Performance Tuning and TCO with CA Datacom

CTC 25 – DBA – Friday, April 29 2016

Kevin Shuma

Abstract

Recap of many of the concepts from throughout the week. Bringing them together to give you the tools to take these concepts and understand how they can work to reduce your ownership costs and improve performance.

Agenda

- I tuned last year why do it again?
- What drives TCO?
- How to measure TCO?
- Techniques to improve TCO
- Version 15.0 Techniques
- Summary

I Tuned Last Year Why Do It Again?

- ## Hardware changes

  IBM z13 - The Processor Unit chip (PU chip) measures 678 mm^2 and consists of 3.99 billion transistors fabricated using IBM's 22 nm CMOS silicon on insulator fabrication process, using 17 metal layers and supporting speeds of 5.2 GHz, which is less than its predecessor, the zEC12.[3][5] The PU chip can have six, seven or eight cores (or "processor units" in IBM's parlance) enabled depending on configuration. New for the z13 is that the PU chip comes packaged in single chip modules, which is a big change from all previous mainframe processors that were mounted on large multi-chip modules. A computer drawer consists of six PU chips and two Storage Controller (SC) chips.[3]

  = = it's really fast!

- ## Operating System changes

  IBM's z/OS® V2.2 operating system and latest IBM® z13 server deliver innovations designed to help you build the highly scalable next-generation infrastructure you need. Together, they offer the capacity, scale, availability, and throughput required to improve business performance, meet response time objectives, protect sensitive data and transactions, and minimize operational risk for an exceptional customer experience. New economic efficiencies allow the z13 with z/OS V2.2 to offer more throughput and capabilities with less impact to the IT budget.

  = = it's really complex!

- ## Database Software changes

  CA Datacom V14.02 and v15.0 delivered over 50 major enhancements that most likely will directly affect how you database applications run.

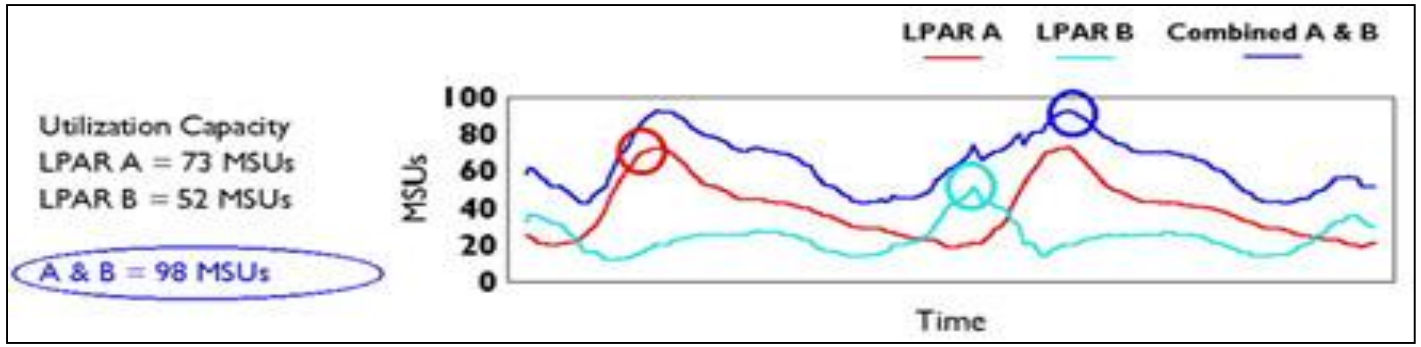
  = = it's really different!

That Sounds Really Scary!!

- It is, and yet it isn't
- Understanding what drives your business and how that business relates to the database applications helps to establish the total cost of ownership (TCO)
- TCO can be different for every site
  - But in most cases, mainframe sites share a set of typically TCO drivers

What Drives TCO?

- Hardware and software charges based on CPU consumption
  - Typically based on peak CPU (MSUs) use during a measured period
    - IBM specialty (zIIP) processors not included
  - IBM Workload License Charges are based on 4 hour averages



  - While reducing CPU in general is good, focusing on the peak periods is better

What Drives TCO? (cont'd)

- Memory typically measured as a "overhead cost"
  - Cost typically spread across all tasks
  - Can be used to reduce IO, and therefore reduce CPU
  - If overused, can drive up CPU in the form of system paging
- Physical IO typically not measured as a "cost"
  - But it consumes CPU for each IO used
  - Not available for dispatching on specialty processor
- DASD Storage
  - Physical costs for DASD "on the floor"
  - Relatively low-cost resource, but
    - The amount of data is growing
    - Most shops are seeing an ever-increasing DASD footprint

How to Measure TCO?

- Business Value Metrics - CPU
  - Requests per CPU

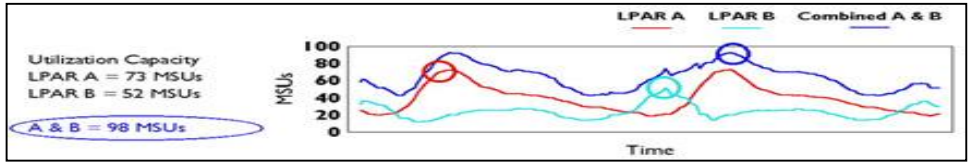
    650,000rq / (600 CP CPU) = **1083rq per total CPU**

  - Adjusted value with zIIP CPU at a % of CP CPU
    - Business rates zIIP at a cost of 10% of CP

  - Requests / (CP CPU + (zIIP CPU * .1))

    650,000rq / (50 + (550 * .1))

    650,000rq / (105) = **6190rq per adjusted CPU**



Remember this graph?
- It is not just CPU but CPU use at the "high periods"
- zIIP does not count towards these periods

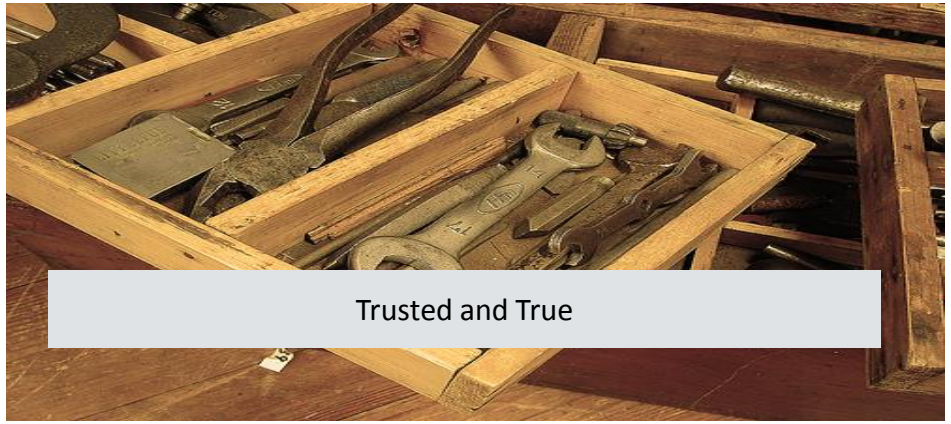
How to Measure TCO? (cont'd)

- Business Value Metrics - IO
  - Requests per IO

    650,000rq / (60,000 IOs) = **10.83rq per IO**

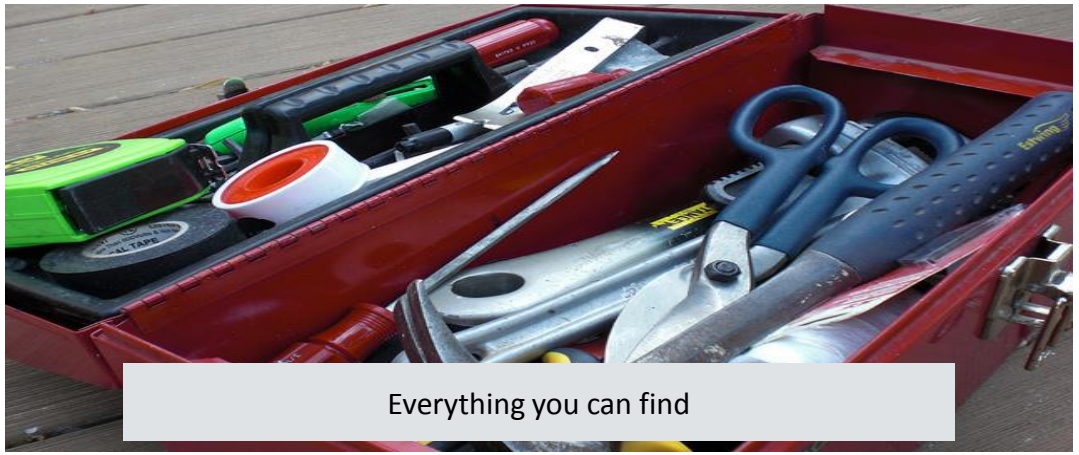
- (?) Business Value Metrics – DASD in use
  - Request per MB, TB, other?
    - Not something regularly measured
    - Should it be?

What's in Your Toolbox?



Trusted and True

New and shiny



Everything you can find

Which is the best choice?

What's in Your Toolbox? (cont'd)

- It doesn't matter as long as it meets your goals?
- Improved performance of CA Datacom
  - Which should result in reduced resource consumption (per request)
  - Which should result in a reduction in the Total Cost of Ownership

Techniques to Improve TCO

- Gain an understanding resource consumption
  - What resources are being consumed
  - What are the performance metrics
  - What are the “hurdles”
    - CPU consumption
    - Resource utilization
    - Response time
    - Locking
    - Logging/recovery
    - DASD IOs
    - More …

Knowing What the Environment is Doing
The Performance Snapshot

- Resource consumption since MUF came up
- General indication of “health”
- Indication of high-use or trouble spots
- Performance benchmark or baseline
  - Always take snapshot at the “same” period
  - When problems occur:
    - Do a quick scan of snapshot for trouble spots
    - Compare current snapshot against “healthy snapshot”
      - Look for differences

“One-time” Performance Snapshot – Various Options

- DBUTLTY AUTOINFO (AutoInfo)
  - Printed report
  - Option for sequential output ready to upload into spreadsheet
- Console command ALL_INFO_REPORT
  - Generates output to PXX or PXX SYSOUT datasets
- MUF EOJ SYSPRINT
  - Printed to SYSPRINT at MUF end of job
- PXX SYSSTAT=SUMMARY
  - On demand requires PXXSTATS=DETAIL
  - PXXSTATS=EOJ or EOJPRT only available at MUF EOJ

“One-time” Performance Snapshot – Comparison

MUF EOJ

MUF UP

Report type	Description	AUTOINFO	ALL_INFO_REPORT	PXX SUMMARY	MUF EOJ SYSPRINT
MUF System Stats	Summary of MUF statistics	Yes	Yes	Yes	Yes (w/EOJPRT)
MUF Area Statistics	Summary of IXX and Data area usage	Yes	Yes	Yes	Yes (w/EOJPRT)
MUF Table Statistics	Summary of logical table use	Yes	Yes	Yes	Yes (w/EOJPRT)
MUF Identification	Execution information	Yes			N/A
MUF Options	Current options set	Yes			ECHO of MUF MSGs
MRDF	MRDF datasets and usage	Yes	Yes		Yes
Acct Table Usage	Statistics on the accounting tables usage	Yes			
Internal stats	Key MUF internal statistics	Summary	Yes		Yes
SMP Stats	Statictics on SMP task usage	Summary	Summary		Yes
Return codes	Counts on return codes	Yes			
SQL Codes	Counts of SQL Codes	Yes	Yes		Yes
SQL Misc Statistics	Statistics on SQL Procedure use	Yes			
CBS Usage statics	Summary of CBS statistics	Expanded	Yes, Subset	Yes, subset	Yes, subset
MUF Configuration	Configuration parameters (DIAG Options, etc.	Yes			
zIIP Usage	Statistics on zIIP availability, usage, etc.	Yes	Yes		Yes
TCB and SRB	Statistics on MUF TCB and SRB usage	Yes	Yes		Yes
BVM report	Business Value Metrics report	Yes			
BVD report	Business Value Details report	Yes			
XCF usage summary	Summary of cross LPAR communication		Yes		Yes
Memory Use Report	Report on memory use - for L2 debugging		Yes		Yes
Log status	Status of the log file		Yes		Yes
Buffer pool	Buffer pool sizes and usage	Yes, in MUF system	Yes	Yes, in MUF system	Yes
Coupler Facility	List of coupler facility structures		Yes		Yes
Current task	List of current executing tasks		Yes		

Automated Performance Snapshots and Deltas

- DBUTLTY AUTOCOLL (AutoCollect)
  - Automated snapshot collection
    - Similar data as AutoInfo
    - Data placed in a set of Datacom tables (DBID 1019)
  - Automated creation of “deltas”
    - Data placed in a set of Datacom tables (DBID 1020)
    - Interval - Difference between snapshots
    - Last – Difference between 1st snapshot and last snapshot
  - On-demand user created deltas
    - Summary – combining deltas together
    - Baseline – combine deltas together and divide to get an average baseline
    - Average performance – combine deltas and calculate average “hour” of performance

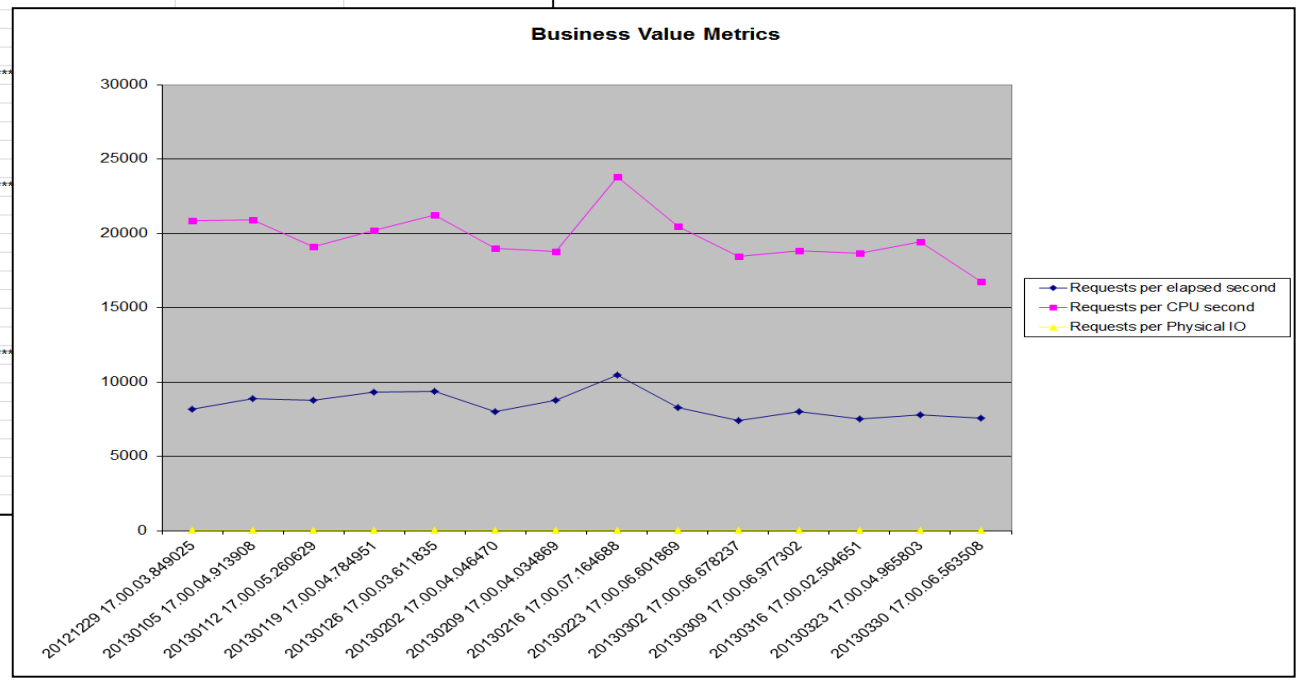
Samples from AutoCollect could be used to map to peak periods

MUF Enabled	20121215 18.38.18.000000	20121222 18.54.55.000000	20121229 20.26.25.000000
MUF Period Begin	20121221 17.00.15.133829	20121228 17.00.16.004840	20130104 17.00.08.700971
MUF Period End	20121229 17.00.03.849025	20130105 17.00.04.913908	20130112 17.00.05.260629
MUF Type	SUMMARY	SUMMARY	SUMMARY
Duration Hours	190.0834	188.5573	190.3197
MUF Name	DATACOMP	DATACOMP	DATACOMP
User Tag			
**********	**********	**********	**********
Elapsed Seconds			
CPU Seconds Used			
Physical I/O			
Database Requests			
Data Area Requests			
Table Requests			
Table Reads			
Table Adds			
Table Deletes			
Table Updates			
**********	**********		
Calculated Business Value Metrics			
Requests per elapsed second			
Requests per CPU second			
Requests per Physical IO			

Business Value Metrics

MUF Enabled	20121215 18.38.18.000000		20121222 18.54.55.000000		20121229 20.26.25.000000
MUF Period Begin	20121221 17.00.15.133829		20121228 17.00.16.004840		20130104 17.00.08.700971
MUF Period End	20121229 17.00.03.849025		20130105 17.00.04.913908		20130112 17.00.05.260629
MUF Type	SUMMARY		SUMMARY		SUMMARY
Duration Hours	190.0834		188.5573		190.3197
MUF Name	DATACOMP		DATACOMP		DATACOMP
User Tag					
**********	**********	**********	**********	**********	**********
Elapsed Seconds	684300				
CPU Seconds Used	268311				
**********	**********	**********	**********		
Request Processing		%			
Database Requests	5590228981	**			
Data Manager Requests	6424590147	114.9			
SQL Requests	183165670	3.2			
**********	**********	**********	**********		
Physical I/O Processing		%			
Total Physical I/O	879154984	**			
Physical I/O - Read	729818746	83			
Physical I/O - Write	149336238	16.9			
Sequential Read Ahead I/Os	25867140	2.9			
Accounting I/O	2359	0			
Logging I/O	110299617	12.5			
**********	**********	**********	**********		
Buffer Processing		5 to 1 Ratios			
IXX Buffer Use 5+	10430556032	301.1			
DXX Buffer Use 5+	18119268991	54.2			
DATA Buffer Use 5+	2914095744	6.8			
DATA2 Buffer Use 5+	3305262766	9.5			
No IXX Buffer Available	120				
No DXX Buffer Available	6824				

Business Value Details



Graphs and Charts

Knowing What the Environment is Doing
Real-time Performance Monitors – CA SYSVIEW

```
SYSVIEW ISPF2 CA31 ---------------------------------- MENU, CA Datacom Menu ----------------------------
Option ===>
------------------------------------------------------------------------------------------------------
Jobname A31ADMUF  ASID 007C  Jobid JOB24246  Datacom 14.0
------------------------------------------------------------------------------------------------------
Option Command  Parameters                    Description
_    1 DCLIST                                 List of Datacom jobs
_    2 DCAREAS                                Directory areas
_    3 DCDBASES                               Directory databases
_    4 DCMUFS                                 MUF identity
_    5 DCTASKS
_    6 MENU     DCDIR
_    7 MENU     DCMON
_    8 MENU     DCCDC
_    9 MENU     DCDBASE
_   10 MENU     DCDEBUG
_   11 MENU     DCOPT
_   12 MENU     DCSQL
_   13 MENU     CSF
```

```
SYSVIEW ISPF2 CA31 ------------------------- DCLIST, CA Datacom System Activity --------------------------
04/15/13 12:05:39
Command ===>
Scroll ===> PAGE
------------------------------------------------------------------------------------------ Lvl 3 Row 1-
18/37 Col 1-131/203
Formats DEFAULT AD DB
Jobname A31ADMUF  ASID 007C  Jobid JOB24246  Datacom 14.0
------------------------------------------------------------------------------------------
Cmd      Name      DCType JobStat  Job-CPU RealStg IOReqs Clocktime  Jobnr Stepname Procname Type JobClass ASID
ASCB      Release SV
_______ A31ADMUF AD      NS       0.698085   38.8M    998  00:18:59  24246 $$$$$$@           JOB  D        007C
00FAD900     14.0
_______ BAR$MUF  AD      NS       3.552073     72M   4351  02:59:45   9874 MUFSTRT           JOB  A        02C3
00EFA400     12.0 23
_______ CHORMUFO AD      NS       00:10:33   93.1M 1.98m  37:04:20  28008 CHORMUFO $$$$$$@  STC  $        0134
00F60780     14.0
_______ CHORMUFP AD      NS       18.28961   71.2M  46249
00F60600     14.0
_______ CHRC3MUF AD      NS       1.277210   43.8M   4290
00F32400     14.0
_______ CHRE3MUF AD
14.0
_______ CHRS3MUF AD      NS       0.869794   33.2M   1261
00F3C680     14.0
_______ CHRT3MUF AD      NS       00:14:07   87.7M 2.74m
00F61180     14.0
_______ CHRZ3MUF AD      NS       00:05:41   91.8M 886784
00F30E00     14.0
_______ CHSQAMUF AD      NS       0.603661     37M   1214
00F0B700     14.0
_______ CH25MUFF AD      NS       00:09:00    113M 1.25m
00F61000     12.0 23
_______ CH30MUFF AD      NS       00:05:42   87.3M 1.34m
00F60E80     14.0
_______ CH31MUF3 AD      NS       19.27286   69.3M  19849
00EF9400     14.0
_______ CMGRMUF  AD      NS       00:01:07   80.6M  45999
00F63E80     14.0
_______ DBSRM8   DB      NS       0.815483   37.5M   1559
00F0B580     14.0
_______ DSNZMFK1 DB      NS       0.390854   64.7M   1936
00F35580     12.0 24
```

```
SYSVIEW ISPF2 CA31 ------------------------- DCSTATS, CA Datacom MUF nternal
Stats -------
Command ====>
------------------------------------------------------------------------------
-----------
Jobname CHRZ3MUF  ASID 020E  Jobid STC44941  Datacom 14.0
Breaks     529      Short I/O Memory      0
------------------------------------------------------------------------------
-----------
Cmd Type          Count Pct% ...10...20...30...40...50...60...70...80...90..100
___ Read 1            1   3%
___ Read 2-8         36  97%
___ Read 9-32         0   0%
___ Read 33+          0   0%
___ Write 1       52641  55%
___ Write 2-12    34057  36%
___ Write 13-24    3071   3%
___ Write 25+      5646   6%
********************************************************************** End of Data
********************
```

How to Improve TCO With Datacom – Reduce CPU

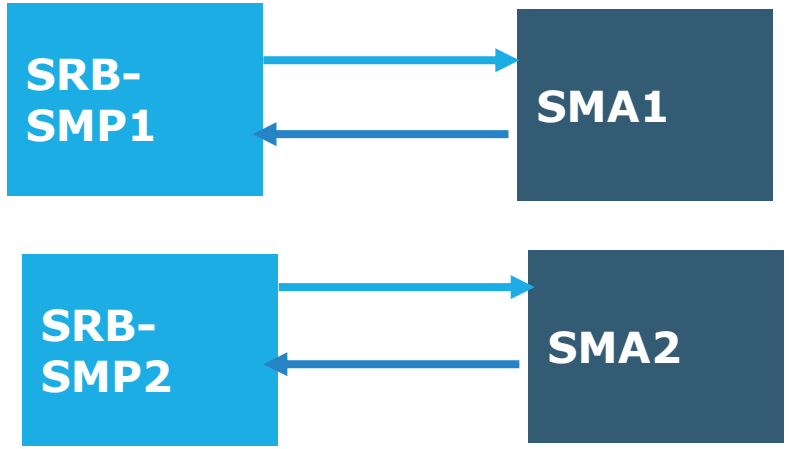
- Activation of zIIP eligibility
  - Specific rules for ISVs
- CA Datacom and zIIP Specialty Engine
  - Release 12.0 – offloads ~33% of CP CPU processing to zIIP
  - Release 14.0 – offloads ~50% of CP CPU processing to zIIP
  - Release 14.01 – **offloads ~80%+** of CP CPU processing to zIIP
  - Release 15.0 – offloads ~90+ of CP CPU processing to zIIP
    - Any technique that reduces IO reduction can improve TCO by swapping GP CPU for zIIP-able CPU

How to Improve TCO With Datacom – Reduce CPU With zIIP

- SMPTASK 5,3,3,**SRB**
  - SMPTASK startup option
  - Current default is TCB
  - Code **SRB** to run SMP tasks as SRBs
  - Main task runs as TCB (as well as other subtasks)
  - First two SMP tasks scheduled to run more aggressively
    - No longer based on ready to run
  - Other SMP tasks scheduled same as when in TCB mode
    - Based on ready to run
  - All SMP tasks are either SRB or TCB

How to Improve TCO With Datacom – Reduce CPU With zIIP (cont'd)

- Each SRB SMP task paired with a DBSMAPR TCB
  - Used for things an SRB can't do
  - Or things a zIIP can't do
  - Passed back and forth as synchronous requests



How to Improve TCO with Datacom – Reduce CPU With zIIP (cont'd)

- Example of SRB/TCB pairs

TCB/SRB USE SUMMARY INFORMATION

ITEM	PROGRAM	USED COUNT	CPU TIME	POSTED COUNT	I/O COUNT
1	DBMUFPR	77	000:00:00.47	66	56
2	*SRB-SMP	1,474,492	000:00:17.54	207,114	0
3	*SRB-SMP	970,405	000:00:11.21	74,101	0
4	*SRB-SMP	687,604	000:00:07.70	66,265	0
5	DBMSBPR	6	000:00:00.02	1	0
6	DBOCSPR	5	000:00:00.03	12	0
7	DBOCSPR	0	000:00:00.00	1	0
8	DBOC2PR	10	000:00:00.02	11	0
9	DBDUSPR	0	000:00:00.00	0	0
10	DBENQPR	15	000:00:00.00	15	0
11	DBCONPR	1	000:00:00.01	1	0
12	DBIOSPR	0	000:00:00.00	1	0
13	DBSMAPR	3,495	000:00:00.11	3,495	3,455
14	DBSMAPR	296	000:00:00.00	296	294
15	DBSMAPR	61	000:00:00.00	61	61

Main task TCB

First pair is busiest pair

Other tasks on TCB

How to Improve TCO With Datacom – Reduce CPU With zIIP (cont'd)

- Customer example: Large bank – week of processing
  - Running V14.01 zIIP
  - Running high number of buffers

TASK_TYPE	PROGRAM_NAME	SEQUENCE_NUMBER	CPU_SECONDS	PHYSICAL_IO
MAIN	DBMUFPR	1	-	224
SMP	*SRB-SMP	1	28,120	-
SMP	*SRB-SMP	2	5,115	-
SMP	*SRB-SMP	3	48	-
SMP	*SRB-SMP	4	-	-
SUB	DBCONPR	1	-	-
SUB	DBDUSPR	1	1	-
SUB	DBENQPR	1	-	-
SUB	DBMSBPR	1	-	-
SUB	DBOCSPR	1	14	-
SUB	DBOCSPR	2	-	-
SUB	DBOC2PR	1	8	-
SUB	DBSMAPR	1	1,041	141,034,552
SUB	DBSMAPR	2	58	7,661,602
SUB	DBSMAPR	3	-	36,921
SUB	DBSMAPR	4	-	727
TOT	TOTAL		34,405	148,734,026
	zIIP CPU		33,283	97%
	CP CPU		1,122	3%
	Requests	4,713,125,058		
	Request per total CPU	136,990		
	Request per adjusted CPU	1,059,058		

Remember: The real TCO comes from savings at peak period

Another site after V14.01 "we have eliminated our overcharges from IBM for using more MSU than we contracted for"

How to Improve TCO With Datacom – Reduce CPU (cont'd)

- Other considerations
  - Are you wasting CPU cycles
    - Are you checking out
      - Application errors (DB RCs or SQLCODEs)
      - PXX dump activity
      - Accounting activity
      - CBS temp indexes
      - SQL temp tables
      - Sequential read ahead
      - ???
    - This is where the activity snapshots can help
      - Look for different/increasing statistics

How to Improve TCO With Datacom – Reduce IO

- Database engines are physical IO consumers
- Physical IOs consume significant amounts of CPU
- Reduction of Physical IO leads to reduction of CPU consumed
  - Logical IOs (memory) are cheaper in CPU the physical IO
  - Logical IOs are zIIP-able (specialty processor eligible)

  - Remember the bank with 97% zIIP

PHYSICAL_READS	PHYSICAL_WRITES	LOGICAL_READS	LOGICAL_WRITES
174,110,661	22,949,238	19,255,247,592	1,279,430,045
	Logical to Physical	111	56

How to Improve TCO With Datacom – Reduce IO
Standard Buffers

- Ensure the buffer pools are large enough to provide high logical reuse

Buffer Processing		5 to 1 Ratios
IXX Buffer Use 5+	5,268,204,568	33675.9
DXX Buffer Use 5+	26,245,112,476	1231.8
DATA Buffer Use 5+	7,694,167,527	38.2
DATA2 Buffer Use 5+	-	0

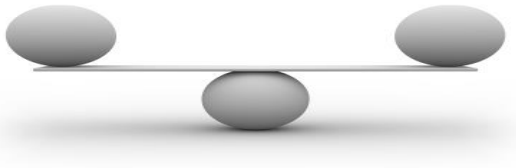
SYSPOOL_CXXNO	99
SYSPOOL_IXXNO	10,000
SYSPOOL_DXXNO	20,000
SYSPOOL_INDEX	4,096
DATAPOOL_DATALN	4,096
DATAPOOL_DATANO	70,000
DATAPOOL_DATALN2	-
DATAPOOL_DATANO2	-

- IXX/DXX buffers are 64-bit
  - Increase each until 5+ reuse “falls off” (increase in reasonable increments)
  - Monitor paging to ensure paging remains reasonable
- DATA/DATA2 buffers are 31-bit or 64-bit (V15)
  - Increase each until 5+ reuse “falls off” (increase in reasonable increments)
  - If using 31-bit storage (do not allow 31-bit to get exhausted)
  - Monitor paging to ensure paging remains reasonable

How to Improve TCO With Datacom – Reduce IO

Data Buffer Size

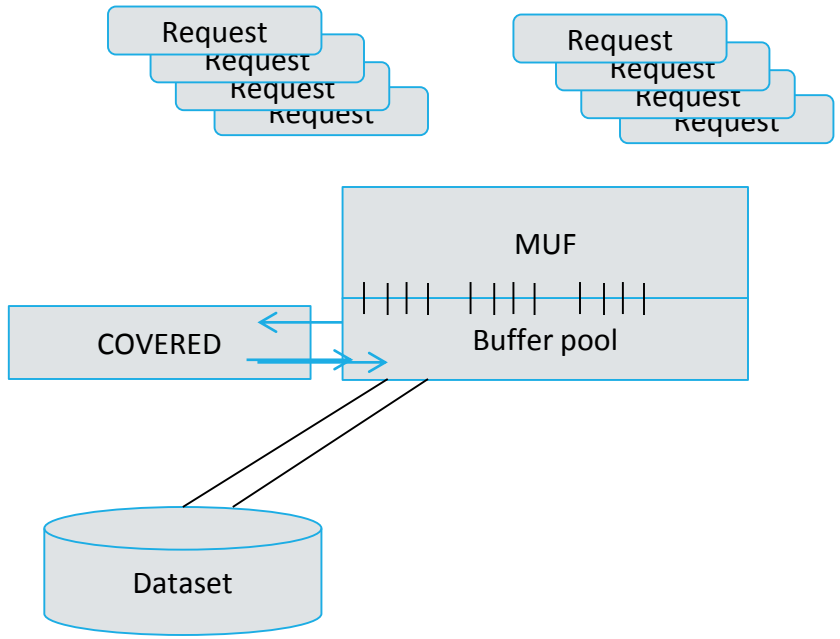
- Implement larger block size for selected data areas
  - If in use make sure they are being used efficiently
- Larger block size could reduce IO
  - Increase from 4K to 17K does not significantly affect cost of IO retrieval
  - The more rows per block, the more opportunity for improved reuse
  - Need to validate using BVMs
- Need to balance DATA/DATA2 buffers to area usage
  - Increase DATA2 as workload is moved to the large size
  - If using 31-bit storage, do not allow 31-bit to get exhausted
  - Monitor paging to ensure paging remains reasonable



How to Improve TCO With Datacom – Reduce IO

COVERED Areas

- Like a private buffer pool for a selected area
  - Provides a memory cache to avoid IO when requested block is not in a buffer



Request needs a row
- Initial request triggers IO to load buffer and COVERED
- Future requests serviced from buffer pool
- When not in current buffer pool, buffer reloaded from COVERED area
- In good situations, buffer reload from covered occurs repeatedly

How to Improve TCO With Datacom – Reduce IO

COVERED FIRST versus ACTIVE

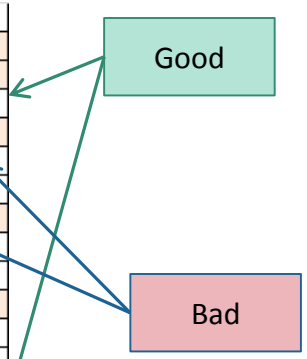
- FIRST uses the least amount of CPU to find a requested block
  - Direct mapping (data block 7 is in memory slot 7)
  - Does not adjust to activity
  - May not work well if area to be covered is too big
- ACTIVE uses more CPU to find a requested block
  - Adjusts to activity, similar to buffering
  - LRU is used to determine which memory slots are reused
  - Indirect mapping (data block 7 may be anywhere in the memory)
  - Works well if the “working set” of blocks fits within the amount of Covered memory

How to Improve TCO With Datacom – Reduce IO

COVERED statistics

- Review stats below – some good choices, some bad choices

DBID	AREA	V/C	F/A	DS	MEM_SIZE	Mem in Megs	BLKS_MAX	BLKS_CURR	TOTAL_READS	MRDF_READS	ACTV_READS	NONF_READS	IO Saved per M
2	DD1	C	A	X	41,943,040	40.0	10,240	10,240	132,580	119,295	13,285	NULL	2,982
2	IXX	C	A	X	20,971,520	20.0	5,120	5,120	79,924	67,605	12,319	NULL	3,380
24	USR	C	F	X	3,145,728	3.0	768	768	39,985	39,418	563	4	13,139
44	MM2	C	A	X	52,428,800	50.0	12,800	12,800	339	181	158	NULL	4
48	BEN	C	F	X	73,400,320	70.0	17,920	17,920	3	-	1	2	-
48	CNN	C	F	X	15,728,640	15.0	3,840	3,840	4	-	3	1	-
52	RFL	C	A	X	491,520	0.5	120	120	100	61	39	NULL	130
93	DTP	C	F	X	15,728,640	15.0	3,840	3,840	3	-	1	2	-
93	ETP	C	F	X	7,340,032	7.0	1,792	1,792	493	373	119	1	53
96	MRK	C	A	X	52,428,800	50.0	12,800	12,800	1,730	66	1,664	NULL	1
97	PAM	C	F	X	40,960	0.0	10	10	215	210	4	1	5,376
100	TAB	C	F	X	10,485,760	10.0	2,560	2,560	10,157	7,921	424	1,812	792
124	TCI	C	F	X	41,943,040	40.0	10,240	10,240	221,223	59,843	4,597	156,783	1,496
	TOT	C	N	N	336,076,800	320.5	94,850	94,850	502,007	310,224	33,177	158,606	85



- Covering is not a “set-once and forget” process
  - Needs reviewing on a monthly basis
- Covering allocations can be changed on the fly to mirror processing needs
- Covering should be targeted to high read areas
  - IXX over DATA if same value

How to Improve TCO With Datacom – Reduce IO

COVERED Looking for Mr. Good Dataset

DBID	AREA_NAME	PHYSICAL_READS	PHYSICAL_WRITES	LOGICAL_READS	LOGICAL_WRITES	Total IO	% of total IO	% of READ Total	% cum.	Blzsize	BLK in-use	Megs to cover	Saved IO per Meg	
	TOT	174,110,661	22,949,238	19,255,247,592	1,279,430,045	197,059,899	100%	100%		na				
52	CSC	42,529,785	3,693	222,052,716	18,179	42,533,478	22%	24%	24%	4096	119,598	467	81,932	
137	ICU	18,855,686	178	26,339,118	819	18,855,864	10%	11%	35%	4096	60,995	238	71,225	
0	LXX	235,701	14,049,415	235,701	264,845,651	14,285,116	7%	0%	35%	na	na			
52	IXX	9,720,622	1,183,152	1,236,503,910	58,899,957	10,903,774	6%	6%	41%	4096	3,511,136	13,715	638	
54	DOE	9,691,489	4,755	38,882,215	425,491	9,696,244	5%	6%	47%	4096	527,057	2,059	4,237	
133	RS2	9,339,824	152,185	141,679,071	13,194,105	9,492,009	5%	5%	52%	4096	561,506	2,193	3,832	
63	HIP	8,992,581	405	204,160,105	52,262	8,992,986	5%	5%	57%	4096	2,406,612	9,401	861	
22	LCP	7,104,610	1,206	33,253,522	2,246,033	7,105,816	4%	4%	61%	4096	538,663	2,104	3,039	
97	RFM	6,272,713	-	41,621,165	-	6,272,713	3%	4%	65%	4096	3,784,213	14,782	382	
52	SVC	5,156,326	591	8,154,109	3,497	5,156,917	3%	3%	68%	4096	423,286	1,653	2,807	
21	LGC	4,807,483	935	265,393,781	207,653	4,808,418	2%	3%	70%	4096	3,249,694	12,694	341	
54	DOR	3,219,502	2,105	19,380,469	304,153	3,221,607	2%	2%	72%	4096	358,220	1,399	2,071	
61	IXX	989,095	2,227,920	575,799,358	142,570,906	3,217,015	2%	1%						
133	RS3	2,313,655	2,350	669,310,088	2,774,097	2,316,005	1%	1%	1%	4096	1,024,627	4,002	520	
145	IXX	1,727,536	555,350	513,371,235	30,696,367	2,282,886	1%	1%	2%	4096	978,710	3,823	407	
33	CIP	2,001,607	271,457	10,053,464	1,117,584	2,273,064	1%	1%	3%	4096	14,258	56	32,345	
61	CCT	1,859,895	340,392	92,657,430	93,138,260	2,200,287	1%	1%	5%	4096	414,419	1,619	1,034	
109	IXX	2,077,641	10,113	1,734,764,292	119,348	2,087,754	1%	1%	6%	4096	841,635	3,288	569	
21	LTR	1,909,710	2,583	3,552,706,305	2,963,872	1,912,293	1%	1%	7%	4096	180,000	703	2,444	Used alloc
52	ALS	1,883,670	4,134	5,737,347	18,817	1,887,804	1%	1%	8%	4096	75,331	294	5,761	
52	RCO	1,765,509	7,116	79,475,473	41,376	1,772,625	1%	1%	9%	4096	459,014	1,793	886	
80	TRH	1,753,145	9,523	29,975,037	11,308,354	1,762,668	1%	1%	10%	4096	1,004,528	3,924	402	

How to Improve TCO With Datacom – Reduce IO

COVERED versus Buffers

- Buffers are the most efficient use of memory when sized to match the data, but
  - Data buffer pools in v14 are limited due to 31-bit
  - Buffers are reused base on LRU (Least Recently Used)
    - A very active batch report against a non-essential table could flush a large portion of the active buffer pool
- COVERED areas
  - Are 64-bit by default
  - Covered blocks are specific to a given area and are not flushed by non associated activity
  - But
    - COVERED areas must still funnel their blocks back through the buffer pools
    - Covering low activity areas uses the same space that high activity areas uses

How to Improve TCO With Datacom – Reduce IO

Best Choice – Use Both

- Build buffer pools to a reasonable size to support typical “peak activity” within the system
- Use Covering to target key high-use (read) datasets
- Always measure every change to make sure it presents business value!!!

How to Improve TCO With Datacom – Reduce IO

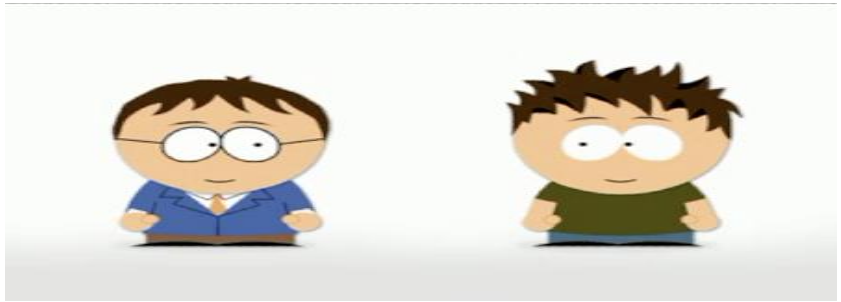
Reduce DASD Footprint

- Implement compression
  - Compression reduces the row image so more rows per block
    - Reduces DASD utilization (reduces cost)
    - Can improve buffer reuse
    - Can improve COVERED efficiency
  - DB compression runs in MUF and is zIIP eligible
    - With zIIP processors, the compression CPU costs are shifted to less costly CPU devices
  - An option that once was considered outdated
    - DASD cheap – CPU Expensive
    - May now be back in style – DASD is cheap but zIIP Cheaper
- Always measure with BVMs!!
- New cool option in 14.02

cheep, cheep, cheep….

Version 15.0 Techniques

- Data buffer pools now go to 64-bit
  - Reduced 31-bit consumption
  - Ability to create larger (or more) buffer pools
- Can be combined with additional buffer pools to build tailored buffer pools that resemble COVERED areas
  - Tailored buffer pools would have some advantage over covered areas
    - No requirement to move “through” a buffer pool like COVERED
    - Buffer LRU processing is highly efficient
- Which is better
  - Verdict is still out
  - We would like to have your feedback



Version 15.0 Techniques Under Review

- Are you seeing lots of “other SRB” with zIIP
- Consider participating in validation study for V15.0
- New zIIP SRB dispatching code
  - V14.02 validation sites have new limited test option
  - V15.0 sites will have a full implemented version

		JARS						
Job Name	Number	Elasped	TCB CPU	SRB-CPU	Total CPU	LXX	IXX	Data
UNCTCBR9	2912	00:13:20.05 *	00:00:33.29 *	00:00:04.83 *	00:00:38.12 *	20	15,431	1,526,828
UNCSMPR9	3331	00:09:08.03 *	00:00:34.38 *	00:00:04.92 *	00:00:39.30 *	20	14,819	1,632,668
UNCSRBR9	3522	00:08:05.83 *	00:00:15.30 *	00:00:15.01 *	00:00:30.31 *	20	18,352	2,245,262
UNCNEWR9	3718	00:06:24.03 *	00:00:09.43 *	00:00:03.64 *	00:00:13.07 *	20	14,819	1,252,531
UNCTCBR9	3790	00:04:54.37 *	00:00:31.76 *	00:00:04.22 *	00:00:35.98 *	20	14,819	1,216,990
UNCSMPR9	3889	00:04:41.22 *	00:00:31.53 *	00:00:03.97 *	00:00:35.50 *	20	14,819	1,219,687
UNCSRBR9	4000	00:05:25.47 *	00:00:09.49 *	00:00:10.25 *	00:00:19.74 *	20	14,819	1,243,627
UNCNEWR9	4127	00:06:54.40 *	00:00:10.12 *	00:00:03.85 *	00:00:13.97 *	20	14,819	1,215,245
UNCTCBR9	4581	00:01:26.14 *	00:00:21.29 *	00:00:01.12 *	00:00:22.41 *	20	14,819	58,826
UNCSMPR9	4862	00:01:28.37 *	00:00:23.86 *	00:00:01.11 *	00:00:24.97 *	20	14,819	58,826
UNCSRBR9	5177	00:01:42.67 *	00:00:01.16 *	00:00:03.16 *	00:00:04.32 *	20	14,819	58,826
UNCNEWR9	5370	00:01:16.48 *	00:00:01.28 *	00:00:00.31 *	00:00:01.59 *	20	14,819	58,826

Summary

- Improving performance and reducing TCO can be accomplished in many ways within CA Datacom
- The top techniques revolve around
  - Move CPU consumption to inexpensive specialty processors
  - Reducing activities that consume CPU without benefit
    - Programs that get “bad return codes” as a normal mode of operation
    - Building temporary tables or indexes
    - Etc.
  - Reducing IO by using buffering and covering to maximize memory use and reduce CPU used to do IO

FOR INFORMATION PURPOSES ONLY

Terms of this Presentation

This presentation was based on current information and resource allocations as of April 2016 and is subject to change or withdrawal by CA at any time without notice. Notwithstanding anything in this presentation to the contrary, this presentation shall not serve to (i) affect the rights and/or obligations of CA or its licensees under any existing or future written license agreement or services agreement relating to any CA software product; or (ii) amend any product documentation or specifications for any CA software product. The development, release and timing of any features or functionality described in this presentation remain at CA's sole discretion. Notwithstanding anything in this presentation to the contrary, upon the general availability of any future CA product release referenced in this presentation, CA will make such release available (i) for sale to new licensees of such product; and (ii) to existing licensees of such product on a when and if-available basis as part of CA maintenance and support, and in the form of a regularly scheduled major product release. Such releases may be made available to current licensees of such product who are current subscribers to CA maintenance and support on a when and if-available basis. In the event of a conflict between the terms of this paragraph and any other information contained in this presentation, the terms of this paragraph shall govern.

Certain information in this presentation may outline CA's general product direction. All information in this presentation is for your informational purposes only and may not be incorporated into any contract. CA assumes no responsibility for the accuracy or completeness of the information. To the extent permitted by applicable law, CA provides this presentation "as is" without warranty of any kind, including without limitation, any implied warranties or merchantability, fitness for a particular purpose, or non-infringement. In no event will CA be liable for any loss or damage, direct or indirect, from the use of this document, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if CA is expressly advised in advance of the possibility of such damages. CA confidential and proprietary. No unauthorized copying or distribution permitted.

Questions?