SMT for z/VM Understanding Capacity Planning and Chargeback

Velocity Software Inc.
196-D Castro Street
Mountain View CA 94041
650-964-8867

Velocity Software GmbH
Max-Joseph-Str. 5
D-68167 Mannheim
Germany
+49 (0)621 373844

Barton Robinson,
barton@velocitysoftware.com

If you can't measure it, I'm just not interested….

The SMT Challenge

Processor cycles wasted for cache load

- **Use them for another "thread"?**
- **Many processes share the core – and cache**
- **Multiple threads share one core – and cache**
- **More processes share core – and cache**

More contention for cache

- **Cache is less effective?**
- **Is there more productive work done?**
- **Or less? What do you charge for?**
- **How do you know? (if you care…)**

SMT Topics

- **Overview**
- **Hardware, Hardware instrumentation**
- **SMT Theory**
- **Hardware Cache, IBM Z**
- **Understanding MFC (smf 113) (CPI,RNI)**
- **Z13 vs z14/z15**
- **Data Validation, Capture ratios**
- **Capacity Planning – what does SMT add?**
- **Chargeback – What are metrics?**

(Please note, zVPS used for ALL analysis)

VELOCITY SOFTWARE

SMT Overview

SMT Objective: Add capacity!

- **How much capacity?**
- **Same everywhere in any Installation?**
- **Workload dependent?**

Measuring capacity requires hardware metrics:

- **PRCMFC in z/VM, SMF 113 for z/OS**
- **Provides hardware metrics**
- **Reported on ESAMFC, ZOSMFC**

SMT Capacity Thoughts

How many cycles used for instructions?

- How many wasted waiting for L1/L2 cache update
- L1 data, L1 instrumentation required prior to exec
- Waiting for DAT

Back to – What is a cpu second?

- We charge for CPU seconds?
- Is it consistent? No!
- How much does it vary (in instructions per second)
- Dependent on workload (cache residency)
- If more contention for cache, more time waiting

What is the CPU Measurement Facility

- Hardware instrumentation
- Statistics by virtual cpu / thread, by LPAR
- 5.18 Monitor records (PRCMFC) (Basic, Extended)
- "Extended" different for z10,196,EC12, z13/14/15
- Shows cycles used, instructions executed and thus CPI

```
Report: ESAMFC          MainFrame Cache Analysis Re
Monitor initialized: 02/27/15 at 20:00:00
-------------------------------------------------
                <CPU Busy> <-------Processor------>
                <percent>  Speed/<-Rate/Sec->
Time      CPU Totl User  Hertz Cycles Instr Ratio
-------- --- ---- ----  ----- ------ ----- -----
20:01:00   0  0.7  0.4  4196M  30.8M 8313K 3.709
```

What is the CPU Measurement Facility (Basic)

CPI: Cycles per Instruction

Report: ESAMFCA MainFrame Cache Hit Analysis
Monitor initialized: 12/10/14 at 07:44:37

		<CPU Busy>		<-------Processor------->			
		<percent>		Speed/<-Rate/Sec->			CPI
Time	CPU	Totl	User	Hertz	Cycles	Instr	Ratio
07:48:35	0	20.8	18.4	5504M	1121M	193M	5.807
	1	21.6	19.6	5504M	1161M	221M	5.264
	2	24.4	22.5	5504M	1300M	319M	4.078
	3	22.4	19.7	5504M	1248M	265M	4.711
	4	19.6	17.6	5504M	1102M	194M	5.683
	5	20.4	18.6	5504M	1144M	225M	5.087
	6	23.9	22.0	5504M	1341M	341M	3.935
	7	17.6	15.4	5504M	949M	160M	5.927
	8	18.5	16.5	5504M	1005M	194M	5.195
	9	22.5	20.6	5504M	1259M	347M	3.629
System:		212	191	5504M	10.8G	2457M	4.733

EC12…

Why you should be interested

```
Report: ESAMFC          MainFrame Cache Analysis Rep
-------------------------------------------------
                <CPU Busy> <-------Processor------>
                <percent>  Speed/<-Rate/Sec->
Time      CPU  Totl  User  Hertz  Cycles  Instr  Ratio
--------  ---  ----  ----  -----  ------  -----  -----
14:05:32    0  92.9  64.6  5000M   4642M  1818M  2.554
            1  92.7  64.5  5000M   4630M  1817M  2.548
            2  93.0  64.7  5000M   4646M  1827M  2.544
            3  93.1  64.9  5000M   4654M  1831M  2.541
            4  92.9  64.8  5000M   4641M  1836M  2.528
            5  92.6  64.6  5000M   4630M  1826M  2.536
               ----  ----  -----  ------  -----  -----
System:         557   388  5000M   25.9G  10.2G  2.542
-------------------------------------------------------
14:06:02    0  67.7  50.9  5000M   3389M  2052M  1.652
            1  67.8  51.4  5000M   3389M  2111M  1.605
            2  69.0  52.4  5000M   3450M  2150M  1.605
            3  67.2  50.6  5000M   3359M  2018M  1.664
            4  60.8  44.5  5000M   3042M  1625M  1.872
            5  70.1  53.8  5000M   3506M  2325M  1.508
               ----  ----  -----  ------  -----  -----
System:         403   304  5000M   18.8G  11.4G  1.640
```

1830 mip cpus (at 100%)

2828 Mip cpus (at 100%) Doing 10% more work

z13…

VELOCITY SOFTWARE

SMT Capacity Thoughts

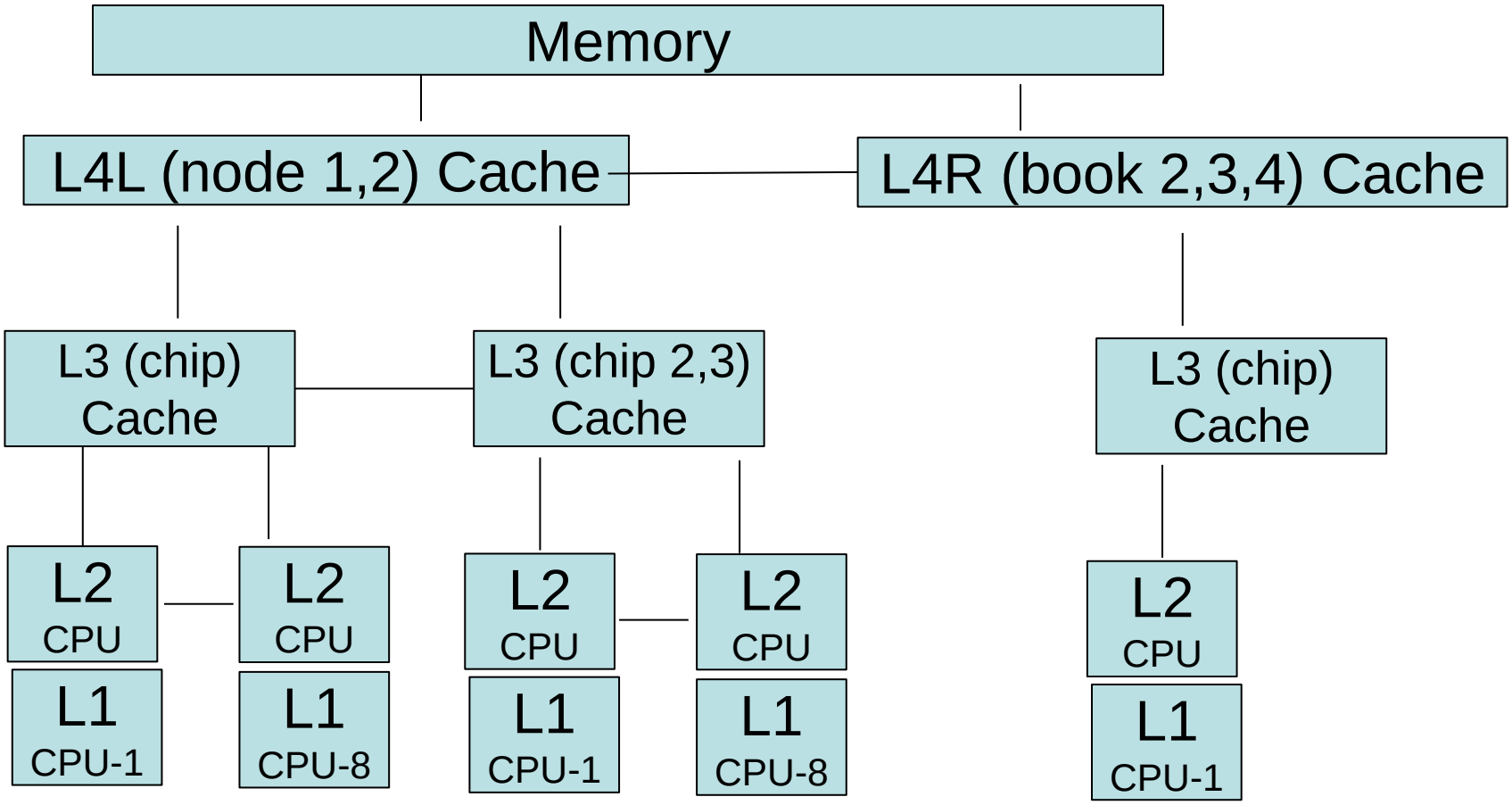
If one thread

- Cycles wasted waiting for L1/L2 cache update
- Cycles wasted waiting for DAT

If two threads

- Wasted cycles could be used by alternate thread
- **If contention for cache / dat, work takes longer**
- Is there an increase in capacity?
- What is performance impact?

Z (z13) Architecture



Question, If 30,000 dispatch / second / cpu, impact?

- Two threads of 30,000 each?

VELOCITY SOFTWARE

Life of an instruction

Cycles unused while:

- L1 Instruction Cache Update
- L1 Data Cache Update
- Address translation

Cycles used for: Instruction

What percent of cycles are “unused”?

This is the potential for SMT

Processor cache comparison

Cache sizes – EC12

- L1: 64k Instruction, 96k Data
- L2: 1MB Instruction, 1MB Data (private, cpu)
- L3: 48MB (Chip, shared 6 CPUs)
- L4: 384MB (Book, shared over 20 CPUs)

Cache Sizes – z13

- L1: 96K Instruction, 128K Data
- L2: 2MB Instruction, 2MB data
- L3: 64MB (Chip, Shared over 8 CPUS)
- L4: 480MB + 224M NIC (per node)

IBM RNI calculations (per John Burg, WSC)

- **Z15/15s RNI =**
  2.9 (0.45*l3p + 1.5*l4lp + 3.2*l4rp + 6.5*memp)/100
- **Z14/14s RNI =**
  2.4 (0.4*l3p + 1.5*l4lp + 3.2*l4rp + 7.0*memp)/100
- **z13 RNI =**
  2.6 (0.4*L3P + 1.6*L4LP + 3.5*L4RP + 7.5*MEMP)/100
- **zEC12 RNI =**
  2.3 (0.4*L3P + 1.2*L4LP + 2.7*L4RP + 8.2*MEMP)/100

Smaller is better, less time loading L1 cache

Higher means less/more opportunity for SMT?

Understanding Hardware Metrics (ESAMFCA)

Reported in "per 100 instructions"

- 0.7% miss is 7 misses per 1,000 instructions
- .02 MEM equates to one memory ref per 5,000
- Example (z15) is low utilization

	--Processor------>			<----------Rate per 100 Instructions					
	<-Rate/Sec->		CPI	**L1**	<---Data source read from---->				
Time	Cycles	Instr	Ratio	**MISS**	L2	L3	L4L	L4R	MEM
09:28:00	488M	384M	1.272	**0.701**	0.590	0.074	0.008	0.000	0.029
09:29:00	485M	385M	1.260	0.701	0.594	0.074	0.007	0.000	0.027
09:30:00	537M	434M	1.238	0.686	0.584	0.070	0.007	0.000	0.024
09:31:00	524M	403M	1.301	0.726	0.608	0.076	0.010	0.000	0.032
09:32:00	535M	420M	1.273	0.720	0.570	0.114	0.009	0.000	0.027
09:33:00	618M	522M	1.184	0.671	0.584	0.061	0.006	0.000	0.020

VELOCITY SOFTWARE

Cycle requirement per source

Z15, Based on RNI calculations (per John Burg)

- **Level 3 = 2.9*.45 = 1.3 cycles**
- **Level 4L = 2.9 * 1.5 = 4.3 cycles**
- **Level 4R = 2.9 * 3.2 = 9 cycles**
- **Memory = 2.9 * 6.5 = 19 cycles**

A lot of cycles can be wasted,

- RNI is a measure

What happens to RNI when add 2nd thread?

- (Yes, gets larger)

VELOCITY SOFTWARE

Cycle requirement per source

What happens to RNI when add 2nd thread?

- Average CPI went from 1.25 to 1.40
- Average RNI went from .55 to .66

Report: **ESAMFCA** MainFrame Cache Magnitudes R

Time	CPU	<CPU Busy> <percent> Totl	User	<-------Processor------> Speed/<-Rate/Sec-> Hertz	Cycles	Instr	**CPI Ratio**	**RNI** From **Burg**	
09:47:00	0	10.9	10.6	5208M	569M	454M	1.254	0.53	
09:48:00	0	11.9	11.6	5208M	621M	523M	1.187	0.42	
09:49:00	0	9.3	9.0	5208M	487M	385M	1.265	0.56	
09:50:00	0	9.5	9.2	5208M	497M	391M	1.270	0.54	
09:51:00	0	9.5	9.1	5208M	497M	380M	1.309	0.65	
09:52:00	t	10.0	9.5	5208M	520M	373M	1.394	0.62	←SMT Enabled
09:53:00	t	11.2	10.8	5208M	587M	448M	1.312	0.48	
09:54:00	t	9.8	9.3	5208M	512M	365M	1.403	0.68	
09:55:00	t	10.5	10.0	5208M	550M	390M	1.411	0.66	
09:56:00	t	10.0	9.4	5208M	521M	366M	1.422	0.75	
09:57:00	t	11.1	10.5	5208M	577M	421M	1.372	0.67	

VELOCITY SOFTWARE

SMT – When to use it?

SMT Announced on z13 without much guidance

Some installations said "good stuff"

- **Oracle, SAP workloads**

Others said "not so good…."

- **Java, Websphere workloads**

The question is why?

And why is z14 (and z15) so much better?

Does SMT provide more capacity?

Which approach is designed for the higher volume of traffic? Which road is faster?

**Illustrative numbers only*

Measurement:

- "person miles"?
- Per minute?

Add lanes and?

VELOCITY SOFTWARE

Does SMT provide contention?

Which approach is designed for the higher volume of traffic? Which road is faster?

**Illustrative numbers only*

Not always more….

VELOCITY SOFTWARE

SMT on z/VM has challenges

- Why is SAP / Oracle better for SMT?
- (30% ITR improvement with SMT)
- Why would z/OS do better with SMT?

Dispatching 30,000 times per second on one thread

- How long is task on CPU? (< 30 microseconds)
- (30 microseconds -> 15,000 cycles, 5k instructions?)
- How long does data remain in L1/L2 cache?
- The more references further out, the worse things get

Relative Nest Intensity – RNI (John Burg, WSC)

- Provides relative wait times

Nesting Steals – Affinity working?

z13, 60 IFLs, LPAR: 14 IFLs (SMT Enabled)

Report: ESAPLDV Processor Local Dispatch Vector Activity

Time	CPU	<VMDBK Steals	Moves/sec> To Master	Dispatcher Long Paths	<-CPU Steals fr <-From Nesting Same	NL1	NL2
19:47:00	0	7442.9	16.5	34163.5	3034	4408	0
	1	5854.5	0	29842.1	2313	3542	0
	2	5363.9	0	23112.3	2466	2898	0
	25	5900.3	0	25649.6	847	5053	0
	26	7022.2	0	28863.4	1035	5987	0
	27	5907.7	0	25927.4	799	5109	0
System:		161948	16.5	757754.4	67K	95K	0

Steals: vmdblks moved to different processor

Dispatcher Long paths:

- vmdblks dispatched (30K/Sec/CPU)
- NL1: Different chip (L3) (check affinity)
- NL2: Different book (L4)No NL2, smaller lpars better?

VELOCITY SOFTWARE

Hardware metric: TLB Analysis – z13

DAT Translation: 30% of the cycles for ONE thread

- Two threads on one core leaves very little for real work

```
Report: ESAMFC        MainFrame Cache Magnitudes Report  ZMAP 4.2.4
-------------------------------------------------------------------------------
             <CPU Busy> <----------> <-Translation Lookaside buffer(TLB)->
             <percent>  Speed/       <cycles/Miss><Writs/Sec>  CPU Cycles
Time      CPU Totl User  Hertz  Ratio  Instr  Data Instr  Data  Cost  Lost
-------- --- ---- ----  -----  -----  ----- ----- ----- ----- ----- -----
07:45:01   0 25.9 24.4  5000M  1.704    159   742  473K  244K 19.77  257M
           1 35.9 34.7  5000M  1.491    138   731  530K  249K 14.17  255M
           2 15.8 13.9  5000M  2.868    206   826  419K  245K 36.30  289M
           3 16.6 15.4  5000M  2.508    212   825  411K  247K 34.90  291M
          23 18.1 17.0  5000M  2.144    197   815  412K  229K 29.44  268M
          24 21.4 19.9  5000M  1.865    114   533  598K  302K 21.35  229M
          25 26.2 24.9  5000M  1.742     98   503  736K  346K 18.71  246M
          26 12.9 11.6  5000M  2.050    154   631  378K  214K 29.92  194M
          27 13.1 11.9  5000M  1.987    156   630  378K  217K 29.64  195M
             ---- ----  -----  -----  ----- ----- ----- ----- ----- ----
System:       514  476  5000M  2.257    176   724   14M 7641K 30.69 7917M
```

One Thread

TLB Analysis – Should SMT be Enabled?

Evaluate other data points:

- z/VM Linux workloads issue: VERY HIGH dispatch
- Why z14 should be great….
- Don’t enable SMT if one thread is consuming your DAT

```
Report: ESAMFC          MainFrame Cache Magnitudes Report
-------------------------------------------------------------
     <CPU Busy> <---- <-Translation Lookaside buffer(TLB)->
     <percent>  Speed <cycles/Miss><Writs/Sec> CPU  Cycles
 ##  Totl User  Hertz  Instr Data  Instr Data   Cost  Lost
 --- ---- ----  -----   ----- ----- ----- ----- ----- -----
 Mem1 907  874  5504M      54   232  117M   36M 29.55 14.8G
 Mem2 1188 1140 5000M     147   364   30M   26M 23.62 14.0G
 VLB4 1703 1366 5000M     185   567   66M   46M 44.59 38.2G
 z13N  216  212 5000M     192   598 3084K 1802K 15.94 1669M
 TCPN  892  757 5000M     217   947   32M   17M 51.46 23.0G
 MTRN  947  868 5000M     265  1283   33M   17M 65.25 30.8G ←
```

VELOCITY SOFTWARE

TLB Problem, z14/z15 Advantages

z13 (z/VM) Problem:

- z/VM does NOT support large pages, needs 256 times TLB
- Linux with java/websphere has VERY high dispatch (30k/sec?)
- Address translation (DAT) required for all parts of instruction
- Some times no cycles left after address translation….

z14 / z15

- The fix
  - If one dat per core is the bottleneck, put on 4…("Quad TLLB")
- Wait for DAT still degrades performance…
- Z14 – no complaints about SMT
- Z15 just as good

Cycles per instruction is critical metric

VELOCITY SOFTWARE

TLB Analysis – Should SMT be

Z14 is "Awesome…."

- IBM doesn't sell value of z14 chip for Linux, z/VM, DAT gives a lot of cycles back.
- 12% DAT cycles (SMT) vs. 30% z13, NO SMT….

ESAMFC MainFrame Cache Magnitudes Rate ZMAP 5.1.0
initialized: 04/08/19 at 19:00:00 on 39064/08/19 19:00:00

	<CPU Busy>		<-------Processor------>				<-Translation Lookaside buffer(TLB)->					
	<percent>		Speed/<-Rate/Sec->				<cycles/Miss>		<Writs/Sec>		CPU Cycles	
CPU	Totl	User	Hertz	Cycles	Instr	Ratio	Instr	Data	Instr	Data	Cost	Lost
0	29.5	28.0	5208M	1535M	822M	1.867	177	284	243K	364K	**9.55**	147M
1	26.8	25.3	5208M	1399M	748M	1.871	178	294	248K	359K	**10.71**	150M
2	37.3	35.1	5208M	1945M	877M	2.219	135	210	446K	818K	**11.91**	232M
3	36.9	34.8	5208M	1925M	914M	2.107	136	212	449K	821K	12.19	235M
4	22.6	20.9	5208M	1181M	530M	2.228	158	263	316K	445K	14.18	167M
5	23.4	21.8	5208M	1219M	590M	2.066	156	260	316K	449K	13.63	166M
6	23.9	21.5	5208M	1248M	615M	2.030	170	284	236K	364K	11.49	143M
7	26.9	25.5	5208M	1402M	730M	1.921	166	265	237K	391K	10.19	143M
8	31.2	29.5	5208M	1628M	792M	2.055	163	257	338K	507K	11.39	185M
9	32.9	31.3	5208M	1715M	878M	1.954	159	247	326K	508K	10.34	177M
10	20.9	19.4	5208M	1093M	504M	2.171	166	276	257K	391K	13.79	151M
11	23.4	22.1	5208M	1223M	658M	1.859	162	265	247K	401K	11.95	146M
12	22.3	20.5	5208M	1162M	526M	2.209	173	302	321K	443K	16.32	190M

VELOCITY SOFTWARE

Capture Ratios for chargeback

If multiple data sources for same “thing”:

- Should they agree?
- If they don’t, who is right?

Metrics that agree:

- LPAR Assigned time (source HMC/SYTCUP)
- z/VM CPU utilization (source z/VM SYTPRP)
- User data (virtual machine data) plus system overhead
- Linux system metrics via snmp (vsi mib)
- Linux process metrics via snmp (vsi mib)

Objective is to know where the resources go

- Can you capture 100%?
- How much fudge factor?
- Which metrics are impacted by SMT???

VELOCITY SOFTWARE

Capture Ratios – LPAR / HMC (ESALPAR)

		<----Logical Processor--->		
VCPU Addr	CPU Type	<---%Assigned--> Total	Ovhd	Emul
zVM 0	IFL	15.7	0.5	15.2
1	IFL	18.8	0.5	18.3
2	IFL	20.7	0.4	20.3
3	IFL	25.1	0.4	24.7
4	IFL	27.2	0.4	26.8
5	IFL	38.4	0.4	38.0
6	IFL	64.8	0.6	64.3
7	IFL	1.1	0.2	0.9
8	IFL	0.8	0.0	0.7
Total	IFL	212.6	3.3	209.3

Physical CPU Management time:

CPU	Percent	Type
140	0.468	IFL
141	0.623	IFL
142	0.606	IFL
143	0.506	IFL
144	0.488	IFL
145	0.449	IFL
146	0.323	IFL
148	0.632	IFL
149	0.263	IFL
150	0.909	IFL
151	0.968	IFL
152	0.940	IFL

LPAR provides (SYTCUP monitor record) for each VCPU

- System (Physical) overhead – not assigned (SYTCUG)
- LPAR (Logical) overhead – assigned to LPARs
- Emulation time – Time LPARs operate

VELOCITY SOFTWARE

Capture Ratios – z/VM (NON SMT)

Report: **ESACPUU** CPU Utilization Report

CPU	CPU Type	Total util	Emul time	User ovrhd	Sys ovrhd	Idle time	Steal time
0	IFL	14.9	12.0	1.3	1.6	84.3	0.7
1	IFL	17.9	16.0	1.5	0.5	81.3	0.8
2	IFL	20.0	18.1	1.4	0.5	79.3	0.6
3	IFL	24.4	22.5	1.5	0.4	75.0	0.6
4	IFL	26.5	24.6	1.4	0.5	72.9	0.6
5	IFL	37.7	35.5	1.7	0.6	61.7	0.6
6	IFL	64.0	60.4	2.8	0.8	35.2	0.8
7	IFL	0.7	0.1	0.1	0.5	99.0	0.3
8	IFL	0.7	0.6	0.0	0.1	99.2	0.1
		206.9	**189.8**	11.6	5.4	688.0	5.1

Report: **ESAUSP2** User data

UserID /Class	<---CPU time--> <(Percent)> Total	Virt	T:V Rat
11:06:00	201.4	**189.8**	1.1
Servers	0.06	0.02	2.6
ZVPS	1.32	1.27	1.0
Linux	199.6	188.2	1.1
IBMStuf	0.17	0.13	1.3
TheUsers	0.23	0.16	1.5

z/VM provides capture ratio of 100.0%

- System overhead – not assigned to users
- User overhead – assigned to users
- Emulation time – user work

User data (ESAUSP2) from USEACT / USELOF

VELOCITY SOFTWARE

Capture Ratios – z/VM – NO SMT

```
ESACAPT      Logical Partition Analysis
-------------------------------------------------------------------
 <----Logical Processor---> <---CPU (percentages----> Capture%
 VCPU  CPU <---%Assigned--> Total  Emul  User   Sys   LPAR
 Addr Type Total Ovhd  Emul  util  time ovrhd ovrhd
 ---- ---- ----- ---- ----- ----- ----- ----- ----- - -----

    0  IFL  15.7  0.5  15.2  14.9  12.0   1.3   1.6    0.98
    1  IFL  18.8  0.5  18.3  17.9  16.0   1.5   0.5    0.98
    2  IFL  20.7  0.4  20.3  20.0  18.1   1.4   0.5    0.98
    3  IFL  25.1  0.4  24.7  24.4  22.5   1.5   0.4    0.99
    4  IFL  27.2  0.4  26.8  26.5  24.6   1.4   0.5    0.99
    5  IFL  38.4  0.4  38.0  37.7  35.5   1.7   0.6    0.99
    6  IFL  64.8  0.6  64.3  64.0  60.4   2.8   0.8    1.00
    7  IFL   1.1  0.2   0.9   0.7   0.1   0.1   0.5    0.76
    8  IFL   0.8  0.0   0.7   0.7   0.6   0.0   0.1    0.95
 ---- ---- ----- ----- -----   ----- ----- ----- ----
Total  IFL 212.6  3.3 209.3 206.9 189.8  11.6   5.4 6  0.99
```

Compare LPAR (SYTCUP) to z/VM (SYTPRP): Capture 99%

- CPU by CPU comparison accurate
- Some scheduling time likely lost

VELOCITY SOFTWARE

Charge back model is NOT 100%

Data requires "fudge factor"

- PRSM Overhead: 1% ?
- LPAR Overhead: 3%?
- LPAR Capture ratio: 1% (capture ratio 99%)
- z/VM System overhead
- z/VM virtual machine overhead
- Virtual machine real work – this is what we charge for

What does SMT do?

Capture Ratios – Linux

Report: ESALNXV LINUX Virtual Processor Analysis Report

Node/ Name	VM ServerID	Node GroupID	<Linux Pct CPU> Total	Syst	User	<Process Data> Total	Syst	User	Capture Ratio
09:28:00									
lxbmq001	LXQMQ001	TheUsers	0.6	0.2	0.3	0.5	0.2	0.3	0.912
lxbpc001	LXQPC001	TheUsers	2.2	0.9	1.3	2.2	1.0	1.3	1.011
lxbsb001	LXQSB001	TheUsers	2.3	0.7	1.6	2.3	0.7	1.6	0.993
lxvmb101	PXVMQ101	TheUsers	1.6	0.5	1.1	1.8	0.6	1.2	1.082
lxvmb102	PXVMQ102	TheUsers	1.7	0.5	1.2	1.6	0.5	1.0	0.922

Capture ratio concept for Linux process table

- Add all the processes, compare to system
- Much more difficult problem than z/VM

Processor utilization – NO SMT

- Numbers agree and make sense
- Can capture virtual machine resources and believe it
- Have value for overheads

SMT Challenges

- Virtual machines share the CPU / core
- The more they share, the slower they go (how slow?)
- Numbers likely not repeatable based on workload
- How much added capacity with SMT for YOUR workload?
- How do you charge?
- (You MUST charge for consumption)

*Processor **Capacity Planning** Concepts*

Processor utilization – **what level is target**?

- Performance –what level of performance required?
- What level of performance management required? Available?
- Capacity Planning – what utilization level is needed financially?

Customer targets

- Target based on performance?
- 80+% plus requires management
- 50% minimizes CPU queue – better performance tradeoff
- Higher utilization is better financially

Capacity planning objective:

- Provide resources to get work done in timely fashion
- Meeting appropriate financial objectives

Processor Measurement Concepts - Utilization

What is "CPU Utilization"? Need to agree on this first?

All zVPS numbers are measured in CPU Seconds

- Percent is always based on CPU seconds divided by wall clock
- What is a CPU second if two threads with SMT?

Impacts measurements of

- LPAR (percent of processor assigned to partition)
- z/VM Virtual Machines (percent of "thread" assigned to virtual machine)
- Linux processes (percent of vcpu)

BUT DO WE AGREE ON WHAT IS IMPORTANT?

- Is it processor utilization?
- Or work completed?

SMT Adds how much capacity?

- How much more throughput?
- Workload dependencies
- How to predict

Z13/14/15 has larger cache

- How long does cache last when 30,000 dispatches / second / processor?
- How much does enabling SMT impact cache?

Capacity Planning Thoughts

How much used capacity at CEC level?

- Total IFL Utilization (ESALPARS / ESALPMGS)
- Totals by Processor type:

```
<---------CPU-------> <-Shared Processor busy->
Type Count Ded shared  Total  Logical Ovhd Mgmt
---- ----- --- ------ ------ -------- ---- ----
CP      11   0     11  892.1    865.2 11.2 15.7
IFL     37   6     31 2466.7   2412.0 30.9 23.8  ← 80% utilization
```

z/VM: One core, Two threads "assigned" 933.7% - 4.1%

- Subtract 138% thread idle (not really excess capacity)
- -> (933%-4)*2 – 138% = **1720**% Thread time (z/VM time)

```
                          <--------Logical Partition------->
                          Virt CPU  <%Assigned>  <-Thread->
Time      Name      Nbr   CPUs Type Total  Ovhd  Idle    cnt
--------  --------  ---   ---- ---- -----  ----  ------  ---
21:25:00  Totals:    00     27  CP   876.3  11.2
          Totals:    00     54 IFL    2443  30.9
          ZVMQAXX    0B     14 IFL   933.7   4.1  138.1     2  ←
```

VELOCITY SOFTWARE

How much used capacity z/VM LPAR?

- Total IFL Utilization (ESACPUU) 1,709%, (capture 99%+)
- User billable Traditional: 1,535% ??

Report: ESACPUU CPU Utilization Report

	<----Load---->					<--------CPU (percentages)-------->					
	<-Users->		Tran		CPU	Total	Emul	User	Sys	Idle	Steal
Time	Actv	In Q	/sec	CPU	Type	util	time	ovrhd	ovrhd	time	time
21:25:00	194	399	0.5	0	IFL	88.4	74.5	1.7	12.2	10.5	1.1
				1	IFL	88.6	76.9	1.9	9.8	10.3	1.1
				2	IFL	89.2	77.7	2.4	9.1	9.7	1.1
				3	IFL	89.2	77.7	1.5	10.1	9.6	1.2
				4	IFL	89.6	78.0	1.7	9.9	9.4	1.0
				5	IFL	89.1	77.7	2.3	9.1	9.9	1.1
				22	IFL	67.2	58.5	1.5	7.1	11.6	21.2
				23	IFL	66.8	58.4	1.4	6.9	12.0	21.3
				24	IFL	74.9	66.4	1.5	7.0	13.9	11.1
				25	IFL	74.4	66.3	1.6	6.5	14.4	11.2
				26	IFL	76.4	68.3	1.3	6.8	12.6	10.9
				27	IFL	75.6	68.2	1.6	5.8	13.4	11.0
System:						1709	1499	36.2	173.6	332.2	759.1

VELOCITY SOFTWARE

Processor Measurements User View

ESAUSP5:

CPU Consumption in percent

- Total all user
- By user
- By Class

```
Report: ESAUSP5      User SMT CPU Consumption Analysi
Monitor initialized: 06/17/20 at 21:23:09 on 3906 ser
-------------------------------------------------------
        <------CPU Percent Consumed   (Total)---->
UserID  <Traditional> <MT-Equivalent> <MT Prorated>
/Class  Total  Virt   Total  Virtual  Total Virtual
-------- ----- -----  -----  -------  ----- -------
21:25:00  1535  1499   1051     1026   1192    1163
 ***User Class Analysis***
Servers   0.04  0.00   0.03     0.00   0.03    0.00
ZVPS      2.38  1.57   1.66     1.04   2.14    1.37
IBMStuf   0.00  0.00   0.00     0.00   0.00    0.00
TheUsers  1532  1497   1049     1025   1189    1162
```

LPAR Assigned Time: 933.7 %

z/VM Thread assigned time: 1720 %

User / virtual machine time: (1499+36=1535)

- Traditional measurements valid, 100% capture ratio

VELOCITY SOFTWARE

Processor Measurements SMT (ESAUSP5)

	<------CPU Percent Consumed				(Total)---->	
UserID	<Traditional>		<MT-Equivalent>		<MT Prorated>	
/Class	Total	Virt	Total	Virtual	Total	Virtual
21:25:00	1535	1499	1051	1026	1192	1163
User Class Analysis						
Servers	0.04	0.00	0.03	0.00	0.03	0.00
ZVPS	2.38	1.57	1.66	1.04	2.14	1.37
IBMStuf	0.00	0.00	0.00	0.00	0.00	0.00
TheUsers	1532	1497	1049	1025	1189	1162
CPU POOL User Analysis						
UTSPOOL	15.30	14.92	10.43	10.17	11.45	11.18
ZVMSHR00	159.4	156.0	107.4	105.0	123.6	120.9

ESAUSR5/ESAUSP5 show SMT user data

Three CPU measures

1) Traditional: Time assigned and dispatched on a thread
2) Time Would take if non-SMT (MT-Equivalent) (**PERFORMANCE!**)
3) Cycles really used (approximately, prorated) (**Capacity,Chargeback**)

How do you do capacity planning? What is 100% busy?

VELOCITY SOFTWARE

Processor Measurements SMT Validity

```
<------CPU Percent Consumed    (Total)---->
UserID    <Traditional> <MT-Equivalent> <MT Prorated>
/Class    Total  Virt    Total  Virtual  Total Virtual
-------- ----- -----    -----  -------  ----- -------
21:25:00  1535   1499    1051     1026   1192    1163
```

ESAUSR5/ESAUSP5 show SMT user data

Equivalent: Time Would take if non-SMT

- (PERFORMANCE ratio 1051 / 1535) – 50% slower

Prorated: Cycles really used (approximately, prorated)

- (Capacity, Chargeback)
- Want to charge for 933% (physical assigned time to LPAR)
- Prorated metrics are too high (1192 / 933)

Expectations?

Low utilization:

- **Capacity not really an issue**
- **Response time should improve**

High utilization – Intense workloads (SAP, Oracle)

- **Capacity should see improvements**
- **Cache utilized well (dedicate engines….)**

High Utilization – polling workload (was, db2)

- **Cache competition very very high**
- **Response time WILL get worse**
- **Capacity may drop – validate with MFC….**

Summary

SMT Capacity Planning

- **Your capacity improvements are "dependent"**
- **Enhancements to capacity measurable**
- **Evaluate each LPAR for SMT value (CPI)**

SMT Chargeback

- **IBM provides metrics at user chargeback**
- **Likely results in overcharging**
- **Develop an added prorate metric**

THANK YOU, And Please Send data for analysis!

Barton@VelocitySoftware.com