



Systems & Technology Group

Connecting the Dots: LPARs, HiperDispatch, zIIPs and zAAPs

Share in Boston, August 2010

Glenn Anderson
IBM Technical Training
grander@us.ibm.com



© 2010 IBM Corporation

What I hope to cover.....

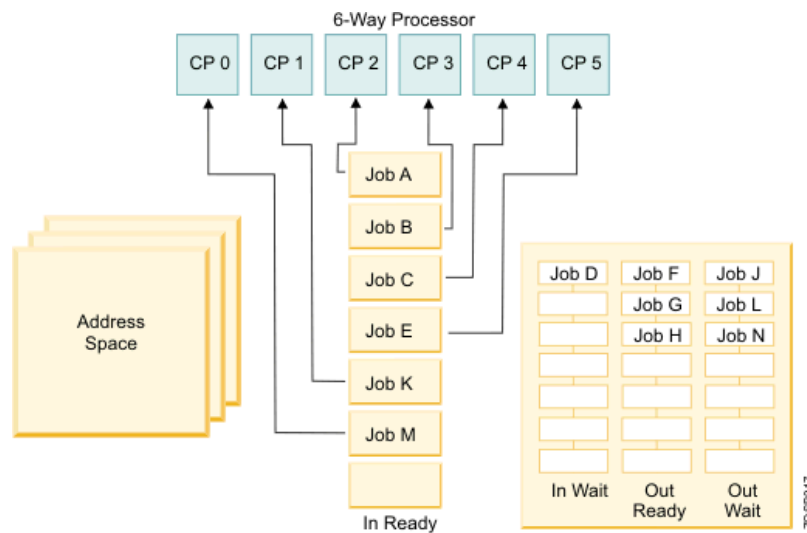
- What are dispatchable units of work on z/OS
- Understanding Enclave SRBs
- How WLM manages dispatchable units of work
- The role of HiperDispatch
- What makes work eligible for zIIP and zAAP specialty engines
- Dispatching work to zIIP and zAAP engines

z/OS Dispatchable Units

- **There are different types of Dispatchable Units (DU's) in z/OS**

- Preemptible Task (TCB)
- Non Preemptible Service Request (SRB)
- Preemptible Enclave Service Request (enclave SRB)
 - Independent
 - Dependent
 - Workdependent

z/OS Dispatching Work

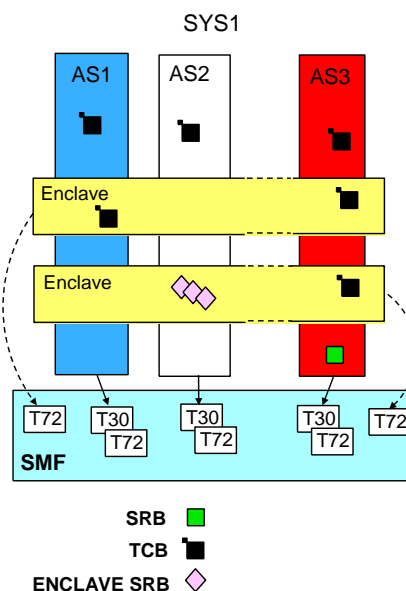


Enclave Services: A Dispatching Unit

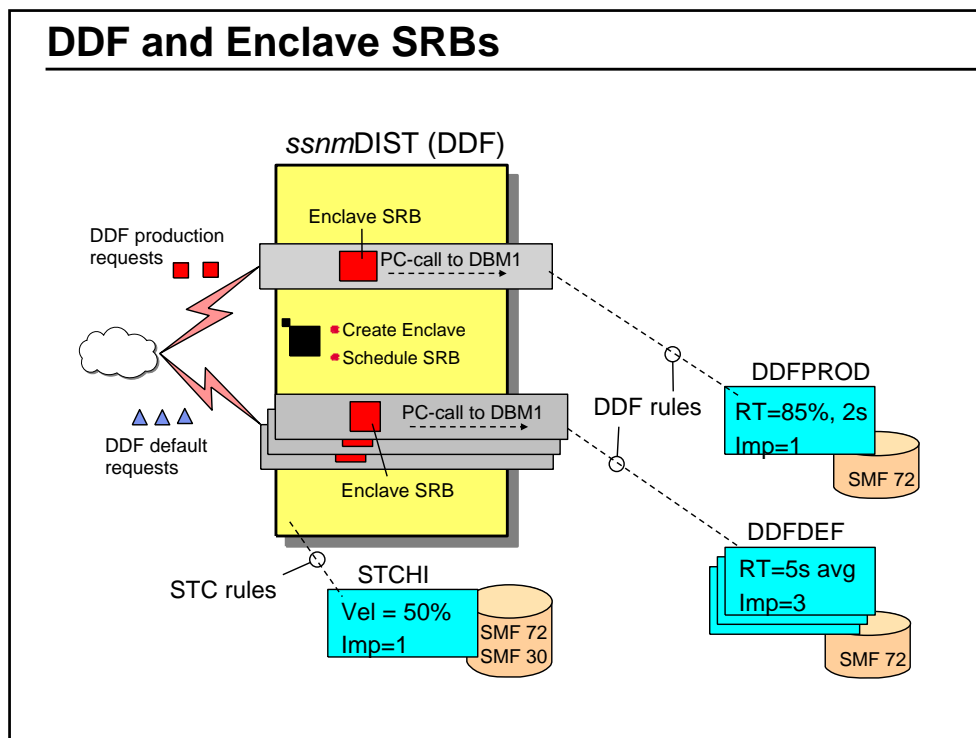
- **Standard dispatching**
 - dispatchable units (DUs) are the TCB and the SRB
 - TCB runs at dispatching priority of address space and is pre-emptible
 - SRB runs at supervisory priority and is non-pre-emptible
- **Advanced dispatching units**
 - **Enclave**
 - Anchor for an address space-independent transaction managed by WLM
 - Can comprise multiple DUs (TCBs and Enclave SRBs) executing across multiple address spaces
 - **Enclave SRB**
 - Created and executed like an ordinary SRB but runs with Enclave dispatching priority and is pre-emptible
- **Enclave Services enable a workload manager to create and control enclaves**

Enclave Characteristics

- Created by an address space (the "owner")
- One address space can own many enclaves
- One enclave can include multiple dispatchable units (SRBs/tasks) executing concurrently in multiple address spaces (the "participants")
 - Enclave SRBs are preemptible, like tasks
 - All its dispatchable units are managed as a group
- Many enclaves can have dispatchable units running in one participant address space concurrently
- RMF produces separate T72 SMF records for independent enclaves



DDF and Enclave SRBs



What is a WLM Transaction?

- **A WLM transaction represents a WLM "unit of work"**
 - basic workload entity for which WLM collects a resource usage value
 - foundation for statistics presented in workload activity report
 - represents a single subsystem "work request"
- **Subsystems can implement one of three transaction types**
 - **Address Space:**
 - WLM transaction measures all resource used by a subsystem request in a **single address space**
 - Used by JES (a batch job), TSO (a TSO command), OMVS (a process), STC (a started task) and ASCH (single APPC program)
 - **Enclave:**
 - Enclave created and destroyed by subsystem for each work request
 - WLM transaction measures resources used by a single subsystem request across **multiple address spaces and systems**
 - Exploited by subsystems - Component Broker(WebSphere), DB2, DDF, IWEB, MQSeries Workflow, LDAP, NETV, TCP
 - **CICS/IMS Transactions**
 - Neither address space or enclave oriented - special type
 - WLM transaction measures resource used by a single CICS/IMS transaction program request

Service Class with Enclave Transactions

```

REPORT BY: POLICY=WLMPOLO1  WORKLOAD=WAS  SERVICE CLASS=WI180%01  RESOURCE GROUP=*NONE  PERIOD=1  IMPORTANCE=1
          CRITICAL              =NONE

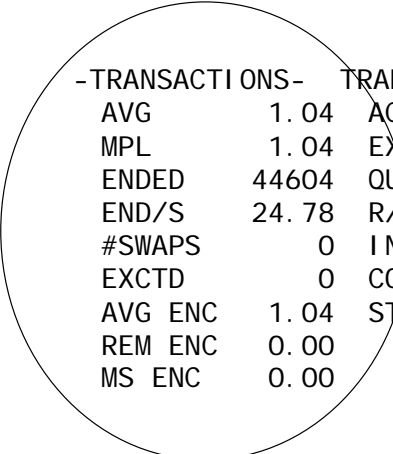
-TRANSACTIONS-  TRANS-TIME HHH.MM.SS.TTT  --DASD I/O--  ---SERVICE---  SERVICE TIME  ---APPL %---  --PROMOTED--  ---STORAGE---
AVG      1.04  ACTUAL                43  SSCHRT  0.0  IOC      0  CPU  225.586  CP   9.36  BLK  0.000  AVG  0.00
MPL      1.04  EXECUTION                41  RESP   0.0  CPU    62663K  SRB  0.000  AAPCP 0.13  ENQ  0.000  TOTAL 0.00
ENDED   44604  QUEUED                1  CONN   0.0  MISO    0  RCT  0.000  I1PCP 0.00  CRM  0.000  SHARED 0.00
END/S    24.78  R/S AFFIN                0  DISC   0.0  SRB    0  IIT  0.000  LCK  0.000
#SWAPS   0  INELIGIBLE                0  Q-PEND 0.0  TOT    62663K  HST  0.000  AAP   3.18  -PAGE-IN RATES-
EXCTD    0  CONVERSION                0  IOSQ   0.0  /SEC   34813  AAP  57.172  I1P   0.00  SINGLE  0.0
AVG ENC  1.04  STD DEV                135  ABSRPTN 34K  I1P   0.000  BLOCK  0.0
REM ENC  0.00  TRX SERV 34K  SHARED 0.0
MS ENC   0.00  HSP 0.0

RESP ----- STATE SAMPLES BREAKDOWN (%) ----- STATE-----
SUB P  TIME --ACTIVE-- READY IDLE -----WAITING FOR----- SWITCHED SAMPL(%)
TYPE (%)  SUB APPL  TYP3  LOCAL SYSPL REMOT
CB  BTE  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0
CB  EXE  93.6  1.5  98.3  0.0  0.0  0.3  0.0  0.0  0.0

GOAL: RESPONSE TIME 000.00.01.000 FOR 80%

RESPONSE TIME EX  PERF  AVG  --EXEC USING--  EXEC DELAYS %  -USING-  --- DELAY % --- %
SYSTEM  ACTUAL%  VEL%  INDX  ADRSP  CPU  AAP  I1P  I/O  TOT  CPU  AAP  0  CRY  CNT  UNK  I DL  CRY  CNT  QUI
JCO      100    78.7  0.5  0.9  11  2.5  0.0  0.0  3.7  2.3  1.2  0.1  0.0  0.0  83  0.0  0.0  0.0  0.0
  
```

Service Class with Enclave Transactions

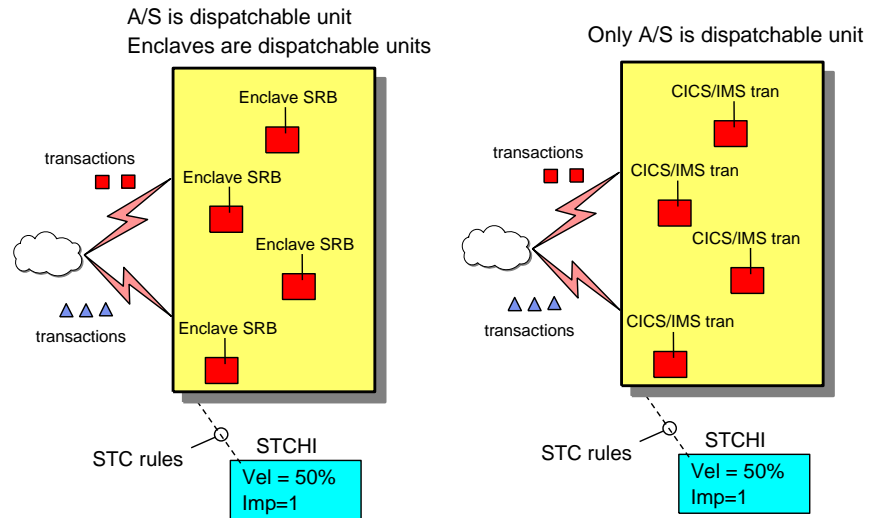


```

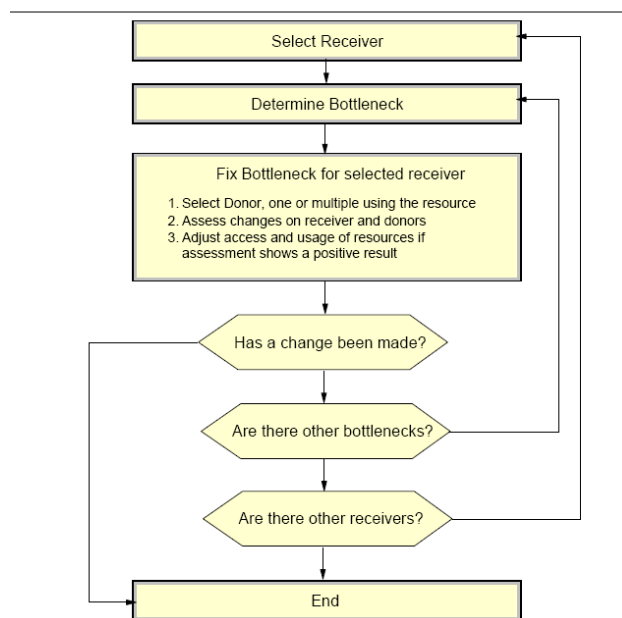
-TRANSACTIONS-  TRANS-TIME HHH.MM.SS.TTT
AVG      1.04  ACTUAL                43
MPL      1.04  EXECUTION                41
ENDED   44604  QUEUED                1
END/S    24.78  R/S AFFIN                0
#SWAPS   0  INELIGIBLE                0
EXCTD    0  CONVERSION                0
AVG ENC  1.04  STD DEV                135
REM ENC  0.00
MS ENC   0.00
  
```

The WLM View

Address Spaces, and the transactions inside



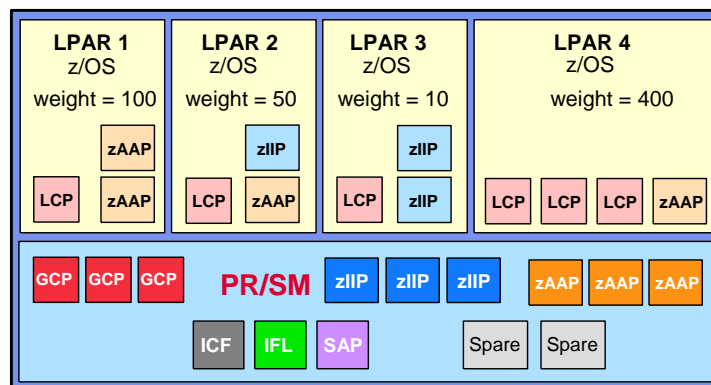
WLM Policy Adjustment Algorithm



WLM Dispatching Priority Usage

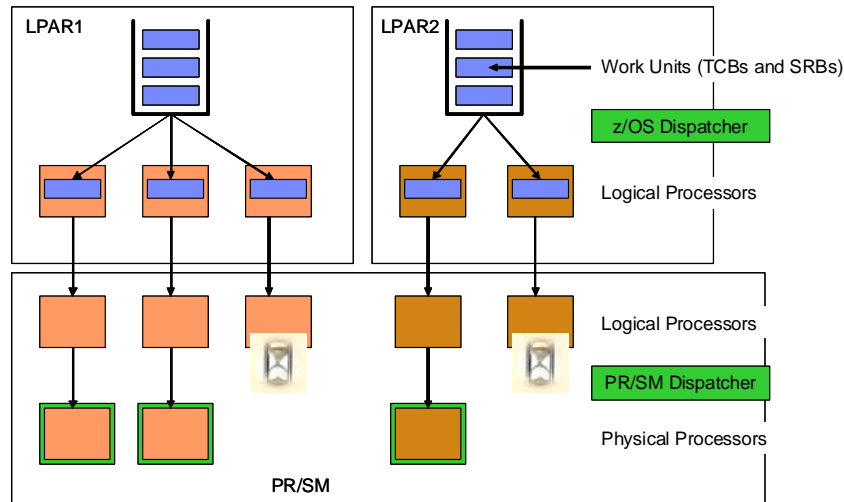
255	SYSTEM
254	SYSSTC
253	<i>Small Consumer</i>
252	Priorities for dynamic policy adjustment
208	
207	
202	Not used
201	
192	Discretionary work Mean Time to wit algorithm

Central Processors in a CEC



- GCPs and Specialty CPs in a CEC
- PR/SM dispatches specialty CPs in the same manner as GCP
 - Managed to weight and number of logical specialty CPs
 - CPs can be shared or dedicated

Dispatching in an LPAR Environment



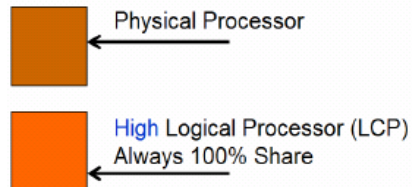
HiperDispatch Introduction

- Design Objective
 - Keep work as much as possible local to a physical processor to optimize the usage of the processor caches
 - As a result systems with high number of physical processors provide a much better scalability
- Function: HiperDispatch
 - Interaction between z/OS and the PR/SM Hypervisor to optimize work unit and logical processor placement to physical processors
 - Consists of 2 parts
 - In z/OS (sometimes referred as Dispatcher Affinity)
 - In PR/SM (sometimes referred as Vertical CPU Management)

HiperDispatch: Processor Share

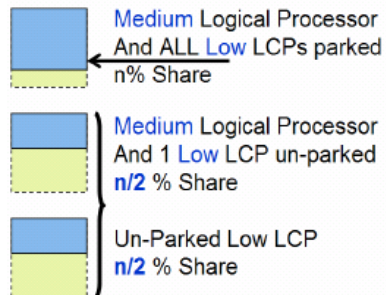
- **High Logical Processor**

- Always 100% Share
- That means
 - Always re-dispatched to its physical processor whenever it has demand



- **Medium and Low Processors**

- Divide the share of the medium processors between them
- That means
 - The share decreases per processor when more low processors become un-parked

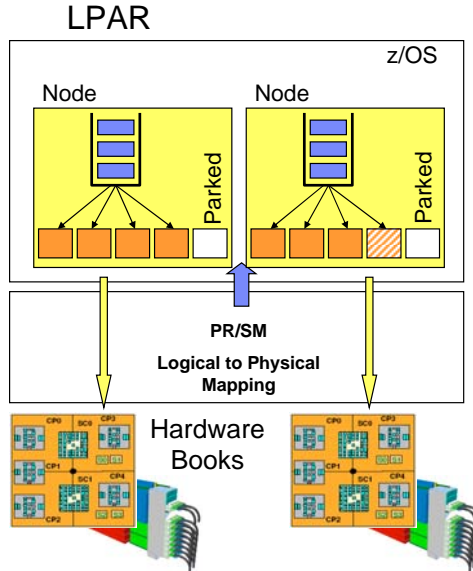


HiperDispatch Mode

- PR/SM
 - Supplies topology information/updates to z/OS
 - Ties *high priority* logicals to physicals (gives 100% share)
 - Distributes remaining share to *medium priority* logicals
 - Distributes any additional service to un-parked *low priority* logicals
- z/OS
 - Ties tasks to small subsets of logical processors
 - Dispatches work to *high priority* subset of logicals
 - Parks *low priority* processors that are not need or will not get service
- **Hardware cache optimization occurs when a given unit of work is consistently dispatched on the same physical CPU**

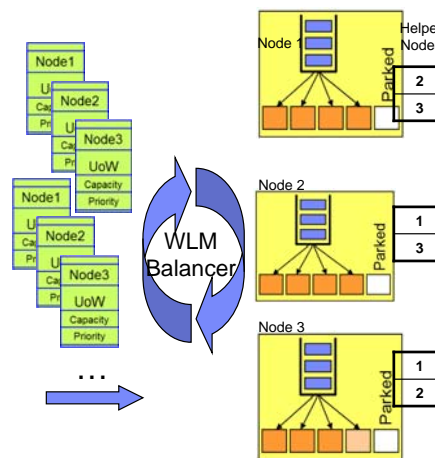
HiperDispatch: z/OS Part

- z/OS obtains the logical to physical processor mapping in Hiperdispatch mode
 - Whether a logical processor has high, medium or low share
 - On which book the logical processor is located
- z/OS creates dispatch nodes
 - The idea is to have 4 high share CPUs in one node
 - Each node has TCBs and SRBs assigned to the node
 - Optimizes the execution of work units on z/OS



HiperDispatch: z/OS Affinity Dispatching

- Affinity dispatching
 - WLM balances the units of work (TCBs/SRBs) across the nodes to equalize the utilization of nodes
 - And to assure that each node has work of different priorities
 - Balancing takes place every 2 seconds
 - For unbalanced situations in between
 - WLM creates lists of helper nodes for each node
 - These helper nodes can be asked to select work from a given node in cases the node is overloaded
 - Helper nodes are sorted to avoid book crossing if possible
- Low share processors
 - WLM parks and un-parks these processors based on demand and utilization of the CEC



HiperDispatch User Interface: RMF

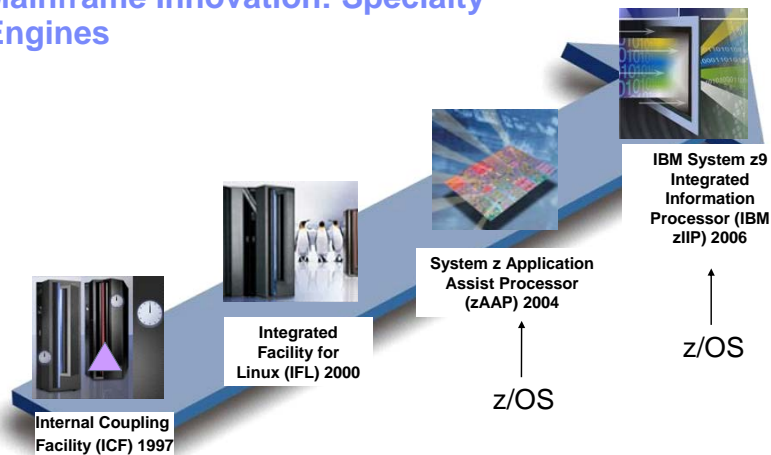
CPU ACTIVITY

z/OS V1R8 SYSTEM ID IT2I DATE 05/07/2008
RPT VERSION V1R8 RMF TIME 10.53.07

-CPU	2097	MODEL	716	H/W	MODEL	E56	SEQUENCE	CODE	00000000000969D0	HIPERDISPATCH=	YES
0---CPU---					TIME %					--I/O INTERRUPTS--	
NUM	TYPE	ONLINE	LPAR	BUSY	MVS	BUSY	PARKED	SHARE	%	RATE	% VIA TPI
0	CP	100.00	94.85	100.0	100.0	0.00	100.0	100.0	1123	0.17	
1	CP	100.00	95.19	100.0	100.0	0.00	100.0	100.0	1095	0.16	
...											
7	CP	100.00	78.09	99.99	0.00	99.5	0.00	0.00	0.00	0.00	
8	CP	100.00	0.10	----	100.00	0.0	0.00	0.00	0.00	0.00	
9	CP	100.00	0.09	----	100.00	0.0	0.00	0.00	0.00	0.00	
A	CP	100.00	78.23	99.99	0.00	0.0	0.00	0.00	0.00	0.00	
B	CP	100.00	0.08	----	100.00	0.0	0.00	0.00	0.00	0.00	
C	CP	100.00	0.08	----	100.00	0.0	0.00	0.00	0.02	0.00	
TOTAL/AVERAGE			63.61	100.0				799.5	8827	0.17	
0 D	IIP	100.00	57.83	57.83	0.00	100.0					
TOTAL/AVERAGE			57.83	57.83				100.0			

System z Specialty Engines

Mainframe Innovation: Specialty Engines



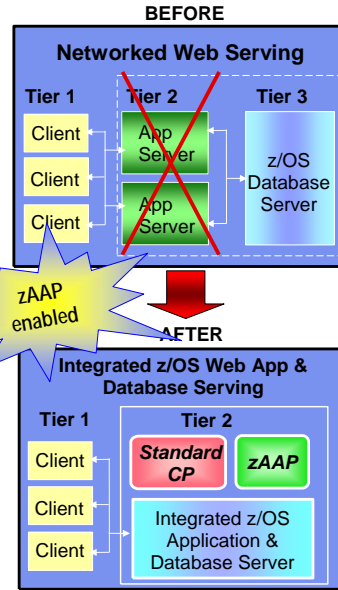
zAAP Processor for Increased e-business Integration and Infrastructure Simplification

The System z Application Assist Processor

- an "assist" processor dedicated exclusively to execution of Java workloads under z/OS
 - includes WebSphere, DB2, CICS
- available on IBM System z10, z9 and zSeries z990/z890
- Exploited with z/OS V1.6 and IBM Java SDK
- executes Java code with no changes to applications
- priced, much lower than standard CPs
- lower maintenance costs than standard CPs
- IBM zSeries software charges unaffected
- can reduce Sub-capacity IBM software charges
- up to 1 zAAP per general CP in a CEC

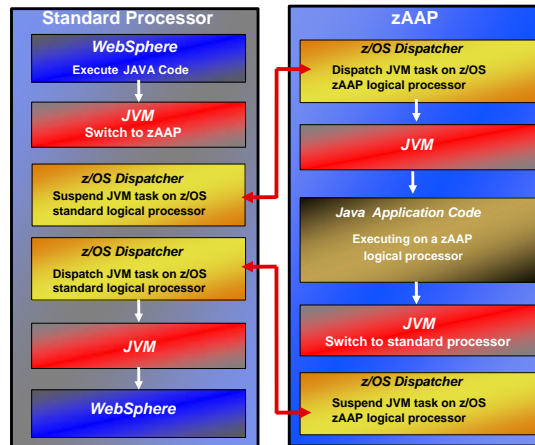
Benefits of zAAPs

- help to consolidate, simplify and reduce server infrastructure
- can eliminate application server processing tier
- leverage core zSeries strengths to manage Java Workloads automatically on z/OS



zAAP Workflow: Executing Java under IBM JVM control

- IBM JVM, parts of LE runtime, and z/OS Supervisor are needed to support JVM execution on zAAPs
- IBM JVM communicates to z/OS dispatcher when Java code is to be executed
- When Java is to be executed, the work unit is "eligible" to be dispatched on a zAAP
- zAAP ineligible work is only dispatched on general purpose processors



IBM System z Integrated Information Processor (zIIP)

Specialty assist processor dedicated exclusively to execution of any workloads under z/OS®

- Specialty engine for the IBM System z9 and IBM System z10 mainframe
- Requires work to be run as an enclave SRB
- Exploiters work with z/OS to determine how much capacity should be redirected to a zIIP
- Exploiters must use licensed interface to enable exploitation

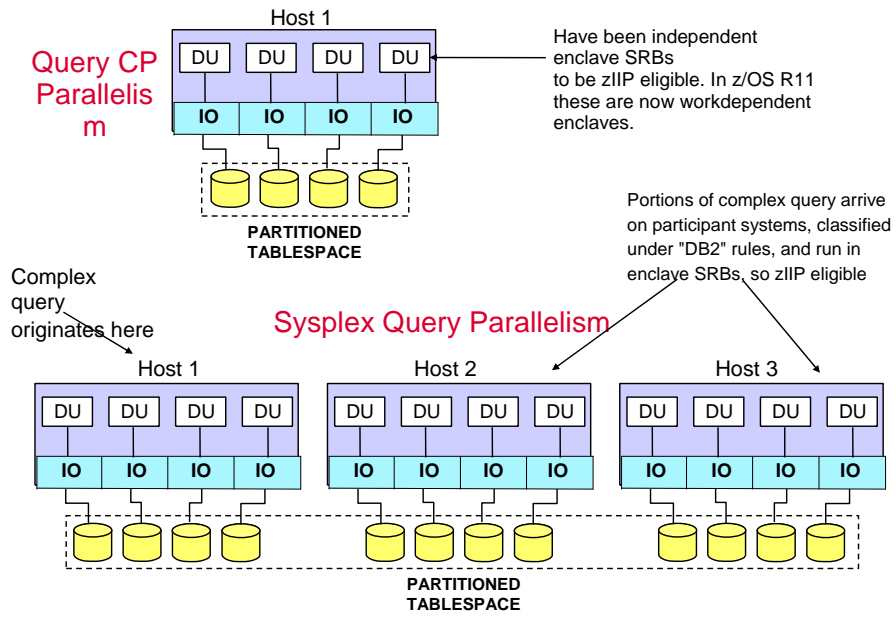


DB2 V8 Exploitation of IBM zIIP

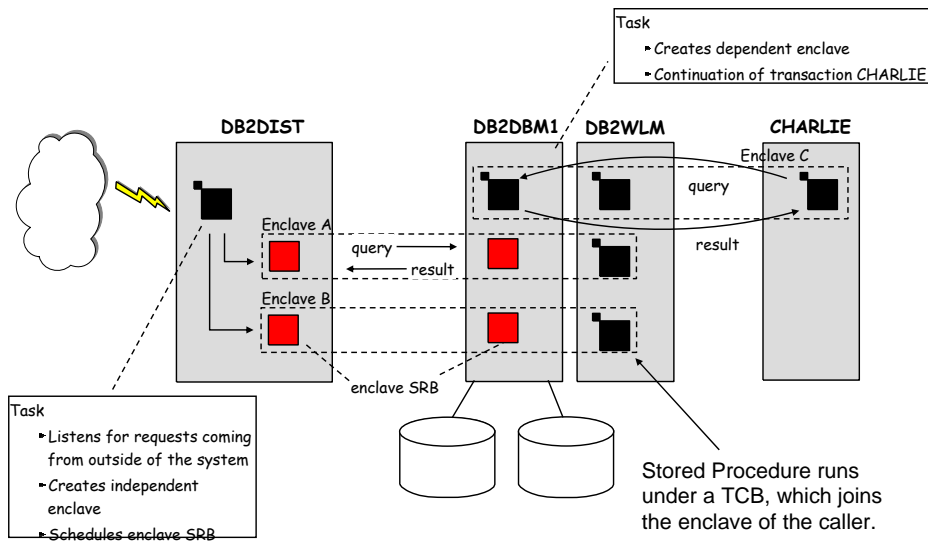
- DB2 for z/OS V8 was the first IBM exploiter of the zIIP but requires:
 - IBM System z9 or z10 processor
 - z/OS 1.6 or later
 - DB2 for z/OS V8 or V9 (either compat mode or full-function mode)
- Portions of the following DB2 for z/OS workloads may benefit from zIIP*
 - Via DRDA over a TCP/IP connection
 - ERP, CRM, Business Intelligence or other enterprise applications
 - Requests that utilize DB2 complex parallel queries
 - Data warehousing applications*
 - DB2 for z/OS utilities*
 - Internal DB2 utility functions used to maintain index maintenance structures

* The zIIP is designed so that a program can work with z/OS to have all or a portion of its enclave Service Request Block (SRB) work directed to the zIIP. The above types of DB2 V8 work are those executing in enclave SRBs, of which **portions** can be sent to the zIIP.

DB2 Parallel Query and Enclave SRBs

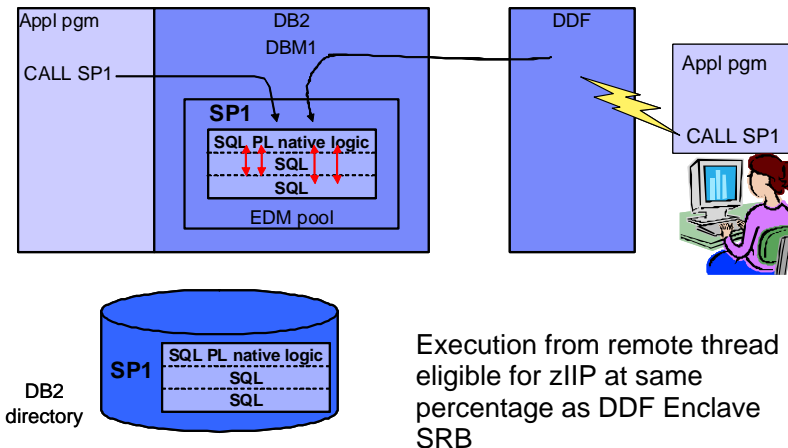


DB2 Stored Procedures



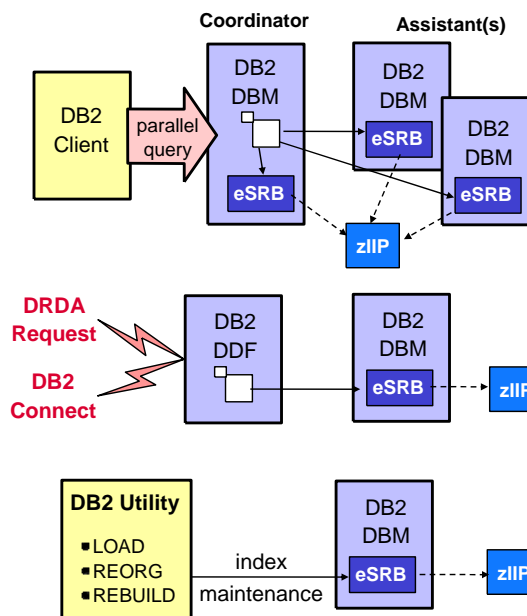
Native SQL Procedures (DB2 Ver 9)

The SQL procedure logic runs in the DBM1 address space



zIIP Processor Workflow Characteristics

- Enclave SRBs are used to:
 - Schedule pre-emptable work
 - Associate multiple threads with same work unit and WLM policy
- DB2 runs work under:
 - Tasks
 - Client SRBs
 - Enclave SRBs
- Some DB2 **enclave SRB** work is zIIP capable
 - Complex parallel query processing
 - DRDA requests over TCP/IP
 - Index maintenance for some utilities
- When DB2 schedules an enclave SRB, it identifies to z/OS dispatcher which eSRBs are zIIP capable



How is IPsec Enabled for zIIP?

- The z/OS Communication Server interacts with z/OS WLM to have all of its enclave SRB work eligible for a zIIP.
 - A single config statement in the TCP/IP profile triggers the CS to request z/OS to direct this Enclave SRB processing to an available zIIP
 - GLOBALCONFIG ZIIP **IPSECURITY** / **NOIPSECURITY**(default)
 - Independent enclave allows the work to be classified to a unique service class to influence IPsec access to zIIPs and general purpose CPs
 - New WLM subsystem type TCP in classification rules
 - Enclave exists for life of TCP/IP stack, so use a period service class with a velocity goal

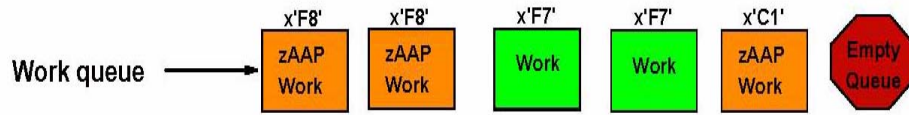


IEAOPTxx Specialty CP Parameters

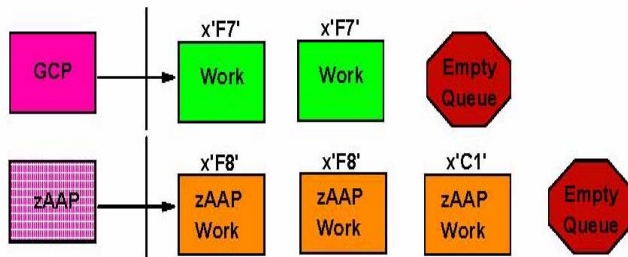
- What to do when the zIIP or zAAP needs help (ready work is waiting on the zIIP or zAAP dispatching queue)

IFAHONORPRIORITY	GCP	zAAP
YES	Process non-zAAP work in priority order including zAAP work in priority order when the zAAP needs help	Process only zAAP work in priority order
NO	Process no zAAP work. Do not process zAAP work even if the zAAP needs help	Process only zAAP work in priority order
IIPHONORPRIORITY	GCP	zIIP
YES	Process non-zIIP work in priority order including zIIP work in priority order when the zIIP needs help	Process only zIIP work in priority order
NO	Process no zIIP work. Do not process zIIP work even if the zIIP needs help	Process only zIIP work in priority order

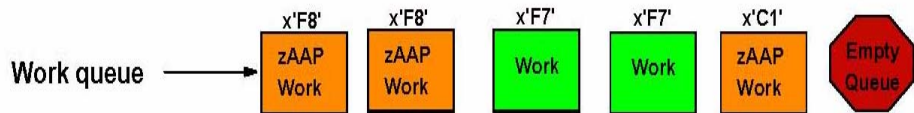
z/OS Dispatcher and zAAP Doesn't Need Help



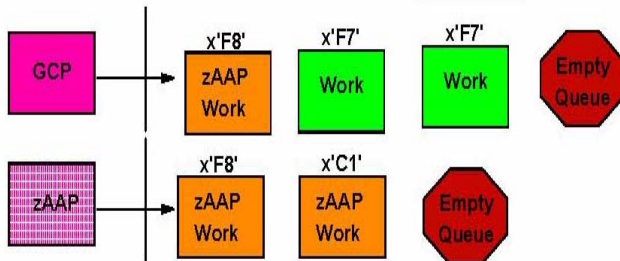
IFAHONORPRIORITY=YES and zAAP [doesn't need help](#)



z/OS Dispatcher and zAAP Engine Needs Help



IFAHONORPRIORITY=YES and zAAP [needs help](#)



Workload Activity Report with zIIP

CPU ACT Usage										
z/OS V1R7			SYSTEM ID SYSC		START 07/07/2006					
			RPT VERSION V1R7 RMF		END 07/07/2006					
CPU 2094	MODEL 750	H/W MODEL S54								
---CPU---	ONLINE TIME	LPAR BUSY	MVS BUSY	CPU SERIAL	I/O TOTAL					
NUM	TYPE	PERCENTAGE	TIME PERC	TIME PERC	NUMBER	INTERRUPT RATE				
0	CP	100.00	63.76	63.76	07B10E	5923				
1	CP	100.00	57.48	57.48	07B10E	4697				
2	CP	100.00	60.67	60.67	07B10E	4971				
3	CP	100.00	40.95	40.95	07B10E	119.3				
CP	TOTAL/AVERAGE		55.72	55.72		15710				
4	IIP	100.00	81.11	81.11	07B10E					
7	IIP	100.00	88.52	88.52	07B10E					
IIP	AVERAGE		84.81	84.81						

REPORT BY: POLICY=WLPOL	WORKLOAD=WAS_WKL	SERVICE CLASS=CISDDF	RESOURCE GROUP=*NONE								
CRITICAL =NONE											
TRANSACTIONS	TRANS-TIME	HHH.MM.SS.TTT	--DASD I/O--	---SERVICE---	SERVICE TIMES	---APPL %---					
AVG	37.87	ACTUAL	19	SSCHRT 11070	IOC	0	CPU 1895.4	CP	160.73		
MPL	37.87	EXECUTION	19	RESP	2.2	CPU	50884K	SRB	0.0	AAPCP	0.00
ENDED	1149619	QUEUED	0	CONN	0.3	MSO	0	RCT	0.0	IIPCP	15.57
END/S	1916.04	R/S AFFIN	0	DISC	1.8	SRB	0	IIT	0.0		
#SWAPS	0	INELIGIBLE	0	Q+PEND	0.2	TOT	50884K	HST	0.0	AAP	0.00
EXCTD	0	CONVERSION	0	IOSQ	0.0	/SEC	84807	AAP	0.0	IIP	155.18
AVG ENC	37.87	STD DEV	122					IIP	931.1		

SYSTEM	RESPONSE TIME	EX	PERF	AVG	-----	USING%	-----	-----	EXECUTION DELAYS	%		
	HHH.MM.SS.TTT	VEL%	INDX	ADRSP	CPU	AAP	IIP	I/O	TOT	IIP	I/O	CPU
SYSC	000.00.00.019	29.3	1.0	37.8	2.8	0.0	2.8	7.8	14.6	12.9	1.0	0.6

Projecting zAAP and zIIP Usage with RMF

- Using RMF to Project zAAP / zIIP by Workload

- enabled with PROJECTCPU parm in IEAOPTxx

SERVICE TIMES	---	APPL	%---	
CPU	1905.9	CP	232.70	
SRB	0.0	AAPCP	0.00	the work in this service class would keep a zIIP engine 83% busy
RCT	0.0	IIPCP	83.26	
IIT	0.0			
HST	0.0	AAP	0.00	
AAP	0.0	IIP	0.00	
IIP	509.7			

What I hope I covered.....

- What are dispatchable units of work on z/OS
- Understanding Enclave SRBs
- How WLM manages dispatchable units of work
- The Role of HiperDispatch
- What makes work eligible for zIIP and zAAP specialty engines
- Dispatching work to zIIP and zAAP engines

z/OS Performance Courses from IBM Training

- **Basic z/OS Tuning Using the Workload Manager (WLM)**
 - ES545
 - 4.5 Days, Hands-on Lab Exercises
- **Advanced z/OS Performance: WLM, Sysplex, Unix Services, Web**
 - ES851
 - 4.5 Days
- **ibm.com/training**