The next step in big data starts with IBM.
Extreme speed. Increased flexibility. Enhanced performance.

DB2 and the zIIP Processor: Exploitation, Not Abuse

Adrian Burke DB2 for z/OS SWAT Team
agburke@us.ibm.com

DB2 11 for z/OS
The Enterprise Data Server for Business Critical Transactions and Analytics.

Important Disclaimer

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY.

WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE.

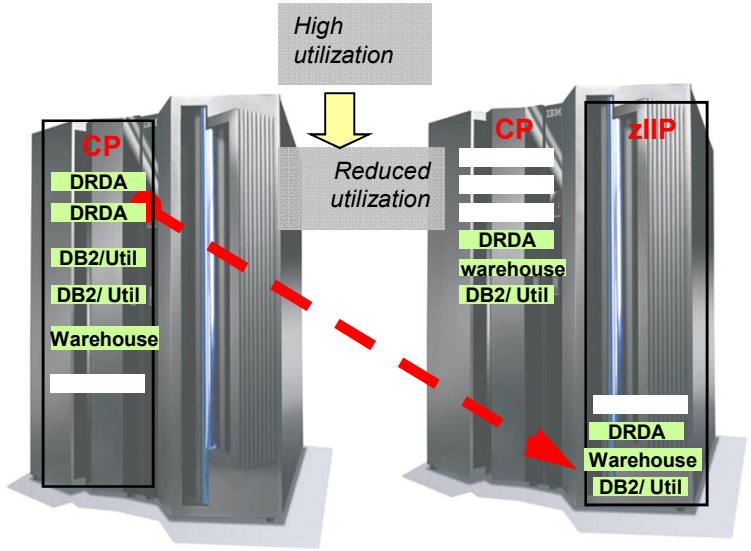
IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION.

NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, OR SHALL HAVE THE EFFECT OF:

- CREATING ANY WARRANTY OR REPRESENTATION FROM IBM (OR ITS AFFILIATES OR ITS OR THEIR SUPPLIERS AND/OR LICENSORS); OR
- ALTERING THE TERMS AND CONDITIONS OF THE APPLICABLE LICENSE AGREEMENT GOVERNING THE USE OF IBM SOFTWARE.

Agenda

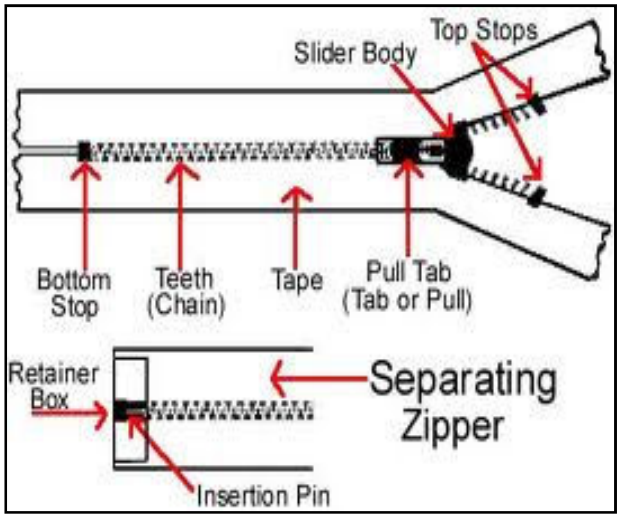
- Background
  - Distributed access
- Capacity
  - Un-zIIPed work
- Eligibility
  - Recent enhancements
- Exploitation
  - What can I control?



Background

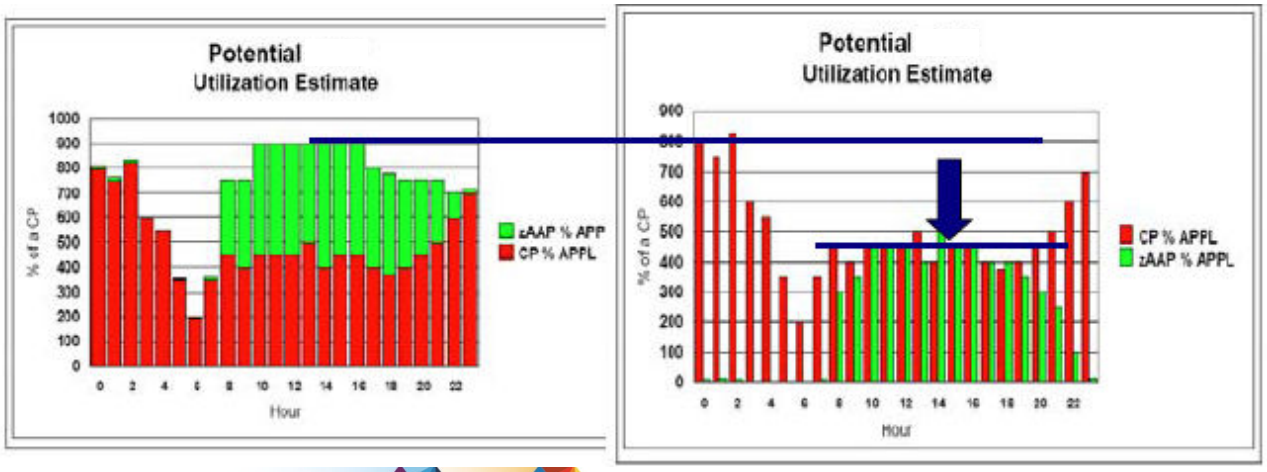
- Generally available in 2006 on the System z9 hardware
- IBM System z Integrated Information Processor (IBM zIIP)
- Quick jargon lesson
  - zIIP-able
    - Work that is zIIP eligible
  - zIIP-ed
    - Work that executed on a zIIP
  - Un-zIIP-ed
    - Eligible work that executed on a general CP

- *Microcode enabled*
- *Dispatcher controlled*



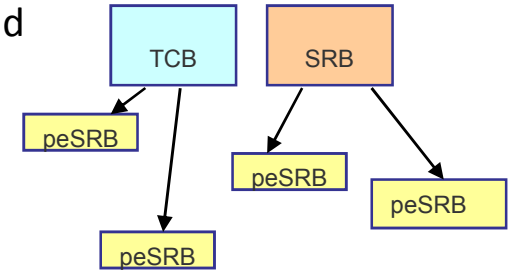
How can specialty engines help me?

- Software costs: MSU units, generally increase with the # of general processors, while zIIP processors do not add to the total MSU count
- Hardware costs: move work from GP to zIIP (zAAP), higher cost to lower cost processors, possibly postpone an upgrade
  - Specialty engines run at full rated speed of processor, so it could be the fastest one on the CEC
- BUT/AND…. it can also result in latent demand processing so processor utilization remains constant



Work is dispatched

- There are four types of dispatchable units in z/OS:
  - Preemptible Task Control Block (TCB)
  - Non-preemptible Service Request Block (SRB)
  - Preemptible Client Service Request Block (client SRB)
  - Preemptible Enclave Service Request Block (enclave SRB)
- Some are zIIP eligible
  - IBM moves TCB and non-preemptible SRB work to enclaves as a way to increase offload



DBM1 SRBs (* means data sharing)

- Asynchronous I/O (enclave SRB in V10)
- Memory management
- Prefetch (enclave SRB in V10)
- Real time stats
- *Castout* (V11)
- *P-lock negotiation*
- *GBP checkpoints*
- Backout (preemptible V10)

DBM1 TCBs

- Open/close
- Pre-format/ extend
- Full system contraction

What are enclave SRBs?

- z/OS dispatches DB2 work in either TCB, Client SRB, or Enclave SRB mode if request is local or an Enclave SRB (Service Request Block) mode if request is distributed.
  - Preemptible enclaves are used to do the work on behalf of the originating address space
  - If you do not have a classification for the distributed work it falls into SYSDEFAULT (Discretionary)
- If the DB2 for z/OS request is coming in thru the DIST address space (i.e. DRDA over TCP/IP) then that work is executed in enclave SRBs, and can run in a different WLM service class than DIST
  - Only the enclave SRB work is eligible to be redirected to the zIIP.

- Why would IBM start exploiting the zIIP with this workload ?
  - Historically distributed workload cost ~50% more than locally attached work
  - Goal is to encourage more → distributed → SOA → Mobile workload on z

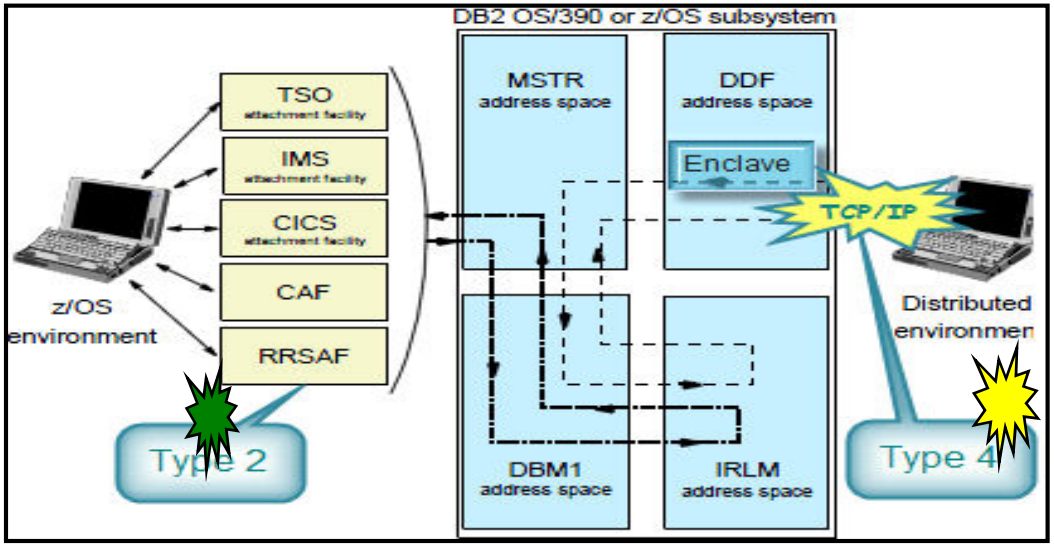
Why not SNA?

- If DB2 for z/OS workload comes over TCP/IP and is DRDA compliant, a portion of that DB2 workload is eligible to be redirected to the zIIP
- Many customers still use DRDA over SNA for DB2 z/OS to DB2 z/OS calls
  - As of DB2 9 SNA incurs overhead due to DIST going to 64-bit addressing
  - Look in the statistics long report and compare the SRB times in the DDF Address space CPU
    - The PREEMPT IIP SRB time should dwarf PREEMPT SRB if the DRDA work is coming in over TCP/IP and thus zIIP eligible
- This customer migrated from SNA to TCP/IP and measured a 24 hour period before and after
  - 58% of the CPU used by DIST address space was offloaded to the zIIP
  - The CPU per commit was reduced by 66%
    - Watch out if you use INBOUND AUTHID translation in SNA, not there in TCP/IP

	CPU TIMES	TCB TIME	PREEMPT SRB	NONPREEMPT SRB	CP CPU TIME	PREEMPT IIP SRB	CP CPU /COMMIT
SNA	DDF ADDRESS SPACE	15.759816	11:40:48.726492	3:25:35.999739	15:06:40.486046	9:45.940413	0.008500
TCP/IP	DDF ADDRESS SPACE	14.758614	6:14:38.618730	22:31.612655	6:37:24.989999	9:06:58.739546	0.002866

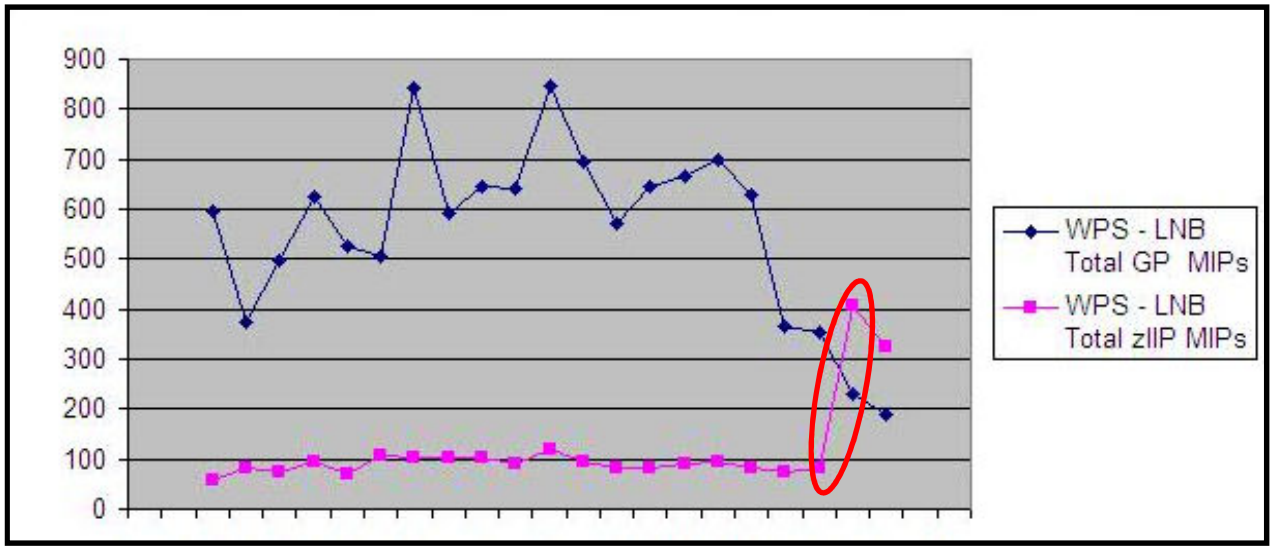
T2 vs. T4 Connections same LPAR

- Offload (T4) or efficiency (T2)
  - Going through DIST adds network translations, another address space, and context switch to an SRB but you get zIIP offload
  - DB2 objective is to improve T2 driver performance to beat T4
  - Even **local** T4 connection hits DIST and TCP stack – need DDF WLM service class
  - Moving T2 to T4 is not 1 for 1 MIP exchange due to overhead
    - But, the worse behaving the application the more there is to offload so in the end it must be tested and compared



zIIP vs. GP while changing to T4 driver

- In 1 day customer switched all T2 to T4 connections, WPS and DB2 z/OS both on same LPAR… Not A Benchmark
  - GP MIPS from 350 → 230
  - zIIP MIPs from about 100 → 405 ….. Not a 1 for 1 exchange



Stored Procedures with zIIPs

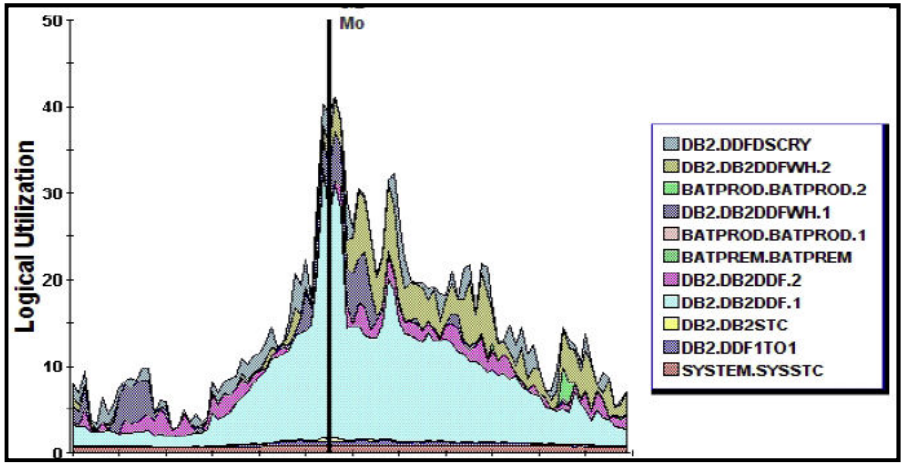
Language	Base Billable Cost	Billable Cost after zIIP and/or zAAP acceleration
COBOL stored proc	X (Baseline)	.88x
C stored proc	.95x	.83x
Remote SQLJ	1.78x	1.06x
SQLJ stored proc	1.21x	1.15x (zIIP + zAAP)
JDBC stored proc	2.11x	1.76x (zIIP + zAAP)
External SQL stored proc	1.62x	1.49x
Native SQL stored proc	1.14x	.65x

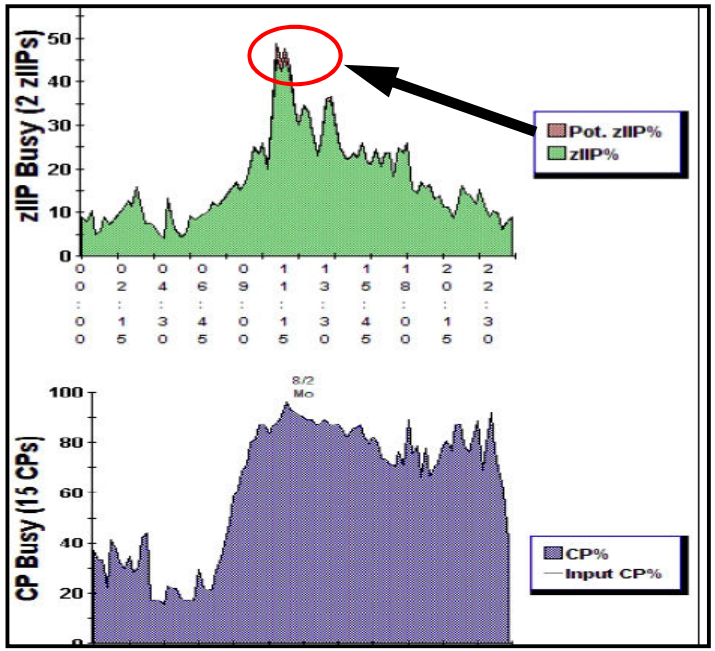
** Need to be invoked remotely for zIIP offload

CAPACITY

How many zIIPz should I have?

- zCP3000 study provided by IBM Techline
  - Send in SMF 30's, 70's
  - Breakdown of eligible work
  - Overlay 4 hour peak
  - See collisions of workloads
  - **Remember to normalize**

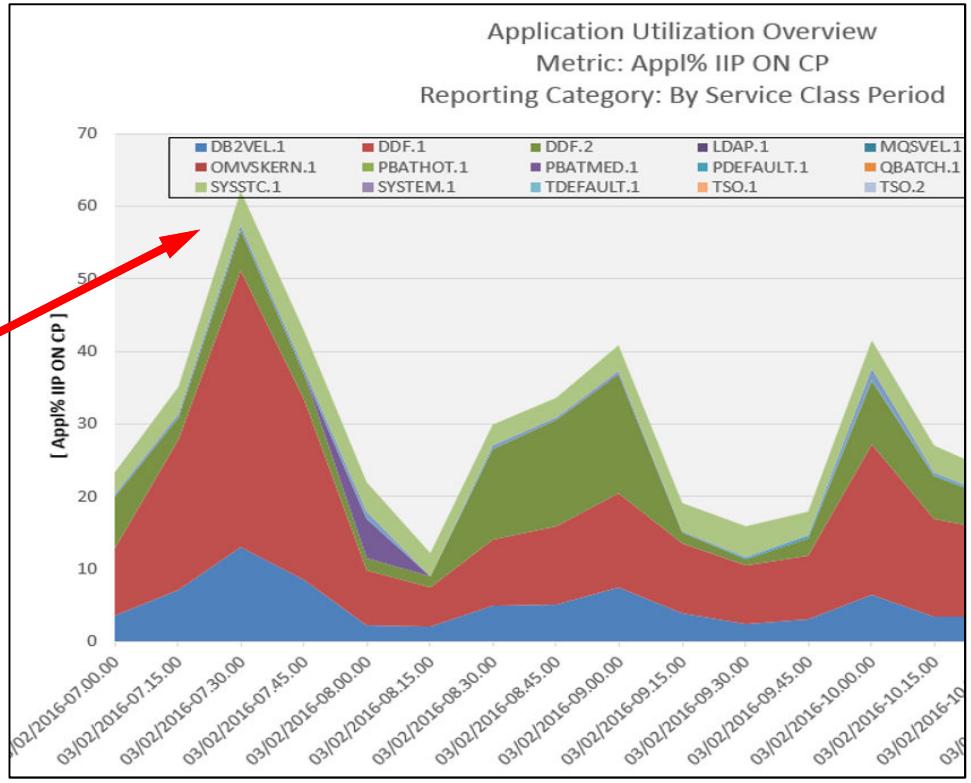




http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS3948

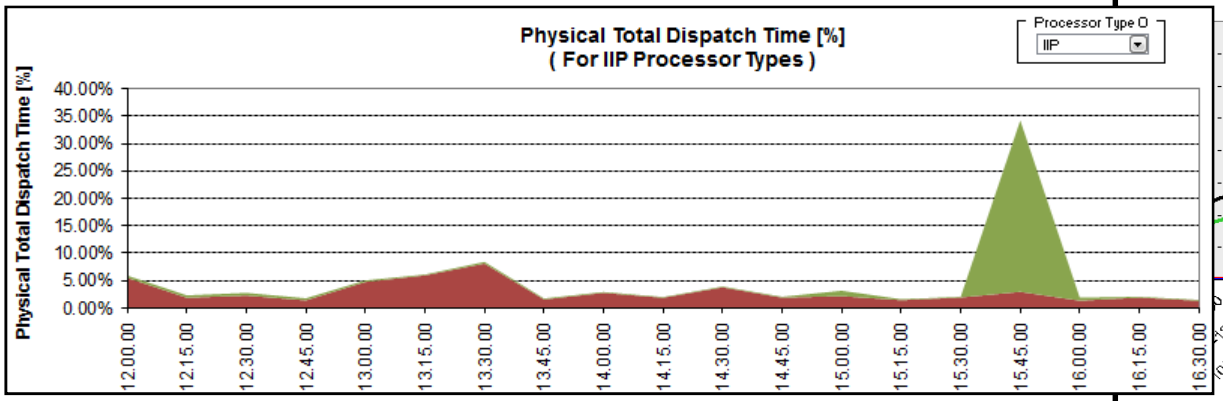
Measuring zIIP overflow

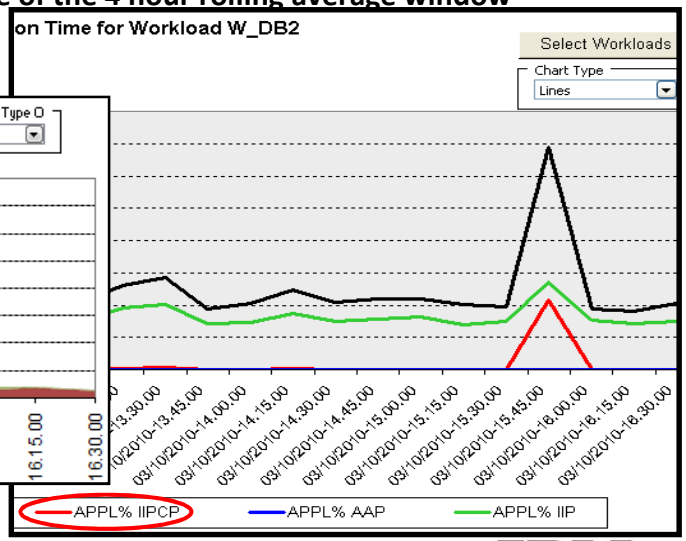
- Capacity planning should monitor zIIP overflow to the GCPs, ***not*** physical utilization
- The WLM activity report (SMF 72-3) records zIIP eligible work that ran on a GCP as a % of utilization of a single GCP (APPL% IIPCP CPU)
  - Broken down by WLM service/report class in RMF
- zIIPs always run at speed of a 701 processor so this workload may only need 30% of a zIIP if it is 2x speed of the GCPs on the box
- zIIP redirect means work wait in a queue for a zIIP, as well as aggravating RNI of the LPAR
  - Relative Nesting Intensity (RNI) affects MIP consumption → no L1, L2, or L3 CPU cache hit if work moves from a zIIP to GCP



zIIP overflow

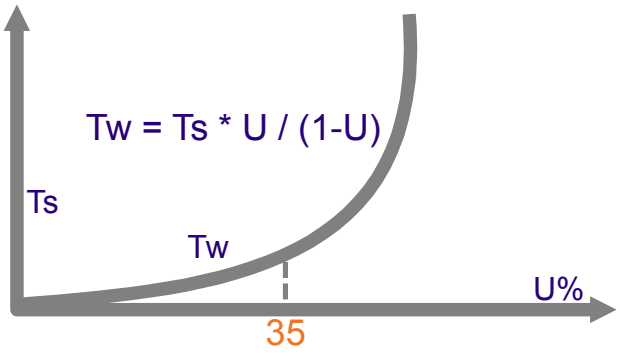
- How many zIIPs do you need (this scenario 12:1 ratio CP to zIIP)
  - zIIP eligible work went to CP either because zIIP is overloaded - Red line on graph (APPL% IIPCP) – missed opportunity for savings
  - **Needs Help algorithm** ensures work does not pile up waiting on zIIP
  - Must have enough capacity to absorb spikes, not just typical offload
    - **Size the zIIP for the spikes, it doesn't matter if it is only 10% utilized outside of the 4 hour rolling average window**
- Law of probability for many CPs vs. zIIPs (next slide)

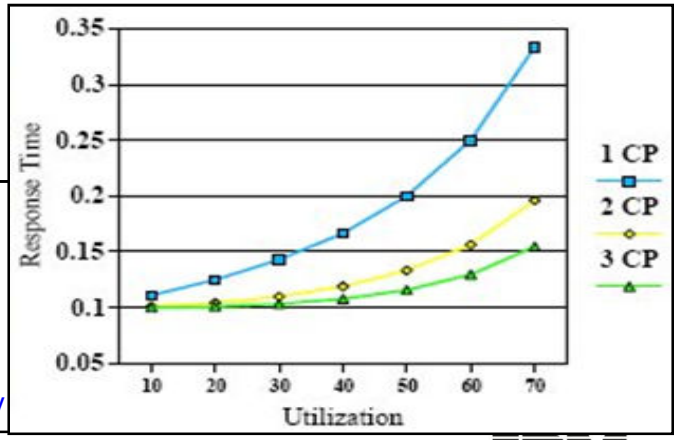




zIIP Overflow

- If 12 CPs are 65% (0.65^12) utilized then each CP is 0.5% instantaneously busy
  - If 1 zIIP is 35% busy then 35% of the time it is 100% busy
  - So with 'needs help' algorithm it is likely some zIIP eligible work could fall back to a CP
    - See IIPHONORPRIORITY, later slide
- Markov's Equation is based on 1 server (CP) in steady state
  - As Utilization approaches 100% wait time approaches ∞
  - This will cause more work to overflow to a CP starting at around 35% utilization of a single zIIP processor
  - More zIIPs = more offload





Tw = wait time of transaction

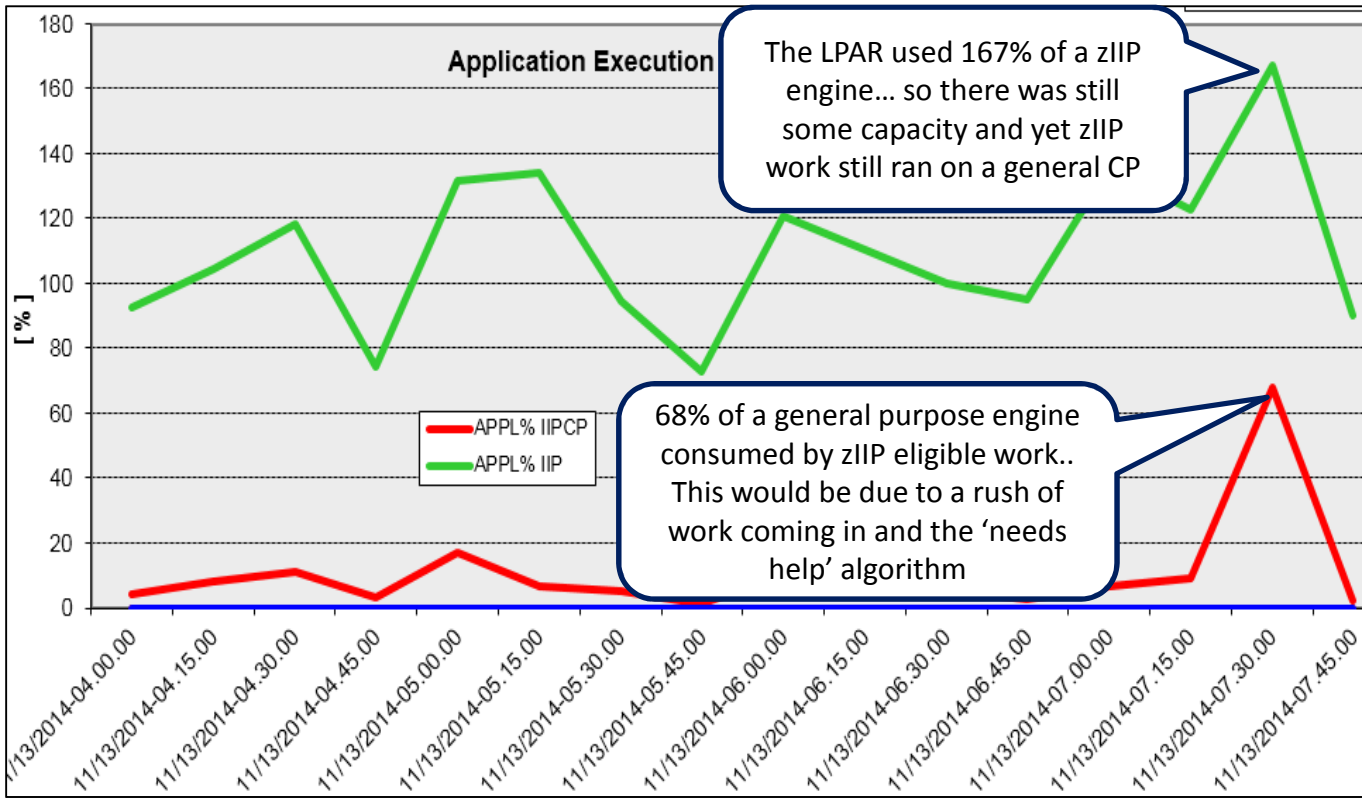
Ts = service time of transaction

U= utilization

The knee of the curve occurs at 35% for 1 processor, thereafter Tw increases drastically

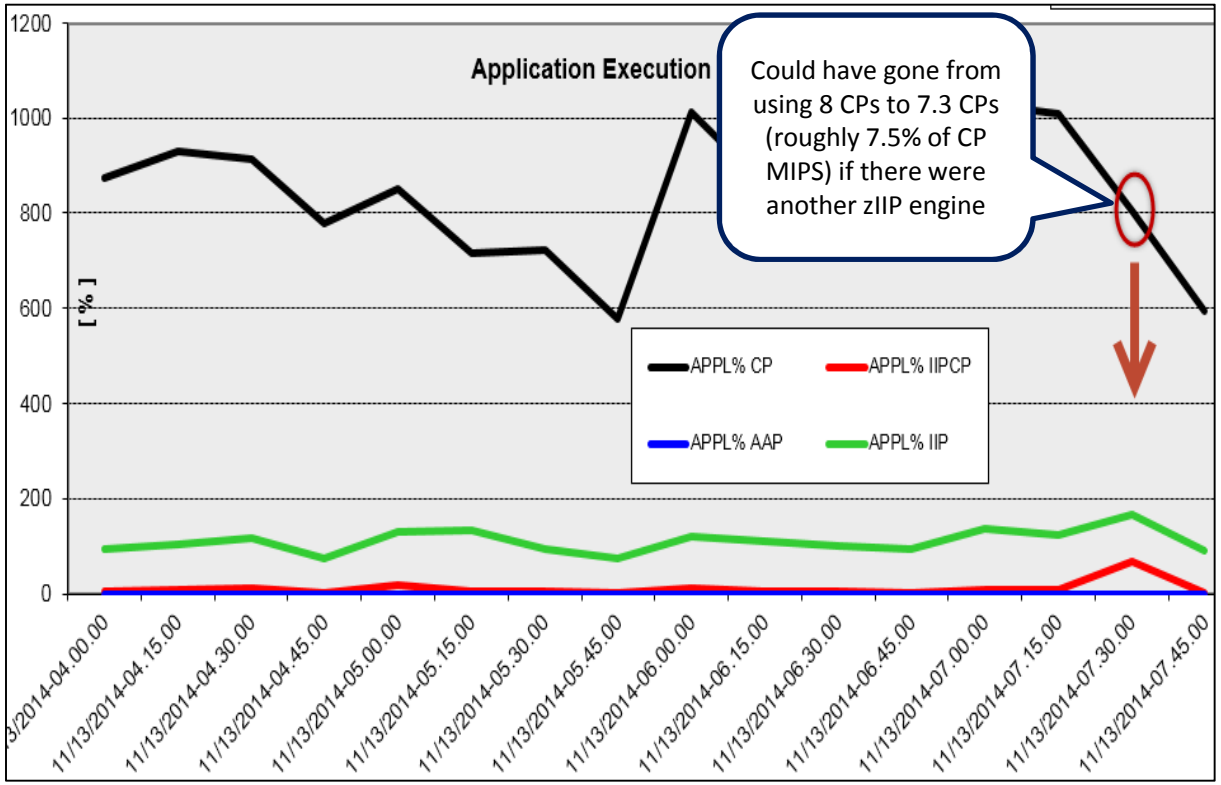
System zIIP Shortage

- Looking at a Customer's 4 hour rolling average peak for the month, 2 zIIPs are installed:
  - The 2 zIIPs are normally not fully utilized, but during the peak there is about 70% of a processor with of IIPCP, or zIIP eligible work that ran on a GP



Total CP usage

- Looking at a Customer's 4 hour rolling average peak for the month, there are just over 10 GCPs in use and 2 zIIPs available
  - When zIIP eligible work ran on the GPs it represented about 7.5% of the chargeable MIPs for DB2 on the system during that interval, which could affect the MLC bill
  - In this case the CPs were full-speed, but if they were knee-capped you would need to multiply the APPL% IIPCP CPU by the MSU ratio difference



zIIP Overflow…

- Aside from simple queue theory there are other reason's you could be seeing zIIP work spill-over to the GCPs
- z/OS local lock contention for zIIP eligible tasks is reported as zIIP eligible time in RMF, but cannot in-fact run on a zIIP processor
  - This local lock is used for storage acquisition (i.e. getmain) and there is only 1 per address space
- If you have single digit % of IIPCP CPU could be due to lock contention
  - Evidence of contention can be seen in PROMOTED for lock (LCK) field in WLM activity report
  - This shows CPU used to promote waiters <u>not</u> IIPCP CPU time, but implies relative use of the local lock, and if the task was zIIP eligible this time would be reported in IIPCP CPU
- In order to prove this was the IIPCP CPU % you could turn IIPHONORPRIORITY= NO
  - **BUT** then you loose system agent offload in V11!!
- If it is in-fact lock contention there is no way to tune this away

--PROMOTED--	
BLK	0.000
ENQ	0.000
CRM	0.000
LCK	0.659
SUP	0.000

DDF workload

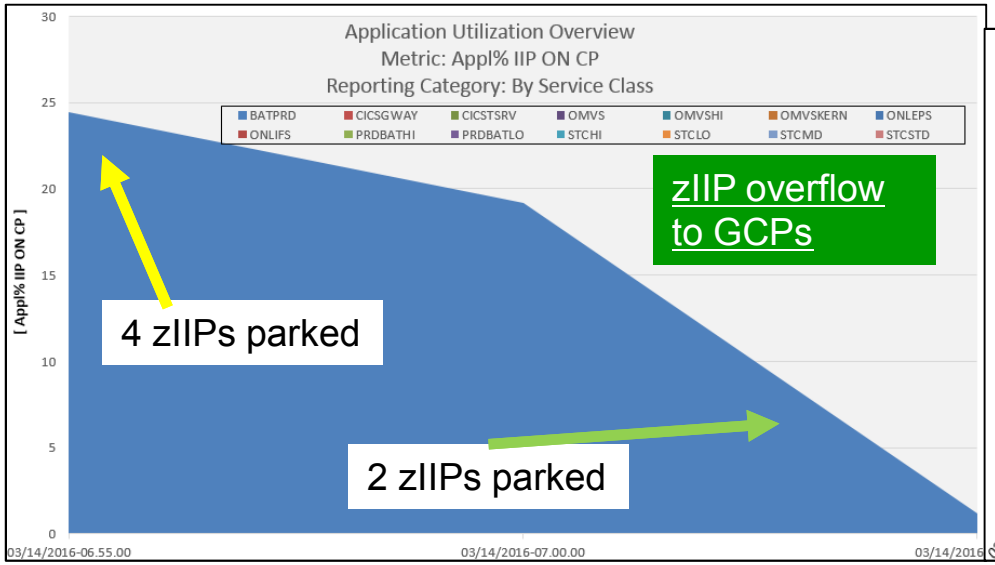
DBM1 address space

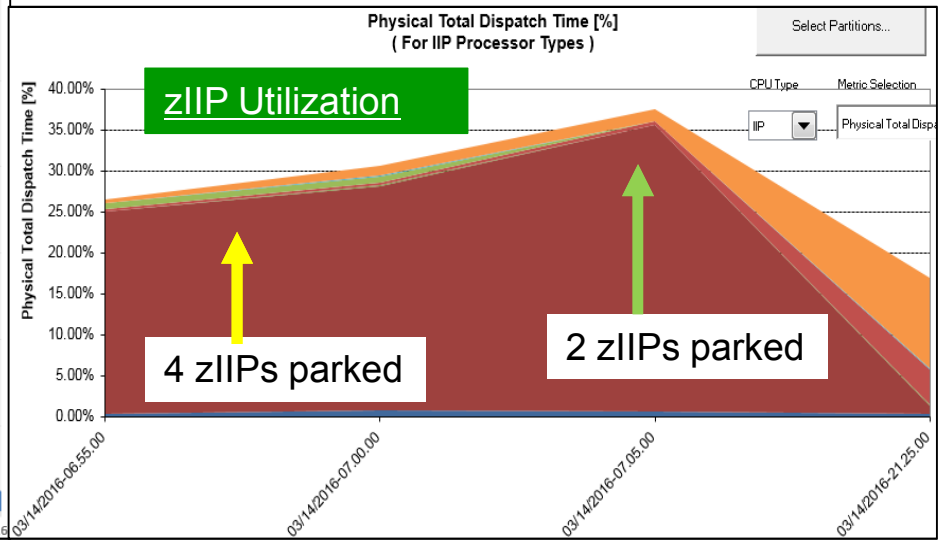
--PROMOTED--	
BLK	0.000
ENQ	0.000
CRM	0.000
LCK	8.699
SUP	0.000

zIIP Overflow…LPAR Weights

- Hiperdispatch is VERY sensitive to the relative LPAR weights (HIGH/MED/LOW)
  - Key is to apportion weights based on actual utilization – not share zIIPs with everyone
  - Otherwise engines will remain parked causing work to spill over to the GCPs
- Many zIIP eligible workloads are 'spikey' in nature – look in CPU activity →
  - Rush of DRDA requests, Utilities, or SQL CPU parallelism

---------NUMBER OF WORK UNITS----------			
CPU TYPES	MIN	MAX	AVG
CP	1	51	9.5
IIP	0	306	2.1





ELIGIBILITY

zIIP Eligibility

Release	Function	Amount Redirected	Pre-reqs
DB2 10	1. All of DB2 v8 and 9 offload++ 2. RUNSTATS 3. Prefetch and deferred write processing 4. Parallelism enhancements 5. multi-version XML clean-up	1. BUILD phase, Remote Native SQL procs, parallelism, 60% DRDA requests 2. Basic RUNSTATS for table, NO Histogram, DSTATS, COLGROUP… BUT index stats almost all offloaded (not DPSIs) 3. 100% (roughly 80% of DBM1 SRB time) 4. Parallelism more likely (80% of child tasks) 5. All of it	1. DB2 10/ z/OS 1.10 2. Run RUNSTATS, no inline STATS 3. Shows up in DBM1 SRB time 4. V10 NFM with rebind 5. PM72526
Other stuff	1. IPSec 2. Global Mirror for z/OS (formerly Extended Remote Copy) 3. HiperSockets for Large messages 4. DFSORT 5. zAAP on zIIP	1. Encryption processing, header processing and crypto validation (93% for bulk data movement) 2. Most System Data Mover processing 3. Handles large outbound messages (multiple channel paths given to SRBs) 4. Sorting of fixed length rows (10-40% Utility), memory object work file sorts 5. zAAP eligible work can move to zIIP	1. N/A 2. N/A 3. GLOBALCONFIG ZIIP IQDIOMULTIWRITE 4. PM62824 and z/OS 1.12 5. z/OS 1.11 base or 1.9 or 1.10 w/ APAR OA27495 / OA38829 if both

zIIP Eligibility

Release	Function	Amount Redirected	Pre-reqs
DB2 11	1. More RUNSTATS 2. LOAD REPLACE with dummy input 3. Most of the system engines (GBP write, castout, log write/ prefetch,) 4. Index pseudo delete 5. PARAMDEG_DPSI	1. COLCARD, FREQVAL, HISTOGRAM statistics, including inline stats (80%, possibly more) 2. 100% of delete processing eligible 3. 100% eligible 4. 100% eligible 5. 100%	1. N/A 2. N/A 3. N/A 4. INDEX_CLEANUP_THREADS >0 5. Parallel query access through DPSI parts
DB2 12	1. Parallel child tasks 2. DRDA fast load 3. RELOAD phase of REORG and LOAD 4. Fast Traversal Block for buffer pools	1. 100% 2. 100% movement of data blocks to LOAD 3. ~17% for REORG, ~90% for LOAD 4. 100% of parent daemon	1. CPU query parallelism 2. Fast load from client 3. N/A 4. Enable FTBs
Other stuff	1. z/OS Connect Adaptor 2. DFSORT…DB2SORT 3. ZAAP on zIIP	1. 100% 2. Sorting of fixed length rows (10-40% Utility), memory object work file sorts… 10-20% for DB2SORT 3. 100%	1. JSON API access to DB2 z/OS data 2. PM62824 3. zAAP support removed with z13

Other Processes

- IPSEC PROFILE.TCPIP settings
  - IPCONFIG IPSECURITY
  - GLOBALCONFIG → ZIIP → IPSECURITY
    - Displaces related CPU cycles to the zIIP (default is NOIPSECURITY)
  - Netstat STATS/-S command will show 'Packets Handled by zIIP'
- Hipersocket multi-write operations, must be 32k in size
  - Usually related to XML, file transfers, SOAP
  - IQDIOMULTIWRITE
    - GLOBALCONFIG → ZIIP → IQDIOMULTIWRITE
      - Default is NOIQDIOMULTIWRITE
- XRC (Global Mirror DFSMS SDM [system data mover]) startup member of PARMLIB
  - ANTAS000, ANTAS0nn, and ANTCL0nn address spaces enabled for zIIP processing
  - ANTXIN00 parmlib member
    - ZiipEnable(FULL) – gives you the max, otherwise YES allows everything but I/O operations)
      - Address space and SRB components offload
- OMPE
  - CICS SLA report builder, up to 73%
  - DB2 Near Term History, all of processing for normalizing the raw SMF
  - DASD UCB sampling 2-10% CPU savings

Other Processes…

- z/OS Connect – runs in a JVM, even inside of CICS region
  - Allows for T2 or T4 connectivity from CICS to DB2
- zOSMF – written in JAVA so it is eligible
  - Look for future releases of DB2 to take advantage of workflows in here
- DFSORT
  - Offload memory object work file code path for fixed/variable length sorts
    - About 30-60% of sort CPU for REORG
- DB2 Sort
  - Offloads fixed and variable length sorts
  - CPU and elapsed time savings over DFSORT with Utilities
- zAAP on zIIP – allow 2:1 zIIP vs. CP now, and NO zAAPs on z13's and up
  - APAR OA38829 for z/OS 1.12 and 1.13 allows zAAP on zIIP even if there is a zAAP installed

More zIIP in DB2 11

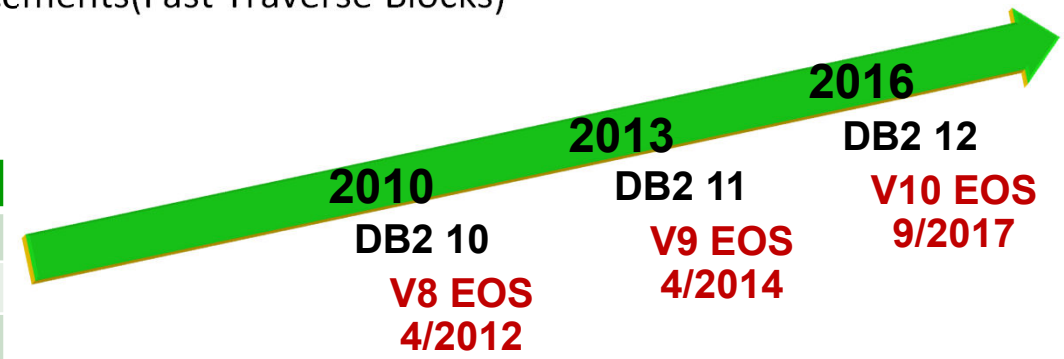
- Finally the majority of RUNSTATS (DSTATS), as well as INLINE stats
- 100% of delete processing with LOAD REPLACE
- Q REP: decompress and decode operations of capture process
- Index pseudo delete engine time will show up under the DBM1 SRB (PREEMPT IIP SRB)
- DPSI parallelism agents (PARAMDEG_DPSI)
- Log write I/O and log prefetch (MSTR) all go to the zIIP Roughly 10-20% of MSTR SRB
  - DBM1 saw another 10-15% additional zIIP offload (larger for heavy data sharing)
    - GBP castout (300), GBP writes (300)
    - Already had prefetch engines (600), deferred write engines (300)

CPU TIMES	TCB TIME	PREEMPT SRB	NONPREEMPT SRB	CP CPU TIME	PREEMPT IIP SRB
SYSTEM SERVICES ADDRESS SPACE	7:53.094670	1:38:43.086689	3:18.683030	1:49:54.864388	13:22.349651
DATABASE SERVICES ADDRESS SPACE	2:04.784117	41:56.094613	13:20.062439	57:20.941169	15:10:25.381584
IRLM	0.119303	0.000007	32:05.263272	32:05.382582	0.000000
DDF ADDRESS SPACE	1:08.358258	2:08:39.436318	1:11.055976	2:10:58.850552	2:11:30.752914

DB2 12 and zIIP

- Expand use for parallelism – 100% of child tasks
- Increase portions of utility processing (RELOAD)
- DB2 Adaptor for z/OS Connect allows more mobile apps access
  - z/OS Connect runs in WebSphere Liberty Profile
- Less wasteful prefetch, less chance of prefetch disabled due to no engine
- In-memory bufferpool enhancements(Fast Traverse Blocks)
- DRDA FastLoad

Version	GA
V7	3/2001
V8	3/2004
V9	3/2007
V10	10/2010
V11	10/2013



DB2 12 ESP started in March, GA'ed on Oct 21st, 2016

Asynchronous I/O (V10+)

- In DB2 10 prefetch and deferred write are zIIP eligible
  - Increase due to index I/O parallelism/ index list prefetch for disorganized indexes/ access path changes/ more dynamic prefetch in V9,V10

```
DB2 VERSION: V8                                    SCOPE: MEMBER                                                    TO: 09/10

---- HIGHLIGHTS ----------------------------------------------------------------------------------------------------
INTERVAL START  : 09/09/11 05:30:01.83    SAMPLING START: 09/09/11 05:30:01.83    TOTAL THREADS      :    90.00
INTERVAL END    : 09/10/11 05:00:02.70    SAMPLING END  : 09/10/11 05:00:02.70    TOTAL COMMITS      :  6328.8K
INTERVAL ELAPSED:      23:30:00.864709    OUTAGE ELAPSED:             0.000000    DATA SHARING MEMBER:      N/A

CPU TIMES                              TCB TIME      PREEMPT SRB   NONPREEMPT SRB       TOTAL TIME  PREEMPT IIP SRB
-------------------------------  ---------------  ---------------  ---------------  ---------------  ---------------
SYSTEM SERVICES ADDRESS SPACE        1:39.995961         0.000000      3:25.079924      5:05.075886              N/A
DATABASE SERVICES ADDRESS SPACE      1:31.822012         0.000000  12:28:38.995808  12:30:10.817820         0.000000
IRLM                                    0.456105         0.000000      3:02.893287      3:03.349391              N/A
DDF ADDRESS SPACE                       2.730084  20:28:36.142998     30:35.615420  20:59:14.488502  19:33:32.868978

TOTAL                                3:15.004163  20:28:36.142998  13:05:42.584438  1 09:37:33.7316  19:33:32.868978
```

```
DB2 VERSION: V10                                   SCOPE: MEMBER                                                    TO: 11/11

---- HIGHLIGHTS ----------------------------------------------------------------------------------------------------
INTERVAL START  : 11/10/11 06:09:00.00    SAMPLING START: 11/10/11 06:09:00.00    TOTAL THREADS      :   290.00
INTERVAL END    : 11/11/11 06:06:00.00    SAMPLING END  : 11/11/11 06:06:00.00    TOTAL COMMITS      : 10749.2K
INTERVAL ELAPSED:      23:57:00.000072    OUTAGE ELAPSED:             0.000000    DATA SHARING MEMBER:      N/A

CPU TIMES                              TCB TIME      PREEMPT SRB   NONPREEMPT SRB       TOTAL TIME  PREEMPT IIP SRB
-------------------------------  ---------------  ---------------  ---------------  ---------------  ---------------
SYSTEM SERVICES ADDRESS SPACE        2:26.595613      2:14.698997        13.547515      4:54.842125              N/A
DATABASE SERVICES ADDRESS SPACE      1:04.360185   5:49:17.448125        11.274434   5:50:33.082744   4:25:03.509555
IRLM                                    0.032864         0.000000      3:39.871402      3:39.904266              N/A
DDF ADDRESS SPACE                       6.096981  2 22:30:18.7722     56:23.794572  2 23:26:48.6638  1 11:39:09.8193

TOTAL                                3:37.085643  3 04:21:50.9193   1:00:28.487923  3 05:25:56.4929  1 16:04:13.3288
```

Asynchronous I/O (V10+)…

- Index I/O Parallelism for updates
  - If there are more than 2 indexes on a table (clustering index does not count) or 2 if the table is defined with APPEND, HASH, or MEMBER CLUSTER
    - DB2 detects an I/O delay we use sequential prefetch engine to do the I/O for each index leaf page in parallel
    - zParm INDEX_IO_PARALLELISM=YES (default) controls it
  - You will see S.PRF.PAGES READ/S.PRF.READ = 1.00 in the statistics report for index buffer pools
    - Use IFCID 357-358 to trace it

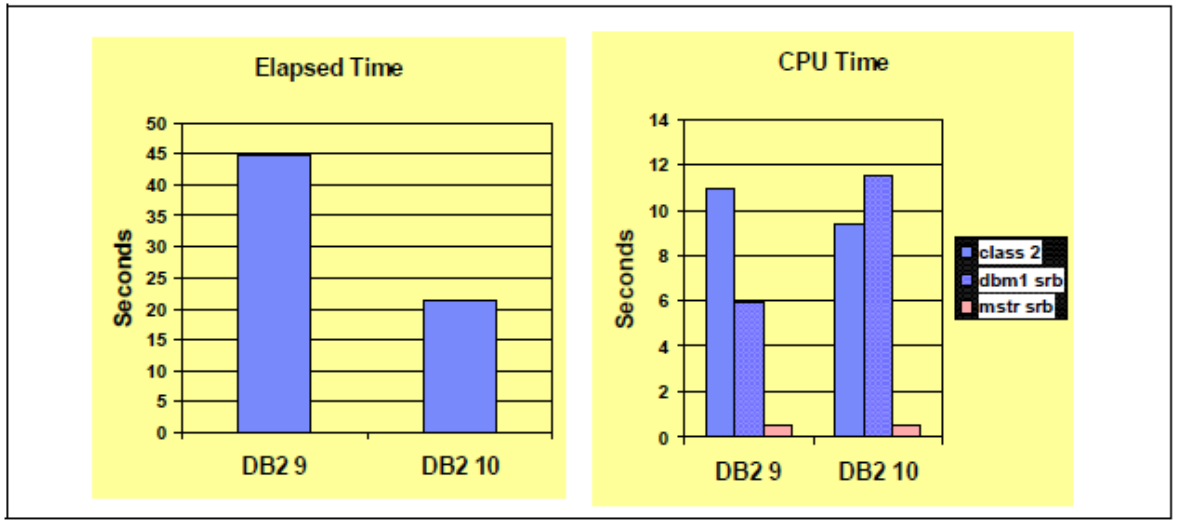


Figure 2-14 Insert index I/O parallelism

Asynchronous I/O (V10+)…

- What happens if Engines are starved of zIIP?
  - Other Read / Write I/O events and time per event will increase
  - PREF. DISABLED – NO READ ENG could increase
  - SYNC SEQ I/O / Sync Write
- Customers have seen batch programs miss their timing windows
- Even if prefetch <u>is not</u> used, may try to schedule it, and app still sees delays with BP hit and no I/Os
  - Increased elapsed time/CPU

CLASS 3 SUSPENSIONS	AVERAGE TIME	AV.EVENT
LOCK/LATCH(DB2+IRLM)	0.060293	48.65
IRLM LOCK+LATCH	0.000465	0.10
DB2 LATCH	0.059829	48.54
SYNCHRON. I/O	28.298614	69721.17
DATABASE I/O	28.298426	69720.92
LOG WRITE I/O	0.000188	0.25
OTHER READ I/O	5.036911	4802.06
OTHER WRTE I/O	0.000000	0.00

TOT4K READ OPERATIONS	QUANTITY	/SECOND	/THREAD	/COMMIT
SEQUENTIAL PREFETCH READS	4472.3K	311.88	12.35	0.55
LIST PREFETCH REQUESTS	1874.3K	130.70	5.18	0.23
LIST PREFETCH READS	745.1K	51.96	2.06	0.09
DYNAMIC PREFETCH REQUESTED	119.0M	8301.34	328.82	14.74
DYNAMIC PREFETCH READS	16325.1K	1138.43	45.09	2.02
PREF.DISABLED-NO BUFFER	285.00	0.02	0.00	0.00
PREF.DISABLED-NO READ ENG	656.00	0.05	0.00	0.00
PAGE-INS REQUIRED FOR READ	811.9K	56.62	2.24	0.10

Asynchronous I/O (V12)…

- More prefetch engines are available for use
  - Moved from 600 to 900 engines per DB2 subsystem
    - Hidden ZPARM SPRMRDU controls the number
    - Still uses ESQA and some below the bar storage so don't go crazy
- Remove unnecessary prefetch scheduling in V12
  - Tracks Dyanamic Prefetch failures
    - If last 3 prefetch requests did not result in prefetch I/O
      - Disable dynamic prefetch in the pool
    - First Synchronous sequential I/O detected
      - Dynamic prefetch is re-enabled
  - Saves
    - zIIP cycles
    - Unnecessary other READ I/O class 3 delays
    - Prefetch disabled NO READ ENGINE
    - LC24 contention
      - ~30k requests/sec = ~ 10k latch/second

Customer has a 70:1 ratio of requests vs. scheduled prefetch

Dsnb414i	Dsnb414i
Dynamic Prefetch Requests	Dynamic Prefetch I/O
36,973,390	550,642

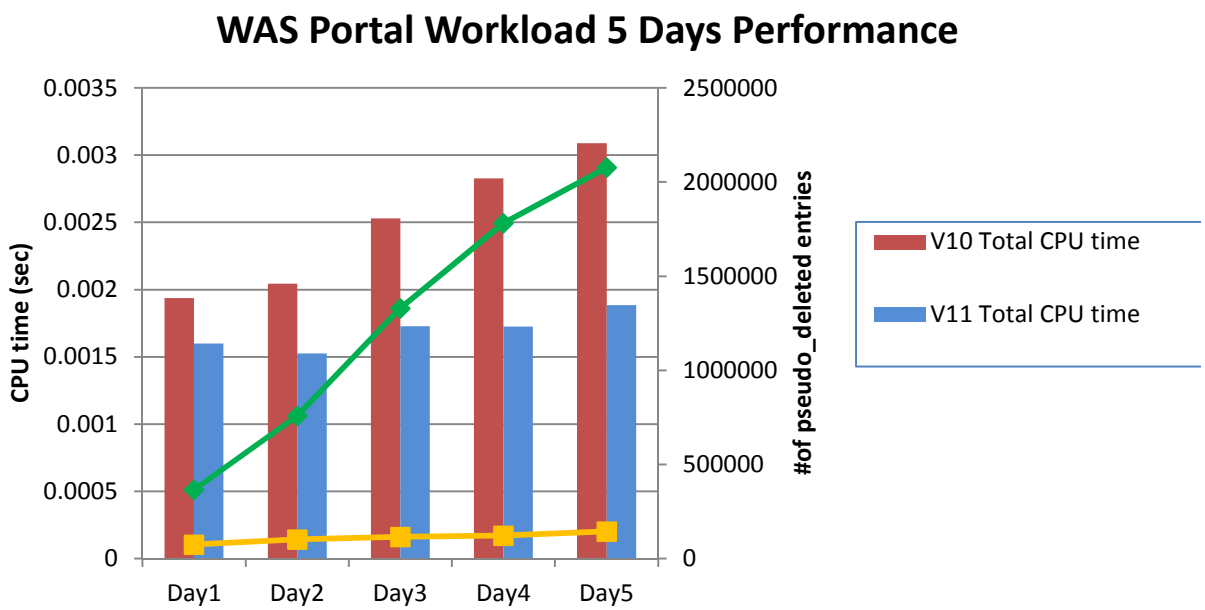
The next step in big data starts with IBM.

Automatic Pseudo Deleted Index Clean-up

- Autonomic solution provided in CM and turned on automatically for all indexes 24*7 – V11
  - Automatic clean-up of pseudo-deleted index entries in index leaf pages
  - Automatic clean-up of pseudo-empty index pages
  - Designed to have minimal or no disruption to concurrent DB2 work
  - Clean-up is done under system tasks, which run as enclave SRBs and are zIIP eligible
    - Parent thread (one per DB2 member) loops through RTS to find candidate indexes
    - Child clean-up threads only clean up an index if it already is opened for INSERT, UPDATE or DELETE on the DB2 member
- Clean-up is customizable
  - Can control the number of concurrent clean-up threads or disable the function using zparm INDEX_CLEANUP_THREADS
    - 0=Disable, 1-128, 10 is default
    - Monitor with IFCID 377
  - Entries in new Catalog table SYSIBM.SYSINDEXCLEANUP
    - Define when / which objects are to be considered in a generic way

Automatic Pseudo Deleted Index Clean-up…

- Up to 39% DB2 CPU reduction per transaction in DB2 11 compared to DB2 10
- Up to 93% reduction in Pseudo deleted entries in DB2 11
- Consistent performance and less need of REORG in DB2 11
- Avoid possible wasted
  - Getpages
  - I/Os
  - Prefetch
  - Deadlocks on insert trying to reuse deleted RID



EXPLOITATION

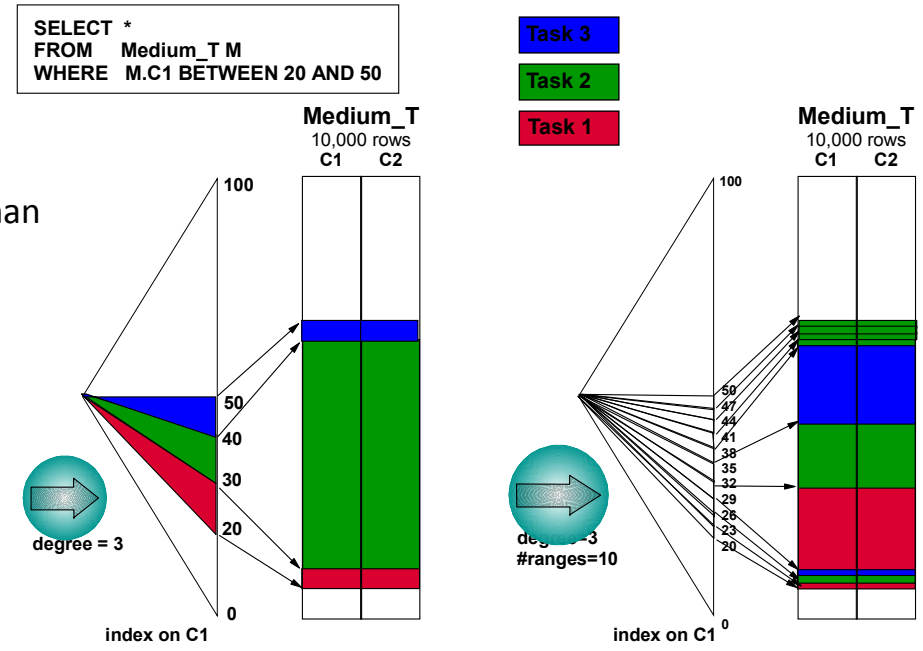
Parallelism offload %

- V8
  - Only Serial tasks cost out by optimizer
  - Parallelism cut on first table
  - limited 1x processors
- V9
  - Optimizer costs parallel tasks
  - Parallelism can be cut on inner table
  - Limited by 4x processors
- V10
  - Limited by 2x processors
- V11
  - Sysplex Query Parallelism is **removed**
  - DPSI parallelism
  - System negotiation based on storage
- V12
  - 100% of parallel child threads eligible
  - I/O parallelism **REMOVED**

If query uses this...	I/O parallelism	CP parallelism
Parallel access through RID list (list prefetch and multiple index access)	Yes	Yes
Materialized views or materialized table expressions at reference time	No	Yes
Security label column on table	Yes	Yes
Parallel access through IN-list	Yes	Yes

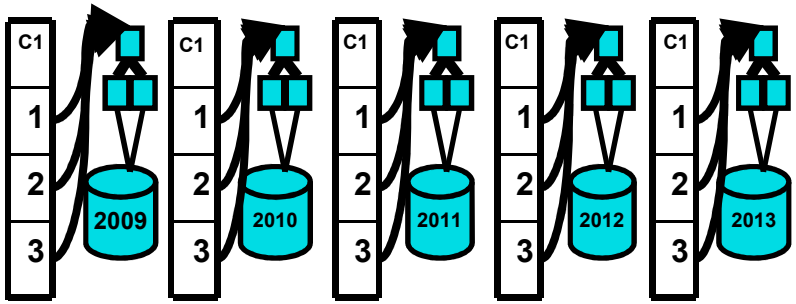
Parallelism…in V10

- Lifted many previous limitations
  - Multi-row fetch, reverse index scan, full outer joins, CTEs, CGTTs
- Improves parallel processing
  - DB2 10 may use the Straw Model workload distribution method
    - More key or page ranges will be allocated than the number of parallel degrees
    - Once a task finishes its smaller range it will process another range



Data Partitioned Secondary Indexes…V11

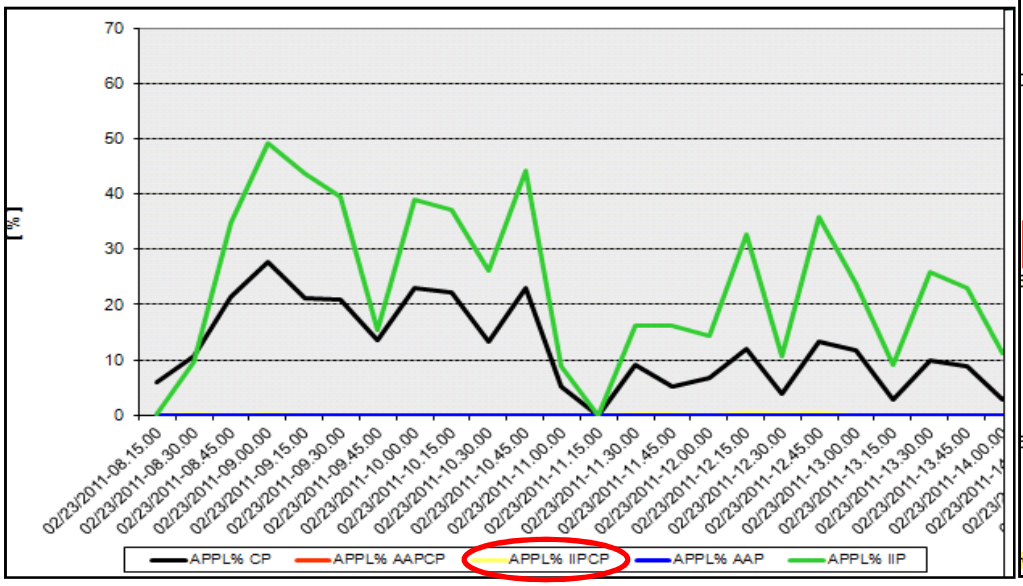
- Part Level Nested Loop Join
  - Parallelism cut on inner table on DPSI boundaries
    - One Child task process per DPSI part
  - Outer table shared in work file for child tasks
    - If less than 32K, uses In Memory Data Cache
  - Controlled by ZParm PARAMDEG_DPSI
  - EXPLAIN
    - PLAN_TABLE.PRIMARY_ACCESSTYPE=P
    - Various indications in DSN_PGROUP_TABLE
  - Can be hinted



- PARAMDEG_DPSI
  - Defaults to 0
    - PARAMDEG controls parallelism
  - 1= DPSI parallel task may be used but are run serially
  - 2 – 254= DPSI may drive parallel tasks up to this value
    - May be larger or smaller than PARAMDEG
    - Other portions of the query would be governed by PARAMDEG

Parallelism in production – COBOL Batch

- 80% of parallel child tasks are zIIP eligible (pre-V12) so it is the best way to affect zIIP Utilization %
  - Here we see there are no zIIP cycles that went to a GCP
  - 4CPs and 1 zIIP installed



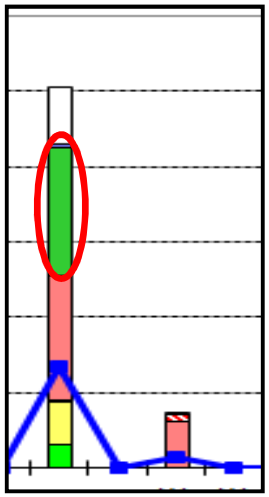
TIMES/EVENTS	APPL(CL.1)	DB2 (CL.2)
ELAPSED TIME	2:37:53.45	2:37:53.19
NONNESTED	2:37:53.45	2:37:53.19
STORED PROC	0.000000	0.000000
UDF	0.000000	0.000000
TRIGGER	0.000000	0.000000
CP CPU TIME	30:44.3617	30:44.3556
AGENT	17:38.9171	17:38.9111
NONNESTED	17:38.9171	17:38.9111
STORED PRC	0.000000	0.000000
UDF	0.000000	0.000000
TRIGGER	0.000000	0.000000
PAR.TASKS	13:05.4446	13:05.4446
SECP CPU	0.000000	N/A
SE CPU TIME	52:07.3400	52:07.3400
NONNESTED	0.000000	0.000000
STORED PROC	0.000000	0.000000
UDF	0.000000	0.000000
TRIGGER	0.000000	0.000000
PAR.TASKS	52:07.3400	52:07.3400
SUSPEND TIME	0.000000	47:44.0115
AGENT	N/A	29:21.9858
PAR.TASKS	N/A	18:22.0257
STORED PROC	0.000000	N/A
UDF	0.000000	N/A
NOT ACCOUNT.	N/A	58:44.9516

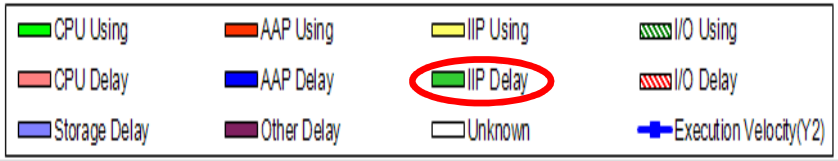
Parallelism Investigation

- RMF Spreadsheet Reporter Response delay report showed delay for zIIPs
  - Part of WLM Activity report
- Lots of unaccounted for time
  - OMPE accounting
  - Child task class 2 time not reported
- SYS1.PARMLIB (IEAOPTxx)
  - IIPHONORPRIORITY = **NO**
    - 3 parallel tasks waiting for 1 zIIP
  - In V11 this will stop system agents from being dispatched to the zIIP
  - Even with IIPHONORPRIORITY=YES Discretionary DDF work will queue for a zIIP and not fall back to a GCP
    - Blocked Workload Support not applicable

```
QUERY PARALLEL.                    TOTAL
-------------------              --------
MAXIMUM MEMBERS                       N/P
MAXIMUM DEGREE                          4
GROUPS EXECUTED                        78
 RAN AS PLANNED                        78
 RAN REDUCED                            0
 ONE DB2 COOR=N                         0
 ONE DB2 ISOLAT                         0
 ONE DB2 DCL TTABLE                     0
 SEQ - CURSOR                           0
 SEQ - NO ESA                           0
 SEQ - NO BUF                           0
 SEQ - ENCL.SER                         0
```

```
CLASS 2 TIME DISTRIBUTION
-------------------------------
CPU     |=====> 11%
SECPU   |
NOTACC  |==================> 37%
SUSP    |=========> 19%
```





CPU delay at about 33%, and the zIIP suspense time at 34%.

What to look for with parallelism

- DSNB440I - shows degraded parallel tasks from buffer pools
  - Basic ROT is VPSIZE*VPSEQT >=320MB
- ACCNT report/trace – Query parallelism section
  - Ran as Planned/Ran reduced
- DSNU397I - Utility message on constrained tasks (SORTNUM)
- -DISPLAY THREAD(*) – PT appears next to parallel tasks
- IFCID 0222 – OMEGAMON activity trace
  - Shows actual number of tasks and degradation
- IFCID 0221 – tells you which buffer pool restricted parallelism

QUERY PARALLELISM	AVERAGE	TOTAL
MAXIMUM DEGREE-ESTIMATED	0.00	0
MAXIMUM DEGREE-PLANNED	0.00	0
MAXIMUM DEGREE-EXECUTED	0.00	0
MAXIMUM MEMBERS USED	N/A	0
GROUPS EXECUTED	0.00	0
RAN AS PLANNED	0.00	0
RAN REDUCED-STORAGE	0.00	0
RAN REDUCED-NEGOTIATION	0.00	0
SEQ-CURSOR	0.00	0
SEQ-NO ESA SORT	0.00	0
SEQ-NO BUFFER	0.00	0
SEQ-ENCLAVE SERVICES	N/A	N/A
SEQ-AUTONOMOUS PROCEDURE	0.00	0
SEQ-NEGOTIATION	0.00	0
ONE DB2-COORDINATOR = NO	0.00	0
ONE DB2-ISOLATION LEVEL	0.00	0
ONE DB2-DCL TEMPORARY TABLE	0.00	0
MEMBER SKIPPED (%)	N/C	N/A
DISABLED BY RLF	0.00	0
REFORM PARAL-CONFIG	0.00	0
REFORM PARAL-NO BUF	0.00	0

```
DSNB440I  DB1S PARALLEL ACTIVITY -
            PARALLEL REQUEST =        7  DEGRADED PARALLEL=        0
```

What You Control for Parallelism..

- Hidden zParm SPRMPTH – DSN6SPRC
  - Threshold below which parallelism disabled
- PARAMDEG – MAX_DEGREE limits parallel groups
  - Static and dynamic SQL (default '0', unlimited)
- CDSSRDEF – SET CURRENT DEGREE special register for dynamic queries
  - Default =1, 'ANY' lets DB2 decide
- DEGREE(ANY) and CURRENTDATA(NO) bind options
  - Or DB2 needs to know if cursor is read-only
- VPPSEQT - % of sequential steal (VPSEQT) for parallel operations
  - Each utility task needs 128 pages in BP
- Star join enabled, number of tables involved
- PARA_EFF - % of cost reduction regarding parallel access path improvement (PM16020)

AccessPath	sequential_cost	parallel_degree	parallel_reduced_cost
AP1	1000	5	400
AP2	2000	20	300

PARMLIB Parameters

- IIPHONORPRIORITY (YES/NO) in IEAOPTxx parmlib member
  - This means if we reach the queue limit and ZIIPAWMT is triggered the dispatcher will route work over to a GP
  - If set to NO in DB2 11 then no system agents will be zIIP eligible
- ZIIPAWMT, ZAAPAWMT – Alternate wait management threshold is how long zIIP will run before checking to see if it needs help from GP

  Ask Level 2 before adjusting!
  - Default 12 milliseconds/ 3.2 for Hiperdispatch
  - In V10/V11 that means system engines may wait 3.2ms
- ZAAPZIIP = YES|NO (IEASYSxx option)
  - Allows zAAP eligible workload to run on a zIIP
- zAAP has other settings not applicable to zIIP
  - IFACrossover – disallow zAAP work on general CP

** Be careful about attempting to FORCE zIIP offload

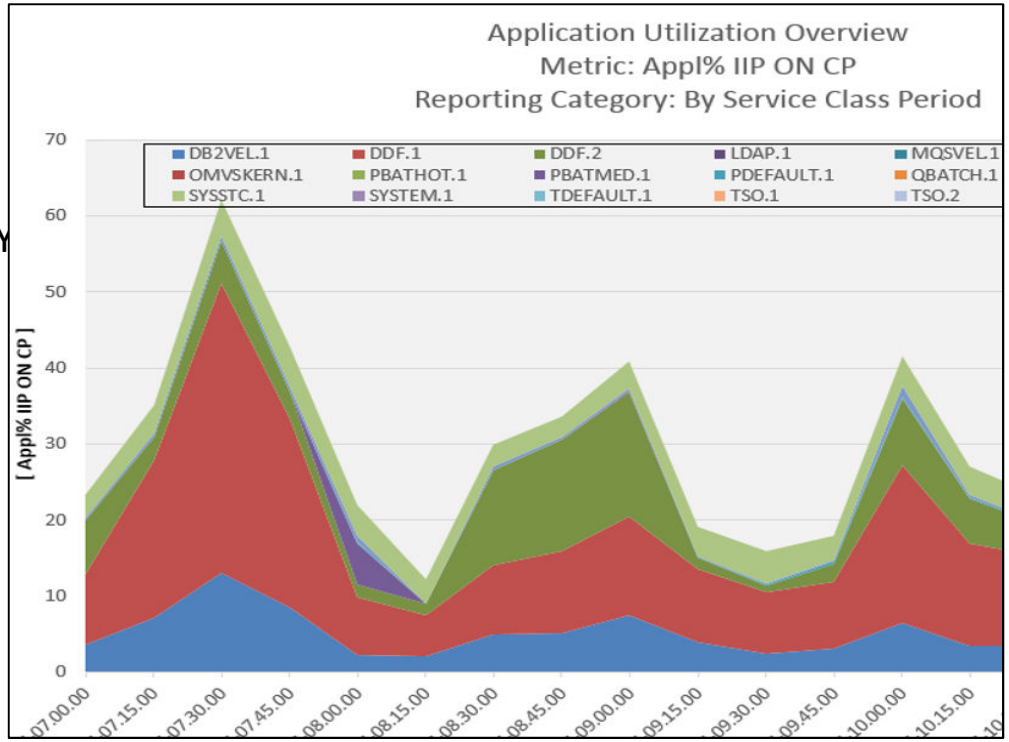
SMT (z13) Symmetric Multi-Threading

- SMT allows control program to run 1 or 2 threads concurrently on 1 CP
  - Can run parallel threads on 1 zIIP and IFL (not on CPs)
    - Z13 has 8 cores per GCP @ 5 GHz
  - If running parallel each task runs slower, but overall utilization is less
    - **IBM Brokerage OLTP workload showed 20% throughput improvement**
  - V12 has 1,800 system agent engines, hence throughput is key
- New IEAOPTxx parameter to control zIIP SMT mode
- MT_ZIIP_MODE=2 for 2 active threads (the default is 1)
  - Define a LOADxx PROCessor VIEW (PROCVIEW) ***CORE***|CPU for the life of the **IPL**
- Without an IPL you can change the zIIP processor class MT Mode (the number of active threads per online zIIP) using IEAOPTxx SET OPT=xx
- Requires HyperDispatch=YES

z13 zIIP capacity:
- *is 38% greater than a zEC12 zIIP*
- *is 72% greater than a z196 zIIP*

Summary

- Monitor zIIP overflow/redirect for capacity planning… not absolute utilization
- If the zIIP peak and your 4-hour rolling average collide… every MSU counts
- Use RMF Spreadsheet reporter to determine BY SERVICE CLASS which workloads are being hindered
  - ApplOvwTrd tab now included in spreadsheet
- Fewer faster zIIPs on an upgrade is not a good idea
- Review LPAR weightings to determine if zIIPs are parked during times of zIIP redirect
- Test SMT and monitor the zIIP redirect



Reference material

- II14219 - zIIP Exploitation
- Subsystem and Transaction Monitoring and Tuning with DB2 11 for z/OS SG24-8182
  - https://www.redbooks.ibm.com/Redbooks.nsf/RedbookAbstracts/sg248182.html?Open
- Techline Sizing with zCP3000 tool: contact your local IBMer
- RMF Spreadsheet Reporting Tool
  - http://www-03.ibm.com/systems/z/os/zos/features/rmf/tools/rmftools.html
- Getting Started Resources
  - http://www-03.ibm.com/systems/z/hardware/features/ziip/resources.html
- Link to article on PARMLIB settings
  - https://www.ibm.com/developerworks/mydeveloperworks/blogs/22586cb0-8817-4d2c-ae74-0ddcc2a409bc/entry/december_17_2012_6_07_am3?lang=en