Planning Considerations for HiperDispatch Mode Version 2

Steve Grabarits

Gary King

Bernie Pierce

Version Date: May 11, 2011

This document can be found on the web, www.ibm.com/support/techdocs
Under the category of “White Papers.”

Planning Considerations for HiperDispatch Mode

In addition to the performance improvements available with the IBM System z10 and z196 processors, z/OS workload management and dispatching have been enhanced to take advantage of the System z10 and z196 hardware design. A new mode of dispatching called HiperDispatch provides additional processing efficiencies. With HiperDispatch, the intent is to align work to a smaller subset of processors in order to maximize the benefits of the processor cache structures, and thereby, reduce the amount of CPU time required to execute work. Access to processors has changed with this mode, and as a result, prioritization of workloads via WLM policy definitions becomes more important.

Introduction to HiperDispatch Mode

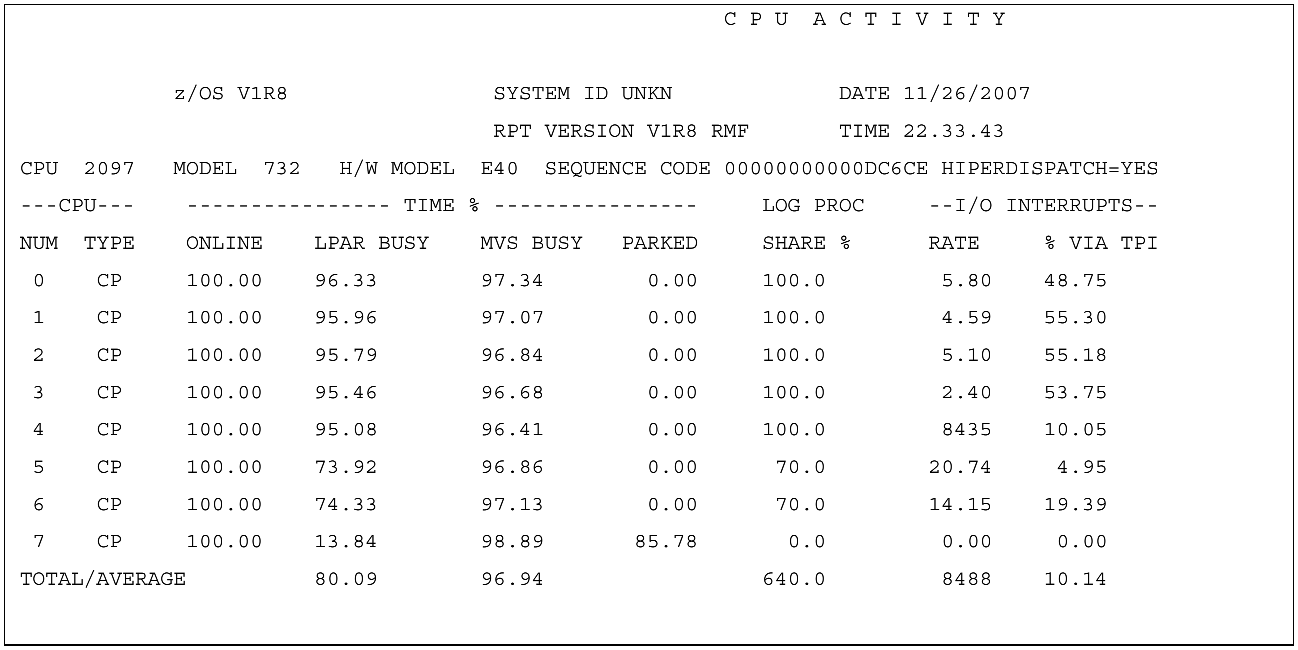
For all levels of z/OS, a TCB or SRB may be dispatched on any logical processor of the type required (standard, zAAP or zIIP). A unit of work starts on one logical processor and subsequently may be dispatched on any other logical processor. The logical processors for one partition receive an equal share for equal access to the physical processors under PR/SM control. For example, if the weight of a logical partition with four logical processors results in a share of two physical processors, or 200%, the LPAR hypervisor will manage each of the four logical processors with a 50% share of a physical processor. All logical processors are used if there is work available, and they typically have similar processing utilizations.

With HiperDispatch mode, the intention is to manage work across fewer logical processors. A new concept of maintaining a working set of processors required to handle the workload is introduced. In the previous example of a logical partition with a 200% processor share and four logical processors, two logical processors are sufficient to obtain the two physical processors worth of capacity specified by the weight; the other two logical processors allow the partition to access capacity available from other partitions with insufficient workload to consume their share. z/OS will limit the number of active logical processors to the number needed based on partition weight settings, workload demand and available capacity. When using HiperDispatch, it is very important for the weight of each logical partition and the resulting share for each type of processor be appropriate for the workload. In HiperDispatch mode, z/OS will not employ the other two logical processors in the example above, unless there is unused capacity by other partitions. z/OS will also take into account the processor topology when dispatching work, and it will work with enhanced PR/SM microcode to build a strong affinity between logical processors and physical processors in the processor configuration. The logical processors for a partition in HiperDispatch mode will fall into one of the following categories:

- Some of the logical processors for a partition may receive a 100% processor share, meaning this logical processor will receive a target of 100% share of a physical processor. These would be viewed as having a *high* processor share. These logical processors are sometimes referred to as vertical high (VH) logical processors. Typically, if a partition is large enough, most of the logical partition's share will be allocated among logical processors with a 100% share. PR/SM will establish a strong affinity between the logical processor and a physical processor, and these processors will provide optimal efficiencies in HiperDispatch mode.

- Other logical processors may have a *medium* amount of physical processor share. The logical processors would have a processor share greater than 0% and up to 100%. These medium logical processors have the remainder of the partition’s shares after the allocation of the logical processors with the high share. These logical processors are sometimes referred to as vertical medium (VM) logical processors. PR/SM reserves at least a 50% physical processor share for the medium processor assignments, assuming the logical partition is entitled to at least that amount of service. For example, a partition with share of 2.1 physical CPs will be assigned one high logical processor and two medium logical processors. The two medium processors will “share” the 1.1 physical processors worth of share which remains after the high processor is designated. If two logical processors are designated as 100%, the share of the third processor would be 10% which is considered inadequate. The share of the two VM processors is improved by any unused share of all VH processors assigned the partition.
- Some logical processors are not needed to allow the partition to consume the physical processor resource associated with its weight. These logical processors are sometimes referred to as “discretionary” or vertical low (VL) logical processors. These logical processors are initially *parked*. In a parked state, logical processors do not dispatch work; they are in a long term wait state. These logical processors are parked when they are not needed to handle the partition’s workload (not enough load) or are not useful because physical capacity does not exist for PR/SM to dispatch (no available time from other logical partitions).

When examining an RMF CPU Activity report in HiperDispatch mode, it will become common to see very different processing utilizations across different logical processors of a logical partition. The following figure shows an example of this behavior, and it also shows two new columns in the report: parked time % and logical processor share %.



```
                                                      CPU ACTIVITY


            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
```

In this example, the logical processor share for the partition of 640% was allocated across five logical processors with a high share of 100%, two logical processors with a medium share of 70%, and one discretionary logical processor with a share of 0% which was parked 85.78% of the time. Logical processor 5 with expected share 100%, is converted to a medium logical processor. The two medium logical processors share the resulting 140% share rather than dividing the share. The discretionary logical processor, CP 7, was not parked 14.22% of the online interval, and it was busy on a physical processor 13.84% of this same interval. The actual MVS busy for the interval can be calculated as the product of 14.22% and 98.89% or 14%. When logical processor 7 is not parked, it joins the medium logical processor pool formed with logical processors 5 and 6. Then the 140% share is used to power the logical processors 5-7. With HIPERDISPATCH=NO, the logical processor share would be 80% for each of the 8 logical processors.

The MVS BUSY fields in the RMF report reflects the effective used capacity for the logical processors and the entire logical partition.   The figures were based on the difference between online time and MVS wait time to provide an operating system perspective of busy time. Parked processors in HiperDispatch mode will generally reflect unavailable capacity at high physical processor utilizations. The formula for MVS Busy has been changed with HiperDispatch mode to exclude Parked Time to show how busy the logical processor was when not parked. The formula is now:

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

Refer to RMF APAR OA24074 for additional details with this change.

HiperDispatch mode changes the effect of the CPU management support of the Intelligent Resource Director (IRD). The WLM LPAR weight management function is unchanged. The Vary CPU Management function is replaced by the parked / not parked aspect of discretionary processors.   As with the IRD function, the initial specification for number of logical processors in HiperDispatch mode is to define as many as are reasonably likely to be needed and productive.

The control authority for global performance data must be enabled (the default) for proper operation with HiperDispatch mode in a logical partition.   This option is selected in the logical partition security controls on the Hardware Management Console.

Processing Benefits

HiperDispatch can lead to improved efficiencies in both the hardware and software in the following two manners:

1) Work may be dispatched across fewer logical processors therefore reducing the “multi-processor (MP) effects” and lowering the interference among multiple partitions.

2) Specific z/OS tasks may be dispatched to a small subset of logical processors which PR/SM will tie to the same physical processors thus improving the hardware cache re-use and locality of reference characteristics such as reducing the rate of cross-book communication.

Therefore, the magnitude of the potential improvement from HiperDispatch is related to:

a. The processor cache topology, sizes and access times
b. Number of physical processors
c. Size of the z/OS partitions in the configuration
d. Logical : physical processor ratio
e. Memory reference pattern or storage hierarchy characteristics of the workload.
f. Exploitation of IRD Vary CPU management

z10

Generally, a configuration where the largest z/OS image fits within a book will see minimal improvement. Workloads which are fairly CPU-intensive (like batch applications) will see only small improvements even for configurations with larger z/OS images since they typically have long-running tasks which tend to stick on a logical engine anyway. Workloads with common tasks and high dispatch rates, as often seen in transactional applications, may see larger improvements depending on the size of the z/OS images involved. Over committed partition configurations, i.e. have higher logical to physical ratios, may see some improvement although the benefit of dispatching to a reduced number of logical processors overlaps with benefits already available with IRD and various automation techniques used to reduce the number of online logical processors to match capacity needs.

The range in benefit is expected to be from 0% to 10% following the sensitivities described above; specifically, configurations with z/OS images small enough to fit in a book or running batch-like workloads will tend to fall at the low-end of the range, multi-book configurations with z/OS images in the 16way to 32way range and running transactional workloads will tend to fall toward the middle of the range, and very large multi-book configurations with very large z/OS images and running workloads with intense memory reference patterns will tend to fall toward the high end of the range.

To be slightly more specific but intended as rule of thumb rather than strong expectation:

1-2% for a 1 book environment - less than 12 purchased CPs/zIIPs/zAAPs

2-4% for a 2 book environment - less than 26 purchased CPs/zIIPs/zAAPs

4-7% for a 3 book environment - less than 40 purchased CPs/zIIPs/zAAPs

7-10% for a 4 book environment - less than 64 purchased CPs/zIIPs/zAAPs

z196

The z196 introduces a new level of cache shared by up to 4 processors on a chip. HiperDispatch attempts to align dispatching affinities with this cache to minimize cross-chip accesses just like HiperDispatch tries to minimize cross-book accesses on the z10 and z196 in multi-book configurations. This means HiperDispatch on a z196 can provide more value than on a z10, particularly to single-book configurations. The magnitude of the improvement is sensitive to the "share" of the physical processors a partition is given. The share of a partition for a type of processor (CP/zAAP/zIIP) is the weight of the partition for the processor type divided by the sum of the weights times the number of physical processors of the type. If the weight of a partition for zIIPs is 75 and the sum of the weights is 100 and there are five zIIPs in the shared pool, this partition is entitled to 3.75 of the five zIIPs or 375% of a physical zIIP. The following table provides estimates for the possible increase in capacity when HiperDispatch is enabled. There are partitions with workloads which may not achieve a value within the ranges but it is expected the percentages are reasonable for most partitions.

Note: Expectations for z196 capacity compared to other processors are provided via Large System Performance Reference (LSPR) and Processor Capacity Reference for System z (zPCR) and are established with HiperDispatch active. The table below should be viewed as how much capacity one is possibly forgoing if HIPERDISPATCH is not enabled.

Share of the partition - assumes 1.5 logical to physical ratio	Number of Physical CPs + zIIPs + zAAPs			
	<=16	17-32	33-64	65-80
0 <= share in processors < 1.5	0%	0%	0%	0%
1.5 <= share in processors < 3	2-5%	3-6%	3-6%	3-6%
3 <= share in processors < 6	4-8%	5-9%	6-10%	6-10%
6 <= share in processors < 12	5-11%	7-13%	8-14%	8-16%
12 <= share in processors < 24	-	8-16%	10-18%	11-21%
24 <= share in processors < 48	-	-	11-21%	12-24%
48 <= share in processors <= 80	-	-	-	14-26%

HiperDispatch Enablement

HiperDispatch mode is enabled by specifying a new parameter, HIPERDISPATCH=YES, in the IEAOPTxx member of SYS1.PARMLIB. This parameter can be changed dynamically with the use of the SET OPT command. The default is HIPERDISPATCH=NO for z/OS releases up to z/OS 1.Release 12. With Release 13 of z/OS, HIPERDISPATCH=YES is the default when hosted on a z196. It is recommended installations use HiperDispatch to take advantage of the processing benefits with System z10. z/OS guests running under z/VM cannot use HiperDispatch mode. Specifying Hiperdispatch=YES in the IEAOPTxx parmlib of a z/OS guest will be ignored. Since logical processors in HiperDispatch mode are referred to as vertical high/medium/low, partitions and logical processors of partitions not in HiperDispatch mode are sometimes referred to as "horizontal".

The expectation for HiperDispatch mode from the above table is low to no enhanced capacity for partitions with less than 1.5 physical processors. These partitions will always have at least two logical processors active at all times in HiperDispatch mode. One may choose to keep these or any partition in horizontal mode. There has been some concern regarding enabling HiperDispatch in some but not all partitions of a z10 or z196. There should be no concern since the PR/SM management of physical processors gives no subtle benefit to vertical high processors. A physical processor is identified to host a vertical high logical processor but the physical processor is not dedicated to the vertical high processor. Since the vertical high processor has 100% share, it is dedicated when there is 100% demand but not when the physical processor is released by the logical processor. The physical processor is immediately available to dispatch any other logical processor from any other partition. As all processors reach a point of demanding share, PR/SM ensures they all receive the share indicated by the weight of the partition.

WLM Policy Considerations

HiperDispatch improves the efficiency of the hardware by creating a closer tie between a work unit and the physical processors on which it will run. However, in doing so, this may reduce the number of processors to which work has access. Reduced access to logical processors may change the processor queuing and response time effects associated with the workload (i.e., in queuing theory, as the number of servers is reduced, the potential for queuing delays is increased). Therefore, insuring proper WLM goals and importance becomes more important with HiperDispatch.

WLM service policy definitions should be reviewed to ensure business goals are still being met. In HiperDispatch mode, work executing in SYSSTC has some additional, unique processing capabilities compared to work executing in user-defined service classes. SRB activity from the SYSSTC class still retains the capability to execute on any available logical processor. This supports the highly interactive requirements for work typically classified to this service class, or more specifically, lots of short-running, local SRBs required for transaction workflow. Examples of address spaces with this type of work which are highly recommended for classification to SYSSTC are VTAM, TCP/IP and IRLM. The WLM service policies should be reviewed to consider the classification of these spaces.

With work assigned to a subset of processors, it becomes more critical in HiperDispatch mode to properly differentiate and prioritize work in the WLM policies. An example of the type of tuning which might be required was seen during IBM performance measurements with the OLTP-W LSPR workload. This environment has a WebSphere front-end connected to CICS/DB2 applications and data, and a workload with a low mean time to wait needing frequent access to CPU needed and prioritization above heavy CPU-bound applications. In the beginning of testing the WebSphere transactions and the CICS transactions were classified with equal WLM importance. With the move to HiperDispatch mode, the workload response times suffered and overall throughput was constrained. It became imperative for the WAS transactions to be given preferential CPU access over the heavy CICS/DB2 work. The WebSphere transactions service class goal was made more important than the CICS service class goals. The flow of work in the system was much smoother with this change. These changes also provided response time benefits with HIPERDISPATCH=NO.

This tuning change is not a general recommendation for setting the relative importance of CICS and WAS workloads. It is presented as an example of the type of tuning which may be needed after implementation of HiperDispatch. If there had been some substantial lower importance or discretionary work in the partition, tuning would probably have been unnecessary. Since the methodology for LSPR is a single workload at approximately 90% CPU utilization, all the work is “important” so it is more essential to be precise than when running heterogeneous workloads.

As with any change of a processor configuration, velocity goals should be reviewed to ensure these goals are appropriate for the new environment.

HiperDispatch mode can provide processing efficiencies with better usage of the System z10 and z196 processor structures. However, with these changes of the dispatching of work among the processors, additional attention and review of the WLM policy may be needed to ensure proper workflow of the system. It becomes more important for critical work to be identified and given an appropriate importance and for important, low mean time to wait work to be classified properly versus other less important or more CPU-bound work.

Considerations for zIIPs and zAAPs in HiperDispatch mode

The determination if currently executing zIIP/zAAP logical processors need help is based on ZIIPAWMT and ZAAPAWMT. The value range for these parameters in HiperDispatch mode is 1600- 499,999 with the default set at 3200. Like CCCAWMT, the value (microseconds) controls the interval for determining if “help” is required for an affinity node. An affinity node is the group of logical processors supporting a subset of the work in the partition. The target size for the affinity node is four logical processors. The actual “need help” interval is 1/8th of the AWMT value so the default is 400 microseconds. The concept of “needs help” is very old. An affinity node needs help when it is unable to run all the work assigned with the current set of logical processors available to dispatch work at the affinity node. Help will be obtained by signaling a logical processor assigned to the node, if any are in wait state. If the state of “need help” exceeds the logical processors assigned to the affinity node, a logical processor assigned a different affinity node of the same type (zIIP/zAAP) may be asked to help with the work at this affinity node.

A zAAP or zIIP affinity node is allowed to ask a CP for help if IFAHONORPRIORITY or IIPHONORPRIORITY is set to YES in the IEAOPTxx member. When all zAAP/zIIPs have been exhausted for help, CPs may be asked to help when honor priority YES is chosen. Because it can be less desirable to have zIIP/zAAP work processed on CPs, the upper bound for ZIIPAWMT and ZAAPAWMT may be set as high as 499,000. The default is 3200. If the zIIP/zAAP work running on CP is unacceptable, try doubling ZIIPAWMT and/or ZAAPAWMT and adjust higher if necessary. Since help is an essential element of HiperDispatch, values larger than approximately 10,000 should be considered with caution. This value controls the interval at which help is obtained from logical processors assigned to the affinity node as well as logical processors outside the node.

Summary

HiperDispatch mode is an optional mode of management of logical and physical processors for the z10 and z196 servers. The objective of improving processor capacity by improving the value of the levels of cache is achieved by complementary management of processing resources between the PR/SM virtualization function of the machine and the z/OS management of work in a partition. With the z10 processors, the primary advantage of HiperDispatch is to manage work in a partition which crosses books or local and remote L2 cache. Since most partitions on z10 fit in a single book, HiperDispatch had significant value for a modest percent of partitions. With z196, the chip level cache (L3 cache) provides an opportunity for increased value for HiperDispatch mode. IBM strongly suggests HiperDispatch mode for all z/OS partitions with z196 servers.

IBM Corporation
New Orchard Rd.
Armonk, NY 10504
U.S.A

Produced in the United States of America
05/11

Information concerning non-IBM products was obtained from the suppliers of their products or their published announcements. Questions on the capabilities of the non-IBM products should be addressed with the suppliers.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM’s future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.