Case Study 5

After Migrating CECs - Performance Issue

Case Study Summary

Velocity Software solves performance problems.

- As a valued customer, we want to pass this knowledge on to you.
- The following is a case study of a solved real-life performance issue.
- This case study will show:
  - The problem as reported by users
  - The problem observations
  - What was found in the Velocity Software data
  - What was suggested to the customer
  - If provided, follow up from the customer

The Problem:

After migrating from two separate LPARs on two different CECs to one LPAR on one CEC, certain guests were experiencing performance issues.

Problem Observations:

- TEST1 and TEST2 were experiencing slow response times.

What the Data Showed (Configuration data)

ESAUSRC – User Configuration shows:

- TEST1 had a Relative SHARE of 1602 with 4 vCPUs
- TEST2 had a Relative SHARE of 99 with 4 vCPUs

```
Report: ESAUSRC      User Configuration                              Velocity Sof
-------------------------------------------------------------------------------
                                                  <-------SHARE-------> <---CPU
                  Account  ACI Grp  <CP POOL> CPU <Normal> <--MAX-> Lim <Count>
UserID   ClassID  Code     Name     PoolName Type  Rel Abs Typ Shre -it Def On
-------- -------- -------- -------- -------- ---- ---- --- --- ---- --- --- --
TEST1    xxxxxxx  xxxxx    TPF            .   CP 1602   .   .    .   .   4  4
TEST2    xxxxxxx  xxxxx    TPF            .   CP   99   .   .    .   .   4  4
```

Relative SHARE is divided between active vCPUs. For each server, each vCPU only got a fourth of what was defined. TEST1 received a SHARE of 400 (basically the normal default) and TEST2 received a SHARE of 25.

See **Setting CPU SHARES** for mor information about SHARE settings.

What the Data Showed (Utilization data)

ESALPAR – Logical Partition Analysis showed:

- LPAR1 has 4 CPs with a weight of 920

```
Report: ESALPARS     Logical Partition Summary                    Velocity Software Corporate
-------------------------------------------------------------------------------------------------------------------------------
          <--Complex--> <--------Logical Partition-------> <-Assigned Shares----->                    <LPAR Capping>  Entitl
          Phys Dispatch              Virt CPU  <%Assigned> <---LPAR--> <VCPU Pct>  Wait <-Thread->   On/ Capping    CPU Cn
Time      CPUs    Slice Name     Nbr CPUs Type Total  Ovhd  Weight  Pct /SYS /CPU   Comp Idle   cnt   Off  Value
--------  ---- -------- -------- --- ---- ---- -----  ----  ------ ---- ---- ----   ---- ------ ---   ---- --------  ------
05/09/24

13:01:00     5  Dynamic Totals:   00    5  CP  377.3   0.4     930  100
                     Totals:   00    1 IFL   22.6   0.1     999  100
                     LPAR1     01    4  CP  376.9   0.4     920 98.9 24.7    99    No  376.3    2   No        .    3.96
                     LPAR1     01    1 IFL   22.6   0.1     999  100  100   100    No  17.06    2   No        .    1.00
                     LPAR2     0F    1  CP    0.5   0.0      10  1.1 1.07 4.30    No      0    1   No        .    0.04
                     LPAR3     0A    0  CP      0     0      35    0    0    0    No      0    1   No        .       0
                     LPAR4     09    0  CP      0     0      35    0    0    0    No      0    1   No        .       0

Totals by Processor type:
<---------CPU-------> <-Shared Processor busy->
Type Count Ded shared  Total  Logical Ovhd Mgmt
---- ----- --- ------ ------ -------- ---- ----
CP       4   0      4  377.6    376.9  0.4  0.3
IFL      1   0      1   22.7     22.5  0.1  0.1
```

Often through the time period, the LPAR is over 90% total utilization.

What the Data Showed (Utilization data)

ESAXACT – Transaction Delay Analysis showed:

- Both users are waiting on CPU

UserID /Class	<-Samples-> Total	In Q	<-----Percent non-dormant (Wait states)- Run	Sim	CPU	SIO	Pag	E- SVM	D- SVM	T- SVM	CF	Tst Idl
TEST1	240	240	7.5	39	44	1.3	0	0	0	0	2.9	4.2
TEST1	240	240	13	3.8	71	0	0	0	0	0	3.8	8.3
TEST1	240	240	6.7	10	64	0.4	0	0	0	0	4.6	13
TEST1	240	240	5.4	6.7	69	0	0	0	0	0	3.8	15
TEST1	240	240	8.8	8.3	68	0.8	0	0	0	0	5.0	9.2
TEST1	240	240	5.8	8.8	71	0.8	0	0	0	0	3.8	8.8
TEST1	240	240	8.3	5.8	72	0	0	0	0	0	3.3	10
TEST1	240	240	7.1	5.4	67	1.3	0	0	0	0	6.7	12
TEST2	240	240	2.1	25	54	0.4	0	0	0	0	18	0.8
TEST2	240	240	0	0.8	98	0	0	0	0	0	0.8	0.8
TEST2	240	240	0.4	2.5	95	0	0	0	0	0	0	2.1
TEST2	240	240	0.4	3.8	89	0	0	0	0	0	4.6	1.7
TEST2	240	240	0	1.7	93	0	0	0	0	0	2.1	2.9
TEST2	240	240	0.8	1.7	94	0	0	0	0	0	2.9	0.4
TEST2	240	240	0.4	2.9	90	0	0	0	0	0	5.8	0.8
TEST2	240	240	0	2.1	94	0	0	0	0	0	1.7	2.5

The ESAXACT data/report is a great way to see what resources are holding up system activity.

What the Data Showed (Utilization data)

ESACPUA – CPU Utilization Analysis showed:

- Even when CPU utilization went down, the User Overhead was high and the SIGP rate rose
- A high User Overhead can mean there are virtual machines with too many vCPUs

```
Report: ESACPUA      CPU Utilization Analysis                    Velocity Software Corporate
-------------------------------------------------------------------------------------------------------------------
         <----Load---->    <CPU percents><--Internal (per second)--> SIGP <-Spin Locks--><UsrExits>
         <-Usrs--> Tran     Totl Ovrhead Diag Inst    SIE Fast  Page Rate Proc  ms/  rate Rate ms/
Time     Actv In Q /sec CPU Util Usr Sys nose  Sim intrcp path fault /sec  Pct spin /sec  /sec Exit
-------- ---- ---- ---- --- ---- --- --- ---- ---- ------ ---- ----- ---- ---- ---- ----- ---- ----
13:01:00  103 50.0  3.7   0  93.9  33 2.2 3556  37K  59232 0.28   0.5  766 0.03 0.00 134.7  151    .
                        2  94.8  35 1.5 2190  37K  61397 0.52   0.3  683 0.02 0.00 105.6    0    .
                        4  93.8  33 1.5 1620  37K  61432 0.33   0.1  961 0.03 0.00 156.5    0    .
                        6  93.2  33 2.5 1552  39K  63620    0   0.0  905 0.04 0.00 164.3    0    .
                        8  13.2 0.3 0.6 4130  436  910.3 7.13   1.4  821 0.01 0.00 50.95    0    .
                        9  14.1 0.2 0.4 2044  282  629.9 10.0     0  395 0.00 0.00 38.27    0    .
                           ---- --- --- ---- ---- ------ ---- ----- ---- ---- ---- ----- ---- ----

System:                    403 134 8.7   15K 151K 247221 18.2   2.2 4532 0.14 0.00 650.4  151 0.00
 General purpose CPUS:     376 134 7.8  8918 151K 245681 1.13   0.8 3316 0.13 0.00 561.2  151    .
 IFL CPUs:                27.2 0.4 1.0  6174  718 1540.2 17.1   1.4 1217 0.01 0.00 89.22    0    .
-------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------
13:11:00   96 52.0  3.3   0  66.1  29 3.6   685  29K  48848 12.0   0.0 6524 0.19 0.00   995  174    .
                        2  67.8  31 2.7   302  31K  52149 0.93     0 7473 0.19 0.00 874.9    0    .
                        4  65.5  29 2.5   457  29K  49442 0.47     0 8008 0.19 0.00 884.9    0    .
                        6  69.8  31 3.0   384  33K  54768 0.27     0 7669 0.24 0.00   953    0    .
                        8   6.4 0.2 0.4  1788  292  623.7 7.47   0.6  455 0.01 0.00 57.77    0    .
                        9   6.9 0.1 0.3  3813  248  532.5 8.47     0  306 0.01 0.00 50.03    0    .
                           ---- --- --- ---- ---- ------ ---- ----- ---- ---- ---- ----- ---- ----

System:                    282 121  13 7430 123K 206363 29.5   0.6  30K 0.83 0.00  3816  174 0.00
 General purpose CPUS:     269 121  12 1828 123K 205207 13.6   0.0  30K 0.81 0.00  3708  174    .
 IFL CPUs:                13.3 0.3 0.7 5602  541 1156.3 15.9   0.6  761 0.01 0.00 107.8    0    .
```

What the Data Showed (Utilization data)

ESAUSP2 – User Percentage showed:

- The two users are often the top users of CPU by a large margin

```
Report: ESAUSP2      User Resource Rate Report

-------------------------------------------------------------
         <---CPU time---> <----Main Storage (pages)----->
UserID   <(Percent)> T:V -<Resident> Lock <-----WSS----->
/Class   Total  Virt  Rat Totl Activ  -ed Totl Activ  Avg
-------- ----- ----- ---- ---- ----- ---- ---- ----- ----
 ***Top User Analysis***
TEST1    139.8 89.40 1.56 1.1M 1147K 4083 1.1M 1143K   1M
TEST2    102.8 64.55 1.59 889K  889K 4082 885K  885K 885K
.        12.16  4.56 2.66 363K  363K 2039 361K  361K 361K
.        11.89  4.22 2.82 977K  977K 2037 975K  975K 975K
.        11.41  4.09 2.79 1.0M 1047K 2037 1.0M 1045K   1M
```

Performance Enhancement Suggestions:

1 – Change the REL SHARE of both machines to better match the need.

- Setting the REL SHARE for each of the two machines doesn't seem to help. One is basically set to the default and the other is set very low for the number of vCPUs.
  - Setting the SHARE too high or too low can affect the performance of the machine and of the system.

Performance Enhancement Suggestions:

2 – Lower the vCPU count for both machines from 4 to 2.

- Both machines are waiting on CPU more than running. Neither machine is using more than 1.5 vCPUs
  - Having too many vCPUs, especially with a lower SHARE, causes overhead and delays.

Performance Enhancement Suggestions:

3 – There is excess capacity not being utilized because of Affinity processing. Use the SYSCONTROL command to change from 1 (default) to 0.

This should drop CPU wait at high CPU utilization times.

```
q syscontrol
DISPATCH THDAFFINITY        ON
DISPATCH PREEMPTLOCAL      OFF
DISPATCH TSEARLY            50
DISPATCH INCHIPBUSY      50000
DISPATCH INCHIPDELAY     50000
DISPATCH INNODEBUSY     100000
DISPATCH INNODEDELAY    100000
DISPATCH INSYSBUSY      200000
DISPATCH INSYSDELAY     200000

SET SYSCONTROL DISPATCH MODLEVEL 0

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
```

Performance Enhancement Suggestions:

4 – Enable Main Frame Cache (MFC) data to show how much work is actually getting done. This is done via the HMC and has no performance impact.

Use Cycles Per Instruction (CPI) to trend how much work is actually being done. When CPI drops, more work is being done.

Instructions for turning on MFC data

Understanding MFC data

Performance Enhancement Suggestions:

5 – Adding a fifth CP to the LPAR will make a positive difference. The four CP's are running at high utilization the majority of the time.

Also, the LPAR is set up to run mostly on CP's, not IFL's. Changing to IFL's will also make a positive difference.

Customer Feedback

No actual feedback was given, however provided data showed:

- The SYSCONTROL command was done and showed improvement.
- MFC data was enabled and showed that cycles per instruction improved.