Prologue - The IBM Large System Performance Reference (LSPR) ratios represent IBM's assessment of relative processor capacity in an unconstrained environment for the specific benchmark workloads and system control programs specified in processor capacity tables. Ratios are based on measurements and analysis. The amount of analysis as compared to measurement varies with each processor.

There are many published sources of processor capacity data available in the industry today. Most of these sources provide data in the form of MIPS tables. MIPS tables available from consultants and industry watchers are not based on independent measurements. Rather, they typically are developed using manufacturer's announced performance claims. Over time, some of these MIPS tables may include a subjective analysis of feedback from various clients of these systems.

The LSPR's focus is solely on processor capacity, without regard to external resources such as storage size, or number of channels, control units, or I/O devices. To assure that the processor is the primary focus, the processor capacity data reported assume sufficient external resources so as to prevent any significant external resource constraints.

Many factors, including but not limited to the following, may result in the variances between the ratios provided herein and actual operating environments:

- Differences between the specified workload characteristics and your operating environment
- Differences between the specified system control program and your actual system control program
- I/O constraints in your environment
- Incorrect assumptions in the analysis
- Unknown hardware defects in processors used for measurement.

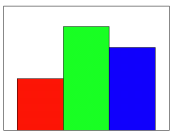
NOTE: the LSPR is designed to represent each processor in its best light; that is, the processor itself is the only limiting factor to doing work.

While MIPS tables may be useful for rough processor positioning, they should not be used for capacity planning purposes. Single-number processor capacity tables are inherently prone to error because they are not sensitive to the type of work being processed or to the LPAR configuration of the processor.

What are the major changes to the z/OS V1R13 LSPR?

The LSPR ratios reflect the range of performance between System z servers as measured using a wide variety of application benchmarks. The latest release of LSPR continues with the methodology introduced with the z/OS V1R11 LSPR. Prior to that version, workloads had been categorized by their application type or software characteristics (for example, CICS, OLTP-T, LoIO-mix). With the introduction of CPU MF (SMF 113) data starting with the z10 processor, insight into the underlying hardware characteristics that influence performance was made possible. The LSPR defines three workload categories, LOW, AVERAGE, HIGH, based on the metric called “Relative Nest Intensity (RNI)” which reflects a workload’s use of a processor’s memory hierarchy.

Continued



What model is used as the “base” or “reference” processor in the z/OS V1R13 LSPR table?

The 2094-701 processor model is used as the base in the z/OS V1R13 table. Thus, the ITRR[1] for the 2094-701 appears as 1.00.

Note: In the zPCR the reference processor may be set at the user’s discretion.

What is the multi-image table in the LSPR?

Typically, IBM System z processors are configured with multiple images of z/OS. Thus, the LSPR continues to include a table of performance ratios based on average multi-image z/OS configurations for each processor model as determined from the profiling data. The multi-image table is used as the basis for setting MIPS and MSUs[2] for IBM System z processors.

What multi-image configurations are used to produce the LSPR multi-image table?

A wide variety of multi-image configurations exist. The main variables in a configuration typically are: 1) number of images, 2) size of each image (number of logical engines), 3) relative weight of each image, 4) overall ratio of logical engines to physical engines, 5) the number of books and 6) the number of ICFs/IFLs. The configurations used for the LSPR multi-image table are based on the average values for these variables as observed across a processor family. It was found that the average number of images ranged from 5 at low-end models to 9 at the high end. Most systems were configured with 2 major images (those defined with >20% relative weight). On low- to midrange models, at least one of the major images tended to be configured with a number of logical engines close to the number of physical engines. On high-end boxes, the major images were generally configured with a number of logical engines well below the count of physical engines reflecting the more common use of these processors for consolidation. The overall ratio of logical to physical engines (often referred to as “the level of over-commitment” in a virtualized environment) averaged as high as 5:1 on the smallest models, hovered around 2:1 across the majority of models, and dropped to 1.3:1 on the largest models. The majority of models were configured with one book more than necessary to hold the enabled processing engines, and an average of 3 ICFs/IFLs were installed.

Can I use the LSPR multi-image table for capacity sizing?

For high-level sizing, the multi-image table may be used. However, the most accurate sizings require using the zPCR tool’s **LPAR Configuration Capacity Planning** function, which can be customized to exactly match a specific multi-image configuration rather than the average configurations reflected in the multi-image LSPR table.

What “capacity scaling factors” are commonly used?

The LSPR provides capacity ratios among various processor families. It has become common practice to assign a capacity scaling value to processors as a high-level approximation of their capacities. The commonly used scaling factors can change based on the version of LSPR. For z/OS V1R13 studies, the capacity scaling factor commonly associated with the reference processor set to a 2094-701 is 593 which is unchanged from that used with z/OS V1R11. This value reflects a 2094-701 configured with a single image of z/OS - no complex LPAR configuration (i.e., multiple z/OS images) effects are included. For the z/OS V1R13 multi-image table the commonly used scaling factor is .944x593=559.792. Note the .944 factor reflects the fact that the multi-image table has

processors configured based on the average client LPAR configuration; on a 2094-701, the cost to run this complex configuration is approximately 5.6%. The commonly used capacity scaling values associated with each model of a processor may be approximated by multiplying the AVERAGE column of ITRRs in the LSPR z/OS V1R13 multi-image table by 559.792. The PCI (Processor Capacity Index) column in the z/OS V1R13 multi-image table shows the result of this calculation. Note that the PCI column was actually calculated using zPCR, thus the full precision of each ITRR is reflected in the values. Minor differences in the resulting PCI calculation may be observed when using the rounded values from the LSPR table.

Of course, using a table of values based on a capacity scaling factor only allows for a gross approximation of the relative capacities among the processor models A more accurate analysis may be conducted by using zPCR to perform a detailed LPAR configuration assessment to develop the capacity ratio between a “before” and “after” configuration.

How much variability in performance should I expect when moving a workload to a zEnterprise EC12 (zEC12) processor?

As with the introduction of any new server, workloads with differing characteristics will see some variation in performance when moved to a zEC12. The performance ratings for a server are determined by the performance of the average workload that represents what we understand to be the major components of our customers' production environments. While the ratings provide good "middle-of-the-road" values, they do represent an average, and by definition some workloads fall higher than the average and some workloads fall below. The zEC12 has fairly balanced improvements both in its micro-processor design (higher clock speed and improved out-of-order execution) and in its memory hierarchy (much larger on-chip shared cache and book-level shared cache). This suggests that the range of variation in performance of workloads with different hardware characteristics will be somewhat tighter than that seen in recent processor generations.

Once my workload is up and running on a zEnterprise EC12 (zEC12), how much variability in performance will I see?

Minute-to-minute, hour-to-hour and day-to-day performance variability generally grows with the size (capacity) of the server and the complexity of the LPAR configuration. With its improved micro-processor and memory hierarchy design, and the capability to be configured with up to 101 engines, the zEC12 has the capability to deliver up to 1.5 times the capacity of the largest previous server. Continued enhancements to HiperDispatch have been made to help reduce the potential for increased performance variability. In the spirit of autonomic computing, PR/SM and the z/OS dispatcher cooperate to automatically place and dispatch logical partitions to help optimize the performance of the hardware, and minimize the interference of one partition to another. However, while the average performance of workloads is expected to remain reasonably consistent when viewed at small increments of time or by individual jobs or transactions, performance can potentially see some variation simply due to the expected larger and more complex LPAR configurations that can be supported by the zEC12.

What is HiperDispatch and how does it impact performance?

HiperDispatch is the z/OS exploitation of PR/SM’s Vertical CPU Management (VCM) capabilities and is exclusive to System z processors since the z10. Rather than dispatch tasks randomly across all logical processors in a partition, z/OS will tie tasks to small queues of logical processors, and dispatch work to a “high priority” subset of the logicals. PR/SM provides processor topology information and updates to z/OS, and ties the high priority logical processors to physical processors. 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. Note the value of HiperDispatch is higher on the z196 and zEC12 processors due to its sensitivity to the chip-level shared cache topology.

A white paper is available concerning HiperDispatch at:

http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101229

What is the performance improvement a z/VM customer might experience on the zEnterprise EC12 (zEC12)?

The performance ratios that a z/VM customer workload might experience when migrating to zEC12 from older processors will vary. For z/VM, a single workload was run for the LSPR which has characteristics similar to the HIGH relative nest intensity workload. However, customer workloads have been shown to cover the full range from LOW to HIGH RNI workloads. Thus, it is suggested that you consider the full range of LSPR workloads.

1. ITRR (Internal Throughput Rate Ratio) - ITRs are useful for determining capacity relationships between processors for a given SCP and workload environment. This is done by dividing the ITR of one processor by the ITR of another to produce an ITR ratio (ITRR). For example, to determine the capacity of processor "B" relative to that of processor "A", use the formula: ITR Ratio (or ITRR) = ITR for CPU-B / ITR for CPU-A .

2. MSUs (Million Service Units) - is a measurement of the amount of processing work a computer can perform in one hour.

Service Unit: The amount of service consumed by an address space and is computed by the formula:
service = (CPU x CPU Service Units) + (SRB x SRB Service Units) + (IOC x I/O Service Units)+ (MSO x Storage Service Units)

where

CPU Service Units = task (TCB) execution time, multiplied by an SRM constant which is CPU model dependent.

SRB Service Units = service request block (SRB) execution time for both local and global SRBs, multiplied by an SRM constant which is CPU model dependent.

I/O Service Units = measurement of individual dataset I/O activity and JES spool reads and writes for all datasets associated with the address space.

Storage Service Units = (central page frames) x (CPU service units) x 1/50, where 1/50 is a scaling factor designed to bring the storage service component in line with the CPU component.

– – –