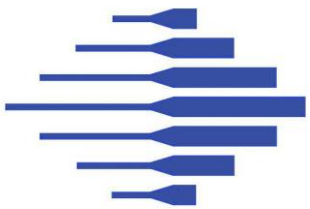
35 YEARS OF PERFORMANCE MANAGEMENT



VELOCITY SOFTWARE

Capture ratios for SMT

Barton Robinson, CTO & Founder
barton@velocitysoftware.com

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

Agenda

Why do I care

- Bogus data in = bogus data out….

Where is data from

- Platform standard interfaces

What did I learn

- PRSM,z/VM,z/VSE,Linux,z/OS,CICS,DB2

What is the correct number for SMT?

- Chargeback
- Capacity Planning

My Focus

Providing Correct Data for System Performance Mgmt:

- Capacity Planning
- Performance Analysis
- Chargeback/Accounting
- Operational Alerts

Business decisions are (hopefully) made based on data

- Better decisions are made on **correct data**…
- Validate the data (Challenge very old "traditional wisdom")
- Understand what is missing – and how much
- When Linux first virtualized, Linux reported CPU incorrectly by up to 2 ORDERS OF MAGNITUDE…. (conclusion: mainframe bad)
- SMT does not exactly add up….

(my) definition of capture ratio

Objective is to know where 100% of resource is used

- System management time ("Physical" overhead)
- Workload management time ("logical" overhead)
- Workload
- IDLE time
- Uncaptured (hopefully zero)

Does platform instrumentation provide 100%?

- PRSM / LPAR: yes
- z/VM: yes
- Linux: yes
- VSE: yes
- z/OS???? DB2?? CICS??

What is the overhead of the platform?

Why care about capture ratios?

Capture ratios validate the data and instrumentation

CPU Data has multiple data sources – do they agree?

- If not, what was missed? Validate the instrumentation …
- PRSM / LPAR – Assigned time vs Operating System reported utilization
  - z/OS smf 70 – what fields show true system overhead?
  - z/VM monitor sytprp – provides measured system overhead

LPAR (HMC data) provides instrumentation for:

- Physical Overhead
- Assigned time
  - Logical Overhead
  - Virtual Assigned time (The Real Work)
- Non-captured time at next level, not reported – about 1%
- But the analysis was very interesting???

Compare Multiple Data Sources

CPU Data has multiple data sources

Do they agree? If not, what was missed?

- z/VM:
  - LPAR data (SYTCUP, SYTCUM)
  - z/VM System CPU (sytprp)
  - z/VM User / Virtual Machine CPU (USEACT,USELOF)
  - Hardware PRCMFC (SMF 113)
- Linux (virtualized linux cpu data was bogus…)
  - Virtual machine data
  - Kernel cpu / irq cpu
  - Process data
- VSE (my very first analysis DOS/VS 34)
  - Virtual machine data (normally)
  - System data
  - Partition data

35 YEARS OF PERFORMANCE MANAGEMENT

Building the "Capture Model"

Every platform has 5 CPU Components

- Hypervisor/OS Management Time (physical overhead)
- Work Management (logical overhead)
- Work time
- IDLE, vs steal time
- Uncaptured – Platform does not define or report

Steal Time

- Virtualized environment, underlying CPU "stolen"
- Not relevant for capture analysis, CPU not utilized

PARK Time

- Not relevant (to me) for capture analysis – CPU not utilized
- But measure unparked time and cycles consumed

Objective is to know what / who is using CPU

VELOCITY SOFTWARE

z/VM Capture Ratio model

5.2 billion cycles per second per cpu

- Where did they all go?
- Set interval = 1 minute to understand variations

For every platform, objective is to accurately show:

- System overhead – Not related to applications
- Application associated overhead
- Application CPU
- Uncaptured – to be identified, objective is zero

If uncaptured CPU is zero (or very low)

- Platform is fully instrumented
- Data can be "trusted" for business decisions
- No "guessing" or "crystal balls"

z/VM LPAR / CEC Data Source

Every z has LPAR data, **One record per VCPU:**

- Assigned Time to LPARs: SYTCUP.LCUCACTM
- LPAR Time (exclude ovhd: SYTCUP.LCUCLPTM
- Add data by LPAR, by Engine Type

<-------Logical Partition------->						
		Virt	CPU	<%Assigned>		
Name	Nbr	CPUs	Type	Total	Ovhd	
VSIVM5	05	2	CP	84.2	0.0	(VSE, z/OS)
VSIVM5	05	2	IFL	1.5	0.1	
VSIVC1	07	1	IFL	23.3	0.1	
VSIVC2	08	1	IFL	0.7	0.0	
VSIVC3	09	1	IFL	0.5	0.0	
VSIVC4	0A	1	CP	2.4	0.0	(VSE)
VSIVC4	0A	1	IFL	0.5	0.1	(linux)
VSIVM1	01	1	IFL	1.3	0.0	
VSIVM2	02	1	IFL	1.2	0.0	
VSIVM3	03	1	IFL	0.4	0.0	
VSIVM4	04	2	IFL	85.5	0.4	
ZOSLP1	0E	2	CP	56.6	0.0	(z/OS)
ZOSLP2	0F	2	CP	56.7	0.0	(z/OS)

LPAR / CEC Capture Ratio model

Full picture of CEC, Add by CPU TYPE (z/VM Model)

- Physical Overhead : SYTCUM.LCUMGTM
- Assigned Time to LPARs: SYTCUP.LCUCACTM
- LPAR Time (exclude ovhd: SYTCUP.LCUCLPTM

Working example (LPARs for z/VM, z/OS, cloud)

Totals by Processor type:

<---------CPU------->				<-Shared Processor busy->			
Type	Count	Ded	shared	Total	Logical	Ovhd	Phys
CP	2	0	2	200.0	199.8	0.1	0.1
IFL	4	0	4	116.0	114.0	0.8	1.1
ZIIP	1	0	1	0.6	0.5	0.0	0.0

CEC Level LPAR Capture ratio – 100%

We DO Know What LPAR consumes the CPU

CP Monitor provides **One record per CPU/Thread:**

- System CPU: sytprp.pfxtmsys (physical overhead (1%)
- User Ovhd: sytprp.pfxutime – sytprp.pfxprbtm (1-2%)
  - (Same concept as PRSM, total assigned time , logical assigned time)
- User CPU: sytprp.pfxprbtm
- IDLE: sytprp.pfxtotwt
- Steal: 100 – (system cpu + user cpu – idle)

		<--CPU (percentages)-->			
CPU	CPU Type	Total util	Emul time	User ovrhd	Sys ovrhd
0	IFL	47.0	45.9	0.6	0.4
1	IFL	50.0	48.9	0.7	0.4
2	IFL	45.5	44.4	0.7	0.4
3	IFL	47.3	46.1	0.8	0.4
4	IFL	42.5	41.0	0.8	0.7
5	IFL	53.6	52.7	0.6	0.3
6	IFL	44.3	43.3	0.6	0.4
7	IFL	56.3	55.3	0.6	0.3
		386.4	377.7	5.4	3.4

z/VM Capture Ratio model

CP Monitor CPU vs PRSM?

- LPAR / PRSM data 100%, What does z/VM see?
- LPAR Data vs z/VM CPU Data: 99.3% (for every CPU…)
- Discrepancy likely setting up and dispatching

<PRSM / LPAR Measurements>					<---z/VM-CPU (percentages)->					VM/
VCPU Addr	CPU Type	<---%Assigned--> Total	Ovhd	Emul	Total util	Emul time	User ovrhd	Sys ovrhd	Stl Pct	PRSM CaptR
0	IFL	62.4	0.7	61.6	61.2	58.4	1.2	1.6	8.71	0.99
1	IFL	62.4	0.6	61.7	61.3	58.6	1.1	1.5	8.59	0.99
2	IFL	62.2	0.6	61.7	61.3	58.7	1.1	1.5	8.20	0.99
.....										
8	IFL	62.3	0.7	61.5	61.1	57.6	1.2	2.2	8.70	0.99
9	IFL	62.6	0.8	61.7	61.3	58.6	1.2	1.5	8.66	0.99
10	IFL	62.5	1.0	61.5	61.1	58.3	1.2	1.6	8.82	0.99
11	IFL	62.6	0.6	62.0	61.6	59.0	1.1	1.5	8.60	0.99
12	IFL	62.5	0.9	61.6	61.2	58.4	1.2	1.6	8.77	0.99
13	IFL	62.2	0.8	61.5	61.1	58.4	1.2	1.5	8.67	0.99
Total	IFL	873.5	10.5	863.1	857.2	818.6	16.4	22.2	121	0.99

VELOCITY SOFTWARE

Charge back model is NOT 100%

Data for chargeback requires "fudge factor"

- PRSM Overhead: 1% ?
- LPAR Overhead: 2%?
- LPAR 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?

Building the "Capture Model"

LPAR Layer (highest layer)

- Overhead "low" – set an alert, high overhead happens
- Capture Ratio 100%
- We know exactly what LPAR is consuming what….

	Mgmt	Logical	Work	Uncaptured	Capture Ratio
HMC/LPAR	.1%	.1%	99%+	1%	99%
z/VM					
z/VSE					
Linux					
z/OS					
CICS					
DB2					

z/VM Capture Ratio model

Compare "system data" to "virtual machine data"

One Record per CPU / Thread

- Virtual Machine "user" CPU Time: sytprp.pfxutime
- VM Problem Time: sytprp.pfxprbtm
- User Overhead: pfxutime - pfxprbtm
- System overhead: sytprp.pfxtmsys
- Idle, "steal"

One Record per Virtual Machine VCPU

- Virtual Machine CPU Time by VM: USEACT.VMDTTIME
- Problem (Virtual) Time: USEACT.VMDVTIME
- USELOF: Logoff
- Add up all the virtual machines, Compare:

CP Monitor, **One record per Virtual Machine/CPU:**

- Add up all the virtual machines, Compare (1995 technology)
- User overhead: Total assigned – total virtual
- User overhead = 383.1 – 377.7 = 5.4
- Add up all users for totals

```
<---CPU time--->
UserID    <(Percent)> T:V
/Class    Total  Virt  Rat
-------- ----- ----- ----
09:01:00 383.1 377.7 1.01

WASM8096 82.95 82.86
WWAS8042 28.56 28.24 1.01
WWAS8038 25.22 24.97 1.01
WWAS8046 24.45 24.20 1.01
WWAS8000 23.82 23.51 1.01
WWAS8005 23.48 23.15 1.01
DB2M8002 23.18 22.81 1.02
.
:
:
```

```
<--CPU (percentages)-->
     CPU    Total   Emul   User    Sys
CPU  Type    util   time  ovrhd  ovrhd
 -   ----   -----  -----  -----  -----
 0   IFL     47.0   45.9    0.6    0.4
 1   IFL     50.0   48.9    0.7    0.4
 2   IFL     45.5   44.4    0.7    0.4
     IFL     47.3   46.1    0.8    0.4
     IFL     42.5   41.0    0.8    0.7
 5           53.6   52.7    0.6    0.3
 6   IFL     44.3   43.3    0.6    0.4
 7   IFL     55.3    0.6    0.3
                    -----  -----  -----
                    377.7    5.4    3.4
```

Building the "Capture Model"

LPAR Layer (highest layer)

- Overhead "low" – set an alert, high overhead happens
- Capture Ratio 100%
- We know exactly what LPAR is consuming what….

	Mgmt	Logical	Work	Uncaptured	Capture Ratio
HMC/LPAR	.1%	.1%	99%+	1%	99%
z/VM	< 1%	<2%	97%+	0%	100%
z/VSE					
Linux					
z/OS					
CICS					
DB2					

Linux Capture Ratio model

Linux data captured via snmp

- System CPU Data by cpu, by system:
- Process Data by process

System data provides

- IRQ, SoftIRQ, Kernal,
- Nice

Process data provides

- CPU data by process, for process and "children"
- Parent process information

Challenge in Linux when process terminates

- CPU added to parents when process terminates

Capture Ratios – Linux

Linux system data vs z/VM data

- Linux Includes IRQ, Krnl time (2%)
- Linux collection time 5-10 seconds prior to z/vm monitor pop

z/VM time (78%)

UserID /Class	<(Percent)> Total	CPU time--> Virt	T:V Rat
RLNX08P0	78.64	74.67	1.1
RLNX08P0	72.33	66.01	1.1
RLNX08P0	53.09	48.31	1.1
RLNX08P0	61.48	56.38	1.1
RLNX08P0	84.47	79.56	1.1
RLNX08P0	93.25	88.30	1.1
RLNX08P0	120.7	116.7	1.0
RLNX08P0	96.25	91.80	1.0
RLNX08P0	83.71	78.61	1.1

Linux time (78%)

<Processor Total	Pct Syst	User	<CPU Krnl	Overh IRQ
78.5	5.5	71.5	0.3	1.3
80.3	5.6	73.0	0.4	1.3
41.0	4.9	34.6	0.3	1.1
63.4	7.2	54.0	0.4	1.7
68.4	5.8	61.0	0.3	1.3
65.5	5.3	59.1	0.3	0.9
127.7	7.1	119	0.4	1.5
98.0	6.5	89.7	0.4	1.4
79.6	6.5	71.4	0.4	1.4

VELOCITY SOFTWARE

Capture Ratios – Linux

Capture ratio concept for Linux process table

- **When Linux process terminates, where does CPU go? – the**
- **Does "crond" get charged anything? No, "children"**
- **Must build process tree**

node/ Name	<-Process Id ID	PPID	<------CPU Percents----> Tot	sys	user	syst	usrt
snmpd	1919	1	0.10	0.08	0.02	0	0
crond	2116	1	0.03	0	0	0.02	0.02
seosd	2515	1	0.02	0.02	0	0	0
selogrd	2549	1	0.02	0	0.02	0	0
db2sysc	3097	3095	1.24	1.01	0.24	0	0
db2fmp	3118	3095	0.02	0.02	0	0	0
db2sysc	3248	3246	0.05	0.02	0.03	0	0
dsmc	30061	1	0.02	0	0.02	0	0

Node/ Name	<----Proces ID	PPID
init	1	1
kthreadd	2	1
migratio	3	2
crond	2116	1
crond	30034	2116
sh	30035	30034
sendmail	30086	30034
postdro	30087	30086
db2syscr	3095	1
db2sysc	3097	3095
db2syscr	3103	3095
db2syscr	3104	3095
db2syscr	3105	3095
db2vend	3107	3095
db2fmp	3118	3095
db2syscr	3246	1
db2sysc	3248	3246
db2syscr	3254	3246
db2syscr	3255	3246
db2syscr	3256	3246
db2vend	3258	3246
db2fmp	3266	3246
login	3326	1
bash	3332	3326

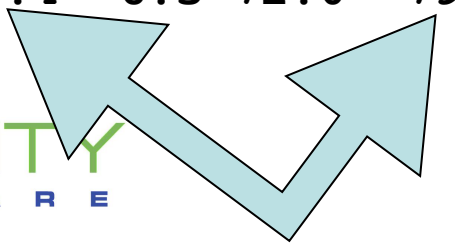
VELOCITY SOFTWARE

Capture Ratios – Linux

Capture ratio concept for Linux process table

- Compare "linux system data" to "Linux Process Data"
- Typically 100%....
- Collecting 1000 processes synchronously has "variation"…
- "system Time" 7-10% ?

Node/ Name	<Linux Pct CPU> Total	Syst	User	<Process Data> Total	Syst	User	Capture Ratio
RLNX01p1	1.7	0.8	0.9	1.8	0.9	0.9	1.056
RLNX02p1	1.3	1.0	0.3	1.3	1.0	0.3	1.000
RLNX03p0	2.0	0.6	1.4	2.0	0.6	1.4	1.000
RLNX04p0	10.6	1.7	8.9	10.7	1.8	8.9	1.007
RLNX05p0	9.0	1.5	7.5	9.0	1.5	7.5	1.000
RLNX06p0	11.6	1.7	9.9	11.6	1.7	9.9	1.000
RLNX07p0	18.3	2.9	15.4	18.3	2.9	15.4	1.000
RLNX08p0	78.4	6.3	72.0	79.8	6.7	73.0	1.018



Building the "Capture Model"

LPAR Layer (highest layer)

- Overhead "low"
- Capture Ratio 100%

	Mgmt	Logical	Work	Uncaptured	Capture Ratio
HMC/LPAR	.5-1%	.5-1%	98%+	0%	100%
z/VM	1-2%	1-2%	95%+	0%	99%
z/VSE	<.1%	6-8%	92-94%	0%	99.9%
Linux	2%	7-10%	90%	< 1%	99% +
z/OS					
CICS					
DB2					

Building the "Capture Model"

LPAR Layer (highest layer)

- Overhead "low"
- Capture Ratio 100%

	Mgmt	Logical	Work	Uncaptured	Capture Ratio
HMC/LPAR	.5-1%	.5-1%	98%+	0%	100%
z/VM	1-2%	1-2%	95%+	0%	99%
z/VSE	<.1%	6-8%	92-94%	0%	99.9%
Linux	2%	7-10%	90%	< 1%	99% +
z/OS	????	13%	85-92%	8-15%?	85-92%
CICS					
DB2					

Building the "Capture Model"

LPAR Layer (highest layer)

- Overhead "low"
- Capture Ratio 100%

	Mgmt	Logical	Work	Uncaptured	Capture Ratio
HMC/LPAR	.5-1%	.5-1%	98%+	0%	100%
z/VM	1-2%	1-2%	95%+	0%	99%
z/VSE	<.1%	6-8%	92-94%	0%	99.9%
Linux	2%	7-10%	90%	< 1%	99% +
z/OS	????	3-13%	85-92%	8-15%?	85-92%
CICS		3%	96-97%	< 1%	99+%
DB2		16%	70%+		70%

Customers complain about SMT

Chargeback numbers are over charging

- Capacity Planning – too many engines

Non-SMT, numbers are correct

customer "correctly" complain that chargeback is broken

Capacity Planning Thoughts

z/VM: One core, Two threads (14 cores, 28 threads)

- "assigned" 933.7% - 4.1%
- **Two threads not always both active -> thread idle time**
  - **Source: SYTCUP/HMC**
- 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 ←

Goal: Account for 933.7% of IFLs

VELOCITY SOFTWARE

SMT Accounting is wrong

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

Workload helped by SMT? Is Monitor user data valid?

- 1535 percent “thread time” (validated against cpu busy)
- 1192 percent core time
- “would be” time 1051,
- Used 1192 percent, could have been 1051. (Both wrong)

HMC / hardware says:

- 933% assigned, thread idle 138%
- (933*2 – 138) /2 = 864 actually consumed
- “MT Prorated” is not a useful number

CPU Analysis for Openshift

IBM "openshift business decision" - 3 engines for "free"

- Based on their data?
- CPU with SMT really is lower
- .

```
Report: ESAUSP5        User SMT CPU Consumption Analys
-------------------------------------------------------
           <------CPU Percent Consumed    (Total)---->
UserID     <Traditional> <MT-Equivalent> <IBM Prorate>
/Class     Total  Virt    Total  Virtual   Total Virtual
-------- ------ -----    -----  -------   ----- -------
07:02:00 414.9   408.0   322.7    317.3   239.7   235.8

  ***User Class Analysis***
OpenShif 355.0   350.3   276.0    272.3   204.9   202.2

  ***Top User Analysis***
RHOSCP1  142.4   140.8   110.1    108.9   82.93   82.01
RHOSCP3  125.2   123.8   97.38    96.34   72.35   71.60
RHOSCP2  86.79   85.04   68.00    66.64   49.31   48.30
```

CPU by Component/Function

Some even "better news"

- CPU numbers are traditional, measured by Linux
- VSI Prorated based on HMC / "Hardware" data

```
Report: ESAUSP5       User SMT CPU Consumption Analysis
Monitor initialized: 03/08/23 at 07:00:01 on 8562 serial 040F78
-------------------------------------------------------------------
         <------CPU Percent Consumed   (Total)---->  <-TOTAL CPU-->
UserID   <Traditional> <MT-Equivalent> <IBM Prorate> <VSI Prorated>
/Class   Total  Virt   Total  Virtual  Total Virtual Total  Virtual
-------- ----- -----   -----  -------  ----- ------- ----- --------
07:02:00 414.9  408.0  322.7    317.3  239.7   235.8 208.2    204.7
 ***User Class Analysis***
OpenShif 355.0  350.3  276.0    272.3  204.9   202.2 178.1    175.7
 ***Top User Analysis***
RHOSCP1  142.4  140.8  110.1    108.9  82.93   82.01 71.43    70.65
RHOSCP3  125.2  123.8  97.38    96.34  72.35   71.60 62.80    62.14
RHOSCP2  86.79  85.04  68.00    66.64  49.31   48.30 43.55    42.67
```

Recap and Thankyou

Capture Ratios validate the data for

- Capacity Planning – know consumption by app
- Chargeback – who consumed exactly what?
- Performance analysis – who is using cpu now?

Corrected SMT data available in zVPS

- (SMT performs much better than I thought)

Thank you for your time!!

Questions and suggestions can be sent to
'barton@velocitysoftware.com'