Class Analysis								
Servers	0.12	0.06	0.09	0.05	0.07	0.03	0.06	0.03
Velocity	0.58	0.55	0.45	0.42	0.35	0.33	0.29	0.28
TheUsrs	3.53	3.53	3.44	3.44	3.12	3.12	1.78	1.78
RHEL	4.55	4.31	3.59	3.40	2.85	2.71	2.29	2.17
COREOS	421.8	416.5	334.3	330.1	264.6	261.4	212.3	209.6
Top User Analysis								
LIMEDW2	194.6	193.1	154.7	153.6	122.5	121.6	97.92	97.18
LIMEDW1	116.0	114.9	91.64	90.69	73.77	73.05	58.39	57.80
LIMEDM1	61.53	59.64	48.63	47.12	38.21	37.05	30.96	30.01
LIMEDI1	49.66	48.93	39.32	38.73	30.12	29.68	24.99	24.62

Linux: Minimize vCPU to meet requirements
Set and maintain REL share to 100 per vcpu

z/VM: Avoid CPU Wait (default dspslice 10ms)

- Reduce vcpu count and increase work per dispatch best
- Or, Reduce wait time: SET SRM DSPSLICE 3 (default 10)
- Calculate CPU consumed / dispatch rate

Each vCPU waits, and then does a small task

- .4ms average CPU
- (But waits 5ms – 10ms if SMT2)

But wait, there is more...

If 8 threads (4 cores) at 60% busy, queueing theory says:

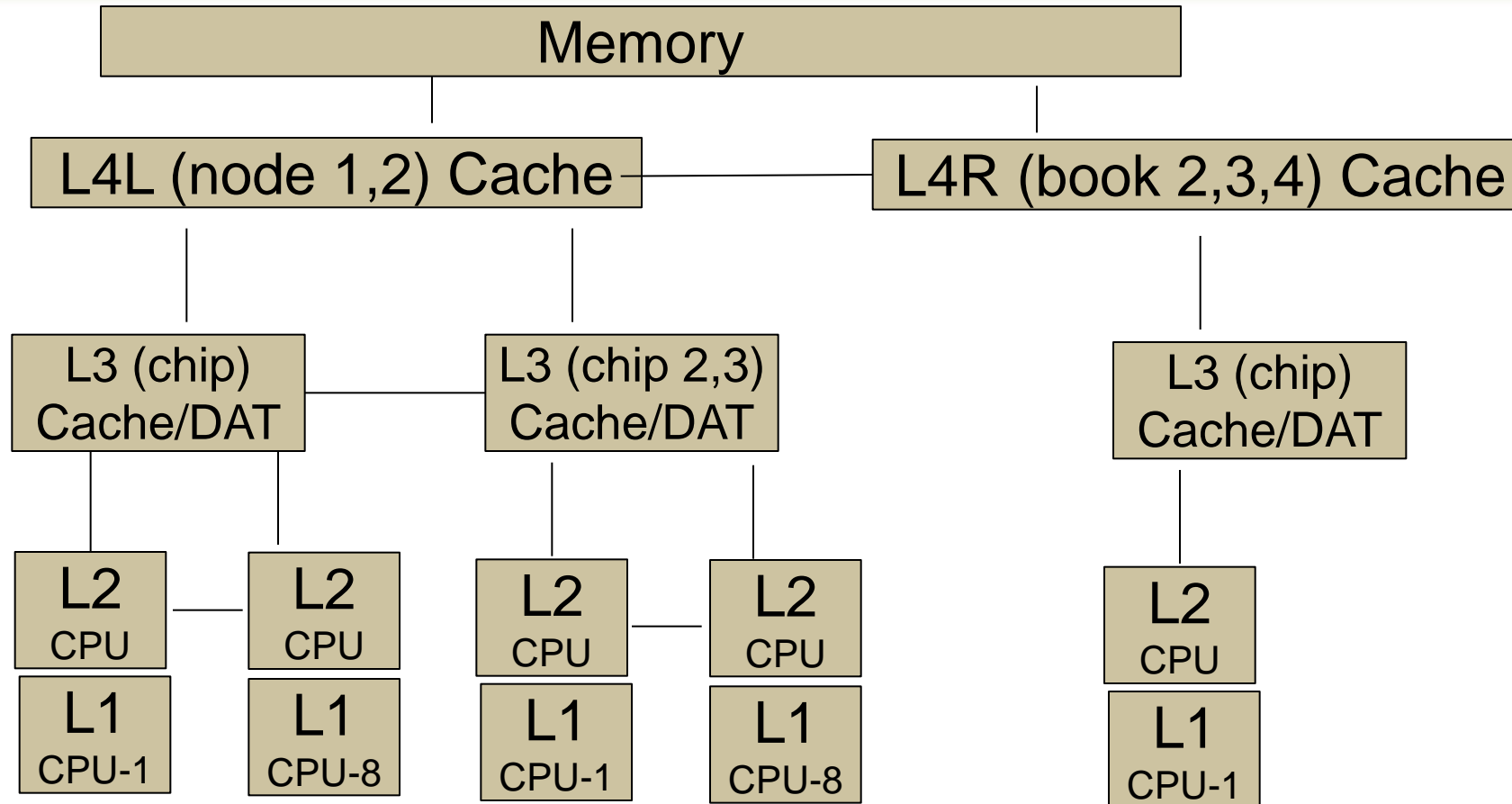
- CPU Queue time should be: .017 time running
- Why is it 2 times running?

Problem: CPU Affinity processing delays 50ms before steal

- 85% of the time there is at least 1 available CPU

The secret command undocumented (yet), used often

- Measured to triple dispatch rate
- Increases capacity



Affinity was meant to better utilize z caching structure

Modlevels Secret Command... – Turn off affinity

```
q syscontrol
DISPATCH THDAFFINITY ON
DISPATCH PREEMPTLOCAL OFF
DISPATCH TSEARLY 50
DISPATCH INCHIPBUSY 50000 ←Delay for steal on chip 50ms
DISPATCH INCHIPDELAY 50000
DISPATCH INNODEBUSY 100000 ←Delay for steal on NODE 100ms
DISPATCH INNODEDELAY 100000
DISPATCH INSYSBUSY 20000 ←Delay for steal on Book 100ms
DISPATCH INSYSDELAY 200000
Ready; T=0.01/0.01 11:24:20
CP SET SYSCONTROL DISPATCH MODLEVEL 0
Ready; T=0.01/0.01 11:24:24
q syscontrol
DISPATCH THDAFFINITY OFF
DISPATCH PREEMPTLOCAL ON
DISPATCH TSEARLY 0
DISPATCH INCHIPBUSY 0
DISPATCH INCHIPDELAY 0
DISPATCH INNODEBUSY 50000
DISPATCH INNODEDELAY 50000
DISPATCH INSYSBUSY 200000
DISPATCH INSYSDELAY 200000
Ready; T=0.01/0.01 11:24:27
CP SET SYSCONTROL DISPATCH MODLEVEL 1...
```

Storage is expensive, overcommit reduces costs

Paging objective: Page out idle / unused pages

- **Inactive storage? Linux storage is not idle**
 - Extra storage used to cache data and programs
 - Linux “dirties” all storage, that z/VM must back
- **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”?

z/VM Paging

- Over commitment of storage causes paging
- **Over commitment of storage reduces cost**
- Paging is common **(manageable)** performance problem

Linux Swapping

- Swapping result of over commitment of Linux storage
- Swapping to VDISK very fast, uses storage when it happens
- Swapping to DASD very slow, always noticeable

Understanding Linux ram (real storage) will save gigabytes real storage

Storage Map, ESASTR1

- **Storage Map to show storage (14GB) use**
 - User resident should be major use
 - Control MDC, Understand vdisk
- **Capture ratio shows accuracy**

Report: ESASTR1 Main Storage Analysis Velocity Software Corporate ZMAP 5.1.2 04/16/21 Pg 2
 Monitor initialized: 04/15/21 at 00:00:00 on 8562 serial 040F78 First record analyzed: 04/15/21 00:00:00

Time	Loggd On	System Storage	Fixed Store	Non-Pgble	Free Stor	Frame Table	<Available> <2gb >2gb	System ExSpc	User Resdnt	NSS/DCSS Resident	<-AddSpace> System User	VDISK Rsdnt	<MDC> Rsdnt	Diag 98	Commit Ratio	Capt-Ratio
-----Users <-----Pages-----> Over-----																
04/15/21																
17:00:00	111	3670016	2878	20879	1153	28672	3170 2501	52291	3387K	35061	75702	0	4729 15418	16K	3.653	0.988
17:15:00	111	3670016	2878	20882	1152	28672	3099 2421	52296	3384K	35078	75713	0	4441 18566	16K	3.653	0.988
17:30:00	111	3670016	2878	20883	1166	28672	3164 2669	52296	3383K	35077	75714	0	4307 19741	16K	3.653	0.988
17:45:00	111	3670016	2878	20872	1147	28672	3195 2389	52298	3381K	35074	75716	0	4270 21989	16K	3.653	0.988
18:00:00	111	3670016	2878	20889	1146	28672	3128 2851	52306	3383K	35079	75722	0	4103 19648	16K	3.653	0.988
18:15:00	113	3670016	2878	20876	1141	28672	3077 2508	52316	3384K	35099	75776	0	4028 19283	16K	4.609	0.988
18:30:00	116	3670016	2878	20880	1075	28672	3137 2544	52360	3349K	32071	122K	0	2118 12337	16K	7.354	0.988
18:45:00	116	3670016	2878	20808	1038	28672	3051 2234	52407	3293K	29914	196K	0	0 47	16K	8.227	0.988
19:00:00	116	3670016	2878	20765	1028	28672	3056 2245	52414	3293K	29082	196K	0	0 127	16K	8.227	0.988
19:15:00	115	3670016	2878	20797	1040	28672	3063 2232	52409	3297K	29522	192K	0	22 73	16K	8.754	0.988
19:30:00	116	3670016	2878	20809	1031	28672	3069 2235	52450	3293K	29065	196K	0	0 6	16K	9.363	0.988

Understanding Virtual Machine Storage

```

Report: ESAUSP2          User Resource Rate Report          Velocit
-----
      <---CPU time--> <----Main Storage (pages)-----> <-Paging (pages)-
UserID <(Percent)> T:V <Resident> Lock <-----WSS-----> Paged <Pgs/Second
/Class  Total  Virt  Rat  Totl  Activ  -ed  Totl  Activ  Avg  2Disk  Read  Write
-----
18:30:00 145.3 133.9 1.1 3.3M 3348K 7048 3.9M 3909K 34K 9147K 27057 15496
  ***Key User Analysis ***
TCPIP    0.15  0.05  3.0 1422  1422  601  817 817.3  817  7750  43.4  8.6
  ***User Class Analysis***
Velocity 5.82  5.43  1.1 3763  3598   5 4593  4271  534 14472 137.4  57.0
SUSE    20.17 19.28 1.0 112K  112K 1534 193K  193K  32K 1048K  2754 828.5
ORACLE  4.66  3.84  1.2 195K  195K  734 381K  381K 190K  473K  2895 936.7
GPFS    12.51 11.68 1.1 195K  195K  975 439K  439K 146K 1332K  4008 1383
TheUsrs 95.37 89.07 1.1 2.6M 2615K 1145 2.5M 2472K 80K 5017K 12958 11022
  ***Top User Analysis***
RHOSBOOT 39.91 38.51 1.0 727K  727K  30  99K 98642  99K  454K  1175  2346
RHOSCP2  8.92  8.20  1.1 250K  250K  19 116K  116K 174K  201K  997.0  1965
RHOSCP1  8.78  8.05  1.1 252K  252K  19 126K  126K 189K  205K  967.6  2005
RHOSCP3  7.83  7.04  1.1 161K  161K  28  48K 47842  80K  125K  1230  1157
  
```

- **Virtual Machine Storage analysis – ESAUSP2 (percent/rate)**
 - Analyze by user – Large consumers?
 - RHOS* users paging too much to get work done
 - RHOS* is OpenShift installation

Understanding Linux Storage Critical

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
    
```


z/VM STORAGE and Paging Architecture Moving Target

z/VM shared storage / Overcommit

- Objective: Page unused pages out to allow re-use
- **Need optimal test before paging to slow disk**
- Optimize page-in when needed (**block paging**)

The problem? Which servers, pages are truly idle

Architectures to choose from:

- Excessive Storage – enough so no paging (expensive)
- Solid State paging device – sort of fast
- Disk paging devices – not fast

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, FAST
- Pages migratable to disk after 30 seconds unreferenced

“New” strategy is IBR (z/VM 6.3)

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

z/VM Storage Management Options

- **System Age List**

- Maximum 5%,
- recommend 5% always
- **SET AGELIST SIZE 5% EARLYWRITES YES KEEPSLOT YES**

```
-Set--AGELIST---.-SIZE--.-n.n--PERCent--.-.
|           | -n.n%-----| |
|           | '-storsize-----' |
| -EARLYWrites--.-Yes-.----|
|           | '-No--' |
| -KEEPSlot--.-Yes-.------'
|           | '-No--'
```

- **CP QUERY AGELIST (default)**

Target size	=	280576K (274M)	2.0% of pageable storage
In use	=	271712K	
Pending writes	=	120296K	
Early writes	=	Yes	
Sizing	=	Variable	

Linux Cache

- Linux avoids I/O by using cache
- Linux will cache gigabytes of data if allowed
- Oracle SGA MUST fit in Linux page cache
- Swap historically was slow SCSI device so storage oversized

Reduce size of Linux Virtual Machine MAJOR Knob.

- Reducing virtual machine size reduces caching of old data
- Define virtual disk for swap
- Virtual Disk paged out when not in use
- Swapping is ok if configured correctly

Reducing virtual storage size may cause swap

- Linux does not swap until out of storage

Swapping to disk

- VERY VERY SLOW
- Other platforms increase storage size because disk is slow
- **Swap to disk if you want to penalize a server**
- Max swap rate maybe 200 on a very good day

Linux Swapping to VDISK

- Not a performance degradation
- 40,000 / second is FAST

Swap Guideline:

- **Define 2 virtual disks, prioritized swap**
- **First one “smaller”, 2nd on 2GB (Insurance)**
- More swap devices for SAP as needed (they are essentially free)
- Use DIAG driver instead of FBA - Reduces I/O by factor of 8

VDISK for swap best practice: Two disks, prioritized – DOUBLE CHECK!

- Two disks per server, goodness
- Should be 1 small swap disk, plus 2nd large disk, goodness
- Prioritized backward though, badness....
- (Address space names have server, virtual address and index)

Owner	Space Name	DASD AddSpc Pages	X- VDSK Blks	Resi- dent	Lock- ed	Stg-> T Migr	Page Slots	Store Blks
Average:								
LINUX1	VDISK\$LINUX1\$\$\$0101\$0041	65791	8738	3.0	0	0	568	0
LINUX1	VDISK\$LINUX1\$\$\$0112\$0042	524K	69905	170	0	0.0	61212	11
LINUX2	VDISK\$LINUX2\$\$\$0101\$0043	65791	8738	3.0	0	0	571	0
LINUX2	VDISK\$LINUX2\$\$\$0112\$0044	524K	69905	85K	0	0.4	346K	2047
LINUX3	VDISK\$LINUX3\$\$\$0101\$0045	65791	8738	3.0	0	0	571	0
LINUX3	VDISK\$LINUX3\$\$\$0112\$0046	524K	69905	2.0	0	0	5767	0
LINUX4	VDISK\$LINUX4\$\$\$0101\$0047	65791	8738	3.0	0	0	571	0
LINUX4	VDISK\$LINUX4\$\$\$0112\$0048	524K	69905	147K	0	0.3	223K	35967
LINUX5	VDISK\$LINUX5\$\$\$0101\$0049	65791	8738	3.0	0	0	568	0
LINUX5	VDISK\$LINUX5\$\$\$0112\$004A	524K	69905	2.0	0	0	4321	0
System Totals:		5901K	39321	233K	0	0.7	669K	38631

Summary: Configuration Guidelines

- **Virtual Machines**

- Relative 100 per virtual CPU
- Shares are a bigger hammer than would appear

- **z/VM System settings**

- SET SRM DSPSLICE 3ms (Validate)
- SET SYSCONTROL DISPATCH MODLEVEL 0

- **z/VM Real Storage Guidelines**

- SET MDC MIN 128m MAX 128m
- Ensure accounting is disabled
- Validate page space

- **Linux**

- Minimize vCPU
- Minimize RAM
- Use VDISK (two) for swap