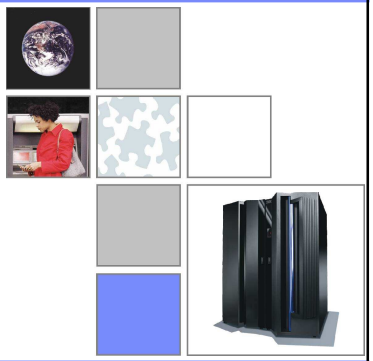
Systems and Technology Group

HiperDispatch on z10

SHARE Winter 2009
Austin, Texas
Session 2831

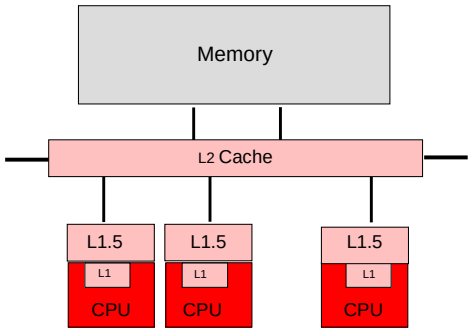
Bob Rogers
IBM Corporation
2455 South Road
Poughkeepsie, NY 12601-5400
rrrogers@us.ibm.com



Processor Design Basics

- **Processor Design**
  - CPU (core)
    - Cycle Time
    - Pipeline
    - Branch Prediction
    - Hardware vs Millicode
  - Memory subsystem
    - High speed buffers (caches)
      - On chip / on Module
      - Private / Shared
    - Buses
      - Number, bandwidth
    - Latency
      - Distance
      - Speed of Light

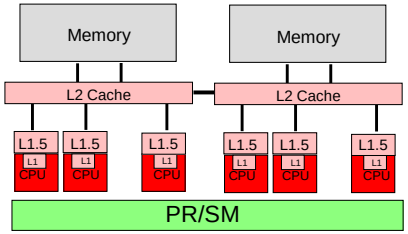
- **Logical View of Single Book**



Hypervisor Overview

- **Hypervisor (PR/SM)**
  - Virtualization layer at Operating System image level
  - Distributes physical resources
    - Memory
    - Channels
      - EMIF
    - Processors
      - Logical processors dispatched on physical processors
      - Dedicated / Shared
      - Affinities
      - Share distribution based on weights

Logical View of 2 Book System

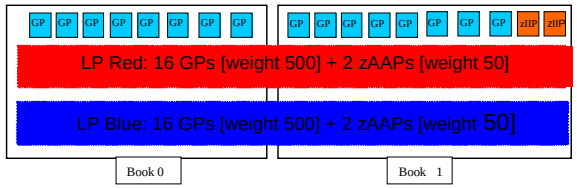


The motivation for HiperDispatch

- Hardware cache can be optimized when a given unit of work is consistently dispatched on the same physical CPU (or related set of CPUs)
  - In the past, System z hardware, firmware, and software have remained relatively independent of each other
  - But, the realities of modern processor and memory designs now make a change appropriate.
    - Different CPUs in the complex have different distances to the various sections of memory and cache (here, "distance" is measured in CPU cycles.)
    - Memory access times can vary from less than 10 cycles to several hundred cycles depending upon cache level and whether the access is local or remote.

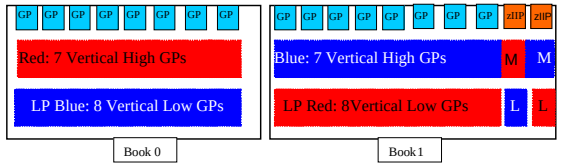
Horizontal CPU management

- PR/SM guarantees an amount of CPU service to a partition based on weights
- PR/SM distributes a partition's share evenly across the logical processors
- Additional logicals are required to receive extra service which is left by other partitions. The extra service is also distributed evenly across the logicals.
- The OS must run on all logicals to gather all its share [z/OS Alternate Wait Management]



Vertical CPU Management

- Logical processors are classified as *vertical high*, *medium* or *low*
- PR/SM quasi-dedicates *vertical high* logicals to physical processors
- The remainder of the share to distributed to the *vertical medium* processors
- *Vertical low* processors are only given service when other partitions do not use their entire share.
- *Vertical low* processors are parked by z/OS when no extra service is available



LP Red: 16 GPs [weight 500] + 2 zAAPs [weight 50]

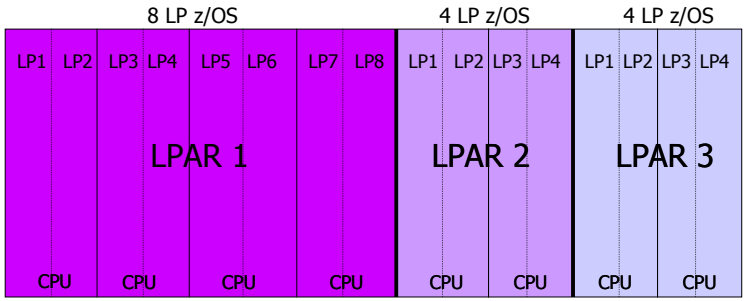
LP Blue: 16 GPs [weight 500] + 2 zAAPs [weight 50]

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 unparked *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

Horizontal CPU Management

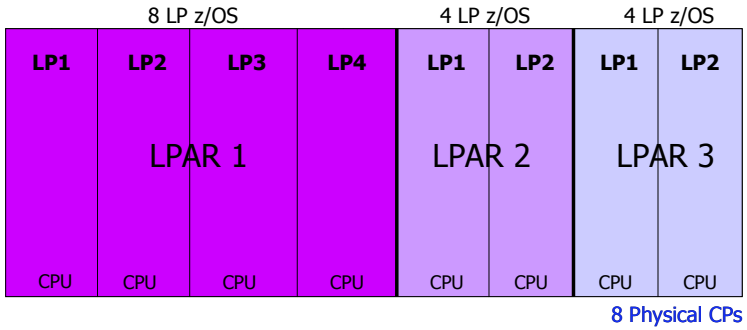
Hiperdispatch **NO**



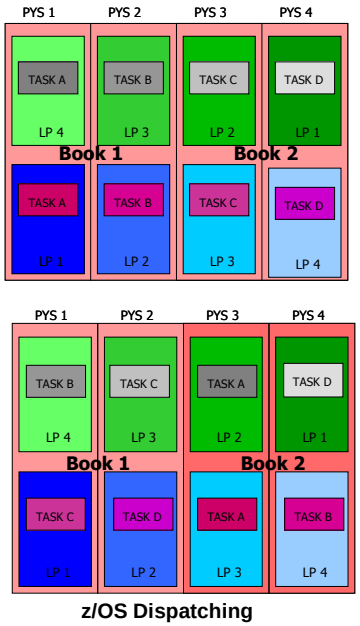
- Typical PR/SM 2-to-1 Logical Processor to physical CP overcommitment
- High competition for physical processors
- z/OS must use all LPs to consume full LPAR weight

Vertical CPU Management

Hiperdispatch YES

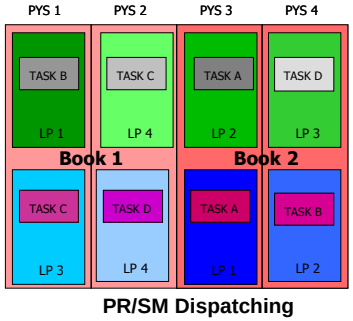


- PR/SM changes to allow z/OS to consume full weight with fewer LPs
- Vertical CPU management - LP to CP mapping is relatively static



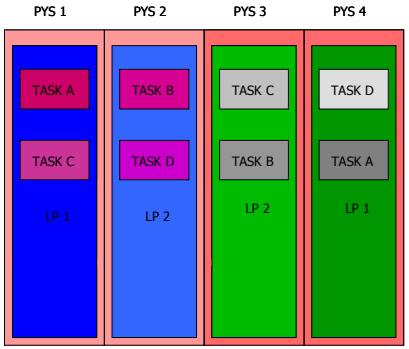
HIPERDISPATCH=NO

- Partition is managed horizontal
- Two levels of Dispatching
  - z/OS dispatches any task on any processor
  - PR/SM dispatches Logical Processors with some level of book affinity



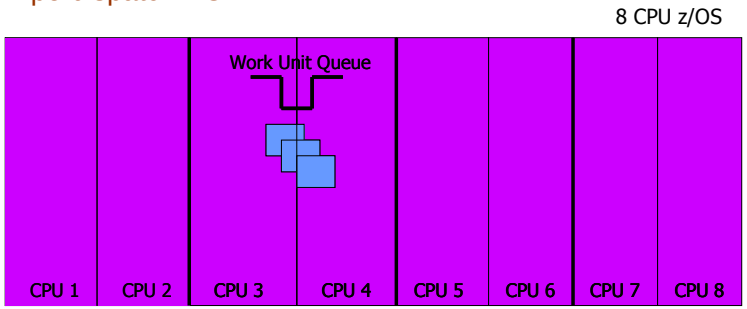
HIPERDISPATCH=YES

- Partition is managed vertical
- PR/SM dispatching is reduced
- z/OS dispatches work with affinity to a small sets of processors



z/OS Dispatcher Affinity

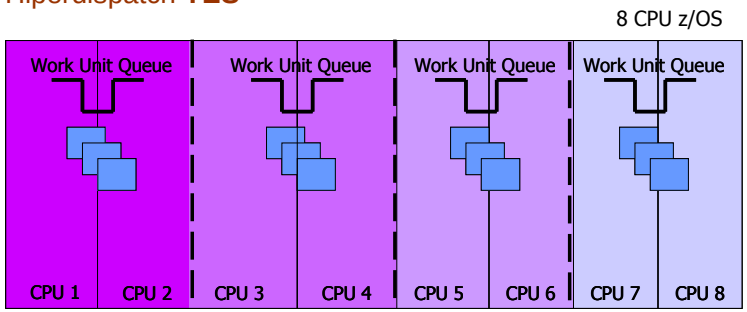
Hiperdispatch **NO**



- All work units have access to all processors of the proper type.
- Cache optimization is minimal.

z/OS Dispatcher Affinity

Hiperdispatch **YES**



- Work units normally have access to only a subset of processor.
- Work imbalances are address by a "needs help" algorithm.
- Cache optimization is improved.

Note that this depiction is simplified for illustrative purposes.
Typically an actual processor subset will have at least 4 processors.

Addressing Workload Variability

- **SRM Balancer** stripes workload across Affinity Nodes by priority in an attempt to keep the work evenly distributed
  - Historic Address space utilization statistics are collected every 2 sec. in an effort to "predict" future requirements
- **Supervisor** implements Needs-Help detection / action to address transient spikes in utilization
  - Maintains priority-based Affinity Node utilization statistics
  - Responsively acts on statistics by asking other LPs for "Help"
- **SRM** tracks PR/SM white space attributes to dynamically address longer term workload requirements
  - Adds / removes LPs to / from existing Affinity Nodes when both the zOS workload warrants it, and the partner LPARs allow it
  - Parks / unparks low priority LPs based on available excess capacity

RMF CPU Activity Report

- In HiperDispatch mode, it is common to see very different utilizations across the logical processors
- Two new columns are added to the report:
  - **Parked %**
    - Percent of the time the LP was in a parked state
  - **Logical Processor Share %**
    - **Percent of a physical processor share that PR/SM will give to this LP if it can use it**
      - ***High Priority* LPs have 100%**
      - ***Medium Priority* LPs have between 1% and 100%**
      - ***Low Priority* (discretionary) LPs have 0%**
- MVS Busy Time definition:

$$\text{MVS BUSY TIME \%} = \frac{\text{Online Time - (Wait Time + Parked Time)}}{\text{Online Time - Parked Time}} * 100$$

See RMF APAR OA24074 for additional details

RMF Monitor I CPU Report

C P U A C T I V I T Y

CPU 2097 MODEL 713 H/W MODEL E26 SERQUENCE CODE 000000000000D6AAD HIPERDISPATCH=YES

---CPU---		---------- TIME % ----------			LOG PROC		----I/O INTERRUPTS---	
TYPE	NUM	ONLINE	LPAR BUSY	MVS BUSY	PARKED	SHARE %	TOTAL RATE	% VIA TPI
CP	0	100.00	69.41	69.41	0.00	100.0	58.67	0.00
	1	100.00	70.75	70.75	0.00	100.0	233.6	0.00
	2	100.00	68.40	68.40	0.00	100.0	254.2	0.00
	3	100.00	63.64	63.64	0.00	65.2	249	0.00
	4	100.00	67.74	67.74	20.00	0.0	1.00	0.01
TOTAL/AVERAGE			67.99	67.99	0.00	365.2		
AA					0.00	100.0		
					0.00	75.0		
TO					0.00	175.0		

The percentage of time that the processor was parked. In HiperDispatch mode, processors with a low amount of physical processor share may be parked. That is, they are not dispatched by z/OS and do not attempt to run work. Without HiperDispatch mode, processors are not parked.

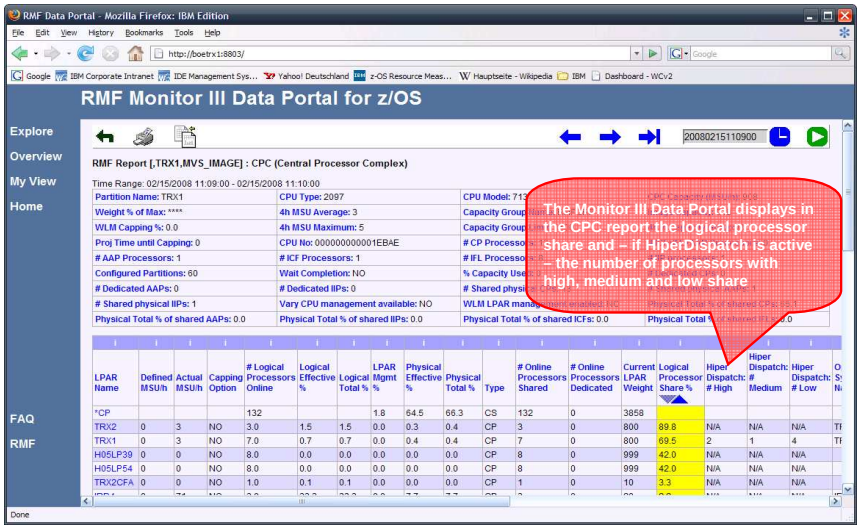
Percentage of the physical processor that the logical processor is entitled to use. Without HiperDispatch mode, the processing weight for the logical partition is equally divided between the online logical processors.
In HiperDispatch mode, logical processors have a high, medium or low amount of physical processor share. High means almost 100% and low means almost or exactly 0% share.

RMF CPU Report Example

```
                                                C P U  A C T I V I T Y
                  z/OS V1R8                 SYSTEM ID UNKN              DATE 11/26/2007
                                            RPT VERSION V1R8 RMF        TIME 22.33.43
CPU  2097   MODEL  732   H/W MODEL  E40  SEQUENCE CODE 00000000000DC6CE HIPERDISPATCH=YES
---CPU---    ---------------- TIME % ----------------     LOG PROC     --I/O INTERRUPTS--
  NUM  TYPE     ONLINE     LPAR BUSY     MVS BUSY    PARKED      SHARE %      RATE     % VIA TPI
    0    CP     100.00     96.33         97.34         0.00      100.0         5.80     48.75
    1    CP     100.00     95.96         97.07         0.00      100.0         4.59     55.30
    2    CP     100.00     95.79         96.84         0.00      100.0         5.10     55.18
    3    CP     100.00     95.46         96.68         0.00      100.0         2.40     53.75
    4    CP     100.00     95.08         96.41         0.00      100.0         8435     10.05
    5    CP     100.00     73.92         96.86         0.00       70.0        20.74      4.95
    6    CP     100.00     74.33         97.13         0.00       70.0        14.15     19.39
    7    CP     100.00     13.84         98.89        85.78        0.0         0.00      0.00
TOTAL/AVERAGE              80.09         96.94                    640.0         8488     10.14
```

Logical paritition share is 640% (6.4 CPUs) with 8 logical processors
5 LPs are *high priority* with 100% share
2 LPs are *medium priority* with 70% share
1 LP is *low priority* (discretionary) with 0% - it was parked 85.78% and busy 13.84%

RMF Monitor III CPC Report



HiperDispatch vs IRD CPU management

- Intelligent Resource Director (IRD) CPU management
  - Automatically adjusts the number of **online** CPUs to achieve the minimum required to run the work of the partition.
  - Actually varies the CPUs online/offline
    - offline CPUs get no share distribution
  - Reacts on a relatively long time scale (minutes)
- HiperDispatch mode
  - automatically adjusts the number of **active** CPUs to achieve the minimum required to run the work of the partition.
  - **"Parks"** unused processors. Parking is simply placing a CPU in a long-term wait until it is again needed to run work.
    - Parked CPUs get no share distribution
  - WLM can park/unpark CPUs over a relatively short time scale (seconds)
- IRD CPU management is disabled in HiperDispatch mode

Special processing for SYSSTC

- Work classified to SYSSTC typically contains lots of short-running local SRBs required for transaction flow.
  - Examples of address spaces recommended to be classified into SYSSTC are VTAM, TCP/IP and IRLM.
- SRBs classified into SYSSTC can execute on any available logical processor even HiperDispatch mode.
- WLM service policies should be reviewed with this in mind.

Controlling HiperDispatch

- HiperDispatch mode is enabled by specifying HIPERDISPATCH=YES in IEAOPTxx
  - The default is HIPERDISPATCH=NO for compatibility
  - But, HIPERDISPATCH=YES is recommended
  - There is a HealthChecker routine to remind if HIPERDISPATCH=NO
- Control authority for global performance data must be enable for proper operation with HiperDispatch mode
  - This option is selected in the logical partition security controls on the Hardware Management Console
  - This is the default selection.