Case Study 6

Maximizing CPU Utilization

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:

We are experiencing high utilization but we need more, without upgrading

Problem Observations:

- The system is showing 97% utilization

What the Data Showed (Utilization data)

ESALPARS – Logical Partition Summary shows:

- There are 40 IFLs that are shared
- IFLs are 97% assigned (3867/4000)
- The system overhead “Ovhd” – 4.9 and “Mgmt” – 4.1 aren’t bad
- CPU delays can be expected. The shares/weights need to be managed

Report: **ESALPARS** **Logical Partition Summary**

Totals by Processor type:

<----------CPU------->				<-Shared Processor busy->			
Type	Count	Ded	shared	Total	Logical	Ovhd	Mgmt
CP	8	0	8	711.0	708.8	1.4	0.7
IFL	40	0	40	3867.5	3858.5	4.9	4.1
ICF	1	0	1	24.2	19.6	0.4	4.2
ZIIP	4	0	4	268.3	266.4	0.6	1.2

97% busy

What the Data Showed (Utilization data)

Report: **ESAXACT** Transaction Delay Analysis

UserID /Class	<-Samples-> Total	In Q	<-----Percent non-dormant (Wait sta Run	Sim	**CPU**	SIO	Pag	D- SVM	T- SVM	CF	Tst Idl
15:01:00	117	137	13	0	15	0	0	0	0	0	72
Hi-Freq:	20280	8108	10	0.1	**11**	0	0.0	0.5	0.0	0	78
User Class Analysis											
P_AP_JB	6300	2990	12	0.1	13	0	0	0	0	0	75
PROD_MQ	180	119	0.8	0	1.7	0	0	0	0	0	97
P_OTHER	1320	299	5.4	0	9.4	0	0	0	0	0	85
P_SUB_JB	3720	1519	6.5	0.2	8.4	0	0	0	0	0	85
P_UTIL	360	239	14	0	16	0	0	0	0	0	70
PROD_WAS	2640	1207	9.0	0.1	15	0	0.2	0	0	0	76
RT_AP_JB	1680	769	11	0	12	0	0	0	0	0	77
RT_OTHER	120	59	10	0	12	0	0	0	0	0	78
RT_SB_JB	240	59	19	0	8.5	0	0	0	0	0	73
RTST_WAS	1320	662	15	0	6.2	0	0	0	0	0	79
zVPS	720	55	15	0	0	0	0	0.1	0	0	85
Other	1680	131	6.1	0	3.8	0	0	6.5	0.8	0	89
CPU POOL User Analysis											
MQPOOL	180	119	0.8	0	1.7	0	0	0	0	0	97

ESAXACT – Transaction Delay Analysis showed on PROD2:

- Running – machines are running
- CPU Wait – is there but not that high

What the Data Showed (Utilization data)

Report: **ESAXACT** Transaction Delay Analysis

UserID /Class	<-Samples-> Total	In Q	<-----Percent non-dormant wait Run	Sim	**CPU**	SIO	Pag	E- SVM	D- SVM
03/16/26 15:01:00	186	302	5.6	0	43	0	0	0	0
Hi-Freq:	25740	18004	6.3	0.1	**35**	0.0	0.1	1.5	0
User Class Analysis									
D_AP_JB	2040	1995	6.4	0.1	30	0	0.4	0	0
DEV_MQ	120	119	3.4	0	29	0	0	0	0
D_QM_OTH	1080	1058	4.4	0	28	0	0.1	0	0
D_SUB_JB	600	591	8.8	0.2	44	0	0.7	0	0
D_UTIL	600	369	4.1	0	20	0	0.5	0	0
D_WAS	720	715	8.4	0	53	0	0	0	0
GT_AP_JB	1800	1416	4.7	0.1	24	0	0.1	0	0
GT_OTHER	360	353	4.5	0	16	0	0	0	0
GT_SB_JB	1320	376	10	0	43	0	0.5	0	0
GTST_WAS	1320	1176	3.7	0	25	0	0	0	0
QM_AP_JB	2280	2245	8.9	0.1	39	0	0.2	0	0
QM_MQ	120	120	5.8	0	58	0	0	0	0
QM_SB_JB	960	837	10	0	60	0	0	0	0
QM_WAS	600	593	5.6	0	41	0	0	0	0
STAGE_MQ	360	357	11	0.3	34	0	0	0	0

ESAXACT – Transaction Delay Analysis showed on DEV2:

- CPU Wait – is much higher

Why is CPU Wait high?

- CPU is limited by the LPAR weight
- There is high CPU Utilization on the box
- Affinity lasts for a time slice
- The time slice is the default of 10ms

What the Data Showed (Configuration/Utilization data)

ESALPARS – Logical Partition Summary:

- Shows the z/VM share of the IFLs (always start here)

Report: **ESALPARS** Logical Partition Summary

	<--Complex-->		<--------Logical Partition------->			<%Assigned>		<-Assigned <---LPAR-->	
Time	Phys CPUs	Dispatch Slice	Name	Nbr	Virt CPUs	CPU Type	Total	Ovhd	Weight	Pct
15:01:00	53	Dynamic	Totals:	00	14	CP	710.2	1.4	966	100
			Totals:	00	46	IFL	3863	4.9	436	100
			VMPROD2	06	**13**	IFL	945.3	2.8	128	**29.4**
			VMDEV1	07	10	IFL	928.4	0.2	90	20.6
			VMDEV2	08	10	IFL	929.7	0.2	90	20.6
			VMPROD1	05	13	IFL	1060	1.7	128	29.4

Totals by Processor type:

<---------CPU------->				<-Shared Processor busy->			
Type	Count	Ded	shared	Total	Logical	Ovhd	Mgmt
IFL	**40**	**0**	**40**	**3867.5**	**3858.5**	**4.9**	**4.1**

- VMPROD2 entitlement: 128/436 or 12% of 40 shared IFLs
- VMPROD2 entitlement is equal to 11.7 IFLs
- VMPROD1 entitlement is the same but only **using** 10.6
- VMDEV1/2 are using 930% shared IFLs (more than their 8.3 entitlement)
- IFLs are running 97% busy
- Not much left…

What the Data Showed (Configuration/Utilization data)

ESALPARS – Logical Partition Summary:

<--------Logical Partition------->				<%Assigned>		<-Assigned <---LPAR-->		<-Thread->		Entitled CPU
Name	Nbr	Virt CPUs	CPU Type	Total	Ovhd	Weight	Pct	Idle	cnt	Cnt
Totals:	00	14	CP	710.2	1.4	966	100			
Totals:	00	46	IFL	3863	4.9	436	100			
VMPROD1	05	13	IFL	**1060**	1.7	128	29.4	**356.0**	2	**11.74**
VMPROD2	06	13	IFL	945.7	2.8	128	29.4	**437.7**	2	11.74
VMDEV1	07	10	IFL	928.4	0.2	90	20.6	0.00	2	8.26
VMDEV2	08	10	IFL	929.3	0.2	90	20.6	0.03	2	8.26

- VMPROD1/2 are not using their entitled but Thread Idle time (356/437)
- VMDEV1/2 are using all of their entitled time – there is no Thread Idle time
- VMPROD1/2 have EXTRA CAPACITY available (Thread idle = unused capacity)
- VMPROD1/2 have CPU Wait time equivalent to their run time

Performance Enhancement Suggestions:

1 – Use the SET SYSCONTROL command to turn off Affinity

- CP SET SYSCONTROL DISPATCH MODLEVEL 0
  - Turning off Affinity helps reduce Thread Idle time and thus gives back capacity
- CP SET SYSCONTROL DISPATCH STEALBARRIER INCHIPBUSY 0 INCHIPDELAY 0 INNODEBUSY 0 INNODEDELAY 0 INSYSBUSY 0 INSYSDELAY 0
  - Large LPARS that cross chips (within nodes) may also want to turn off node delay

2 – Use the SET SRM command to set the Dispatch Time Slice to 1

- CP SET SRM DSPSlice 1
  - The default Dispatch Time Slice is either 5ms (non-SMT) or 10ms (SMT). Setting it to 1 for Linux environments usually improves performance by giving back the Thread Idle capacity