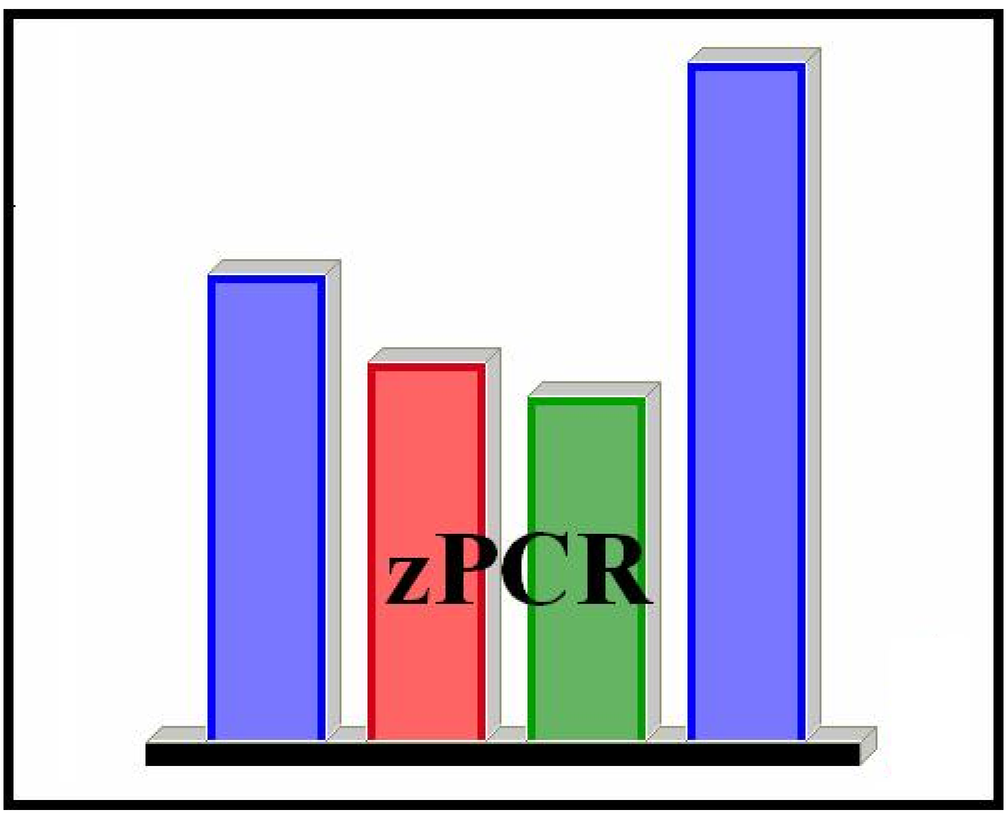
zPCR Version 7.1

LSPR Workloads

and

Capacity Planning Considerations



LSPR Workloads 2010c25.doc

07/16/2010

Table of Contents

Summary of What Has Changed

LSPR data as of 07/22/2010

zPCR Version 7.1, dated 07/22/2010

1. Single Set of LSPR Data

A single set of LSPR data is now used to project capacity across the entire IBM System z product line. The current LSPR data is based primarily on z/OS-1.11, with data points included for z/VM (Linux guests) and native Linux.

Previous versions of **zPCR** relied on separate LSPR tables for each z/OS release, including z/OS-1.4, z/OS-1.8, z/OS-1.8, and z/OS-1.9. Now all capacity results will be based solely on the new z/OS-1.11 LSPR table. When defining a partition configuration, the various releases of z/OS can still be referenced. Capacity results will not differ. However, the version information will be used to enforce various SCP limitations, such as the maximum number of CPs supported, and whether zAAP or zIIP CPs are supported.

2. New Workload Categories

Previous versions of **zPCR** provided access to the LSPR workload primitives that were published. In addition, suggested workload mixes were defined to represent the capacity bandwidth for production workloads on System z processor models. Most capacity planning studies used the suggested workload mixes.

LSPR data now includes three generic workload categories, named ***Low***, ***Average***, and ***High***. The data representing these workload categories is based on actual measurements of various workload primitives, which are no longer published. The workload categories are actually mixes of the primitives. The ***Low*** and ***High*** categories are intended to represent the bandwidth or bounds of capacity expectation for any given processor model, when compared to another model, while the ***Average*** category lies in the middle. The full meaning of these names is discussed later in this document.

When defining partitions in **zPCR**, the new workload category names must be used in lieu of the old workload names (primitives and suggested mixes). To improve granularity with the new LSPR data, **zPCR** has defined two additional workload categories, ***Low-Avg*** and ***Avg-High***. These fall midway between the published names.

A general alignment exists between the new workload categories and the old suggested workload mixes. However, the relationships are not direct. Therefore, capacity results with the new LSPR data will likely vary from those with the older LSPR data. The degree to which results vary should be minimal. And the latest LSPR data should always be considered as the most accurate characterization of capacity for IBM System z processors.

When opening a former **zPCR** study, conversion of the old workload names (primitives and suggested mixes) to the corresponding new workload categories will automatically be performed. SCPs will also be adjusted for the revised **zPCR** SCP naming convention.

When modeling an existing LPAR configuration in **zPCR**, a workload category must be chosen for each partition. The workload assignment will be based on (1) the ***LPAR Host*** processor model, (2) the partition ***DASD I/O Rate***, and (3) the partition ***Utilization***. The workload determination can be made by using **zPCR**'s ***Workload Selection Assistant***. When the three metrics are entered, a pointer will indicate the best workload category match. The workload determination can also be made external to **zPCR** (see Selection based on DASD I/O Rate).

When generating an LPAR configuration from ***RMF*** data, the ***LPAR Host*** processor model and partition ***Utilization*** are known. On the ***Create LPAR Configuration from RMF*** window, the ***DASD I/O Rate*** for each partition can be directly entered, automatically invoking the ***Workload Selection Assistant***, and making an appropriate workload category assignment. If no ***DASD I/O Rate*** is entered for a partition, the ***Average*** category will be assigned.

When generating an LPAR configuration from ***EDF*** data, the ***LPAR Host*** processor model and partition ***Utilization*** are known. For partitions where SMF Type 74 records were captured, the partition's ***DASD I/O Rate*** is also available. On the ***Create LPAR Configuration from EDF*** window, the ***DASD I/O Rate***, when available for a partition, will be filled in and the appropriate workload assignment will be made. When the DASD I/O Rate for a partition is not obtained from the EDF, it can be entered manually, automatically invoking the ***Workload Selection Assistant***, and making an appropriate workload category selection. If no ***DASD I/O Rate*** is entered, the ***Average*** category will be assigned to the partition. In addition, since the ***EDF file*** includes any captured ***CPU Measurement Facility*** (SMF type 113 data), the same window will indicate (as a ***113 Hint***) the workload category that would have been chosen for the partition based on the hardware counter data. If the ***113 Hint*** workload category is to be used, it must be assigned in **zPCR**'s ***Partition Definition*** window or ***Partition Detail Report*** window.

3. New Typical Reference-CPU Setting

Due to the new LSPR workload categories, it is no longer possible to exactly repeat the results of studies made with previous **zPCR** versions. The new workload categories somewhat affect scaling for the most frequently installed processor models. In order to produce capacity results for these processor models similar to those made with previous LSPR data, the typical **MIPS** (or **PCI**) value for the ***Reference-CPU*** has been adjusted, as follows:

Previous typical ***Reference-CPU*** setting: **2094-701** at **602 MIPS**

New typical ***Reference-CPU*** setting: **2094-701** at **593 MIPS**

When loading a previous **zPCR** study, if its ***Reference-CPU*** scaling-factor appears to be a **MIPS** or a **PCI** value, a dialog will appear offering to reset that scaling-factor. The suggested new scaling-factor will be the old scaling-factor multiplied by the ratio of **593** ÷ **602**, or **0.985**.

Concerns about this change can be avoided by using **zPCR**'s ***Calibrate*** function to assign a **MIPS** or **PCI** capacity value for the current configuration. Then alternate configurations being considered will be shown with capacity values appropriate for that account. Calibrate can be done from either the ***Host Summary Report*** window or the ***Partition Detail Report*** window by clicking the **Calibrate Reference-CPU** button.

Summary

Capacity results from **zPCR version 7.1** should be considered just as reliable as those from past versions. In addition, this version provides support for the recently announced **zEnterprise 196** processor models.

Capacity results for any specific LPAR configuration can vary significantly from those observed in the ***LSPR Multi-Image table***. Differences are due to the degree to which the specific LPAR configuration deviates from the typical configuration used for the ***LSPR Multi-Image table***. The use of zAAP, zIIP, IFL, and ICF specialty engines will also contribute to deviation, since these CP types are not included in the typical configurations. The ***LPAR Configuration Capacity Planning*** function in zPCR should be considered as a more accurate way to assess the capacity expectation for any specific LPAR host, since it considers the exact partition configuration, including the individual SCP/workload environments being run in each partition and any specialty CPs that are configured. Users should always focus on the "***Relative Capacity***" of one LPAR configuration to another, rather than on the absolute **MIPS** results for a single configuration.

Overview of LSPR Data

LSPR data as of 07/22/2010

LSPR General Discussion

LSPR data is derived from workloads measured in LPAR-mode. Each workload is run in a simple LPAR configuration (i.e., a single shared partition with all of the physical CPs assigned). z/OS and z/VM measurements are made with 64-bit addressing, and the Linux measurements are made with 31-bit addressing.

LSPR data is based on a variety of workload primitives, each designed to focus on a major type of data processing activity, such as batch, interactive, or on-line/database transactions. These workload primitives do not concentrate on a single activity such as a specific job or application. Instead, each includes a broad mix of activity related to the processing activity being characterized. Focusing on a broad mix helps to assure that resulting capacity data is not skewed.

Some of the workload primitives are designed to stress the processor, while others are designed to stress the storage subsystem. As a result, the overall capacity bandwidth of each processor model can be understood. The capacity bandwidth for production workloads, while considerable, is generally less than that revealed by the workload primitives. Therefore, published LSPR data is based on various mixes of the measured workload primitives. The mixes are selected so as to provide capacity planning relationships within the bandwidth considered reasonable for production workloads.

High-level descriptions of the various workload primitives used for LSPR in the past are provided later in this document (see LSPR Workload Primitives).

LSPR Versions starting 07/22/2010

A single set of LSPR data covers the entire IBM System z product line. The current data is based on z/OS-1.11, and includes three workload categories (sets of values) for each processor model, ***Low***, ***Average***, and ***High***. To provide increased granularity for capacity planning purposes, **zPCR** provides 2 additional workload categories, ***Low-Avg*** and ***Avg-High***, positioned midway between the primary category values. The table also includes a single workload category for z/VM and a single workload category for Linux.

The names assigned for workload categories relate to ***NEST Intensity***. ***NEST*** refers to the processor's storage subsystem, starting after the level 1 cache, all the way out to central storage. ***Intensity*** refers to the rate at which the processor must go beyond the level 1 cache, out to the storage hierarchy, to find an instruction or data. Also considered is, on the average, how far out on the storage hierarchy it must go. Each successive level of storage that must be accessed requires more time than the previous level, with accesses to central storage being the most costly.

The storage subsystem design varies (often significantly) between System z processor families. Each specific design has its own effect on the performance and capacity characteristics of a workload. In order that the published LSPR workloads reside at the same ***NEST Intensity*** position regardless of the processor family, the term ***Relative NEST Intensity*** (***RNI***) is used.

Every workload, be it production or benchmark, will have its own ***Relative NEST Intensity*** characteristics. Historically, the perspective has been that a CPU-intensive workload implies ***Low Relative NEST Intensity*** and an I/O-intensive workload implies ***High Relative NEST Intensity***. This relationship often proves true, thus providing a method to assign the workload category that best represents a production workload based on its ***DASD I/O Rate***. Past experience has shown that 70-80% of all production workloads fall in the ***Average Relative NEST Intensity*** category.

While the above relationships are generally true, CPU-intensive workloads such as batch can exist that have ***High Relative NEST Intensity***. And I/O-Intensive workloads such as online or transaction oriented database applications can exist that have ***Low Relative NEST Intensity***. Therefore, the historical rule-of-thumb may not always result in choosing the workload category that will best represent production work. A more reliable method would be to understand the actual ***Relative NEST Intensity*** of the production workload. On System z10 and later processors running z/OS, ***CPU Measurement Facility*** (SMF type 113) records can be captured, providing hardware counter data. Analysis of this data should provide an understanding of the actual ***Relative NEST Intensity*** of the production work, therefore becoming reliable for selecting the most appropriate LSPR workload category.

LSPR data is intended for making capacity comparisons for between processor models assumed to be running the same SCP and workload environment. LSPR data is not intended for making comparisons between versions of z/OS or any other software. With **zPCR**, LSPR data is assessed such that it can be used to represent a given production workload for the purpose of projecting the capacity for a proposed system relative to that of the currently installed system.

Two separate tables comprise LSPR data.

1. The ***Single-Image Capacity Ratios Table*** is derived from measurements made on single-partition hardware configurations. This table provides the basis for **zPCR** algorithms concerning the effects of partitioning. Since this table represents a single partition with a LCP:RCP ratio of 1.00, partitioning effects are essentially ignored.
2. The ***Multi-Image Capacity Ratios Table*** is built using **zPCR**'s ***LPAR Configuration Capacity Planning*** function to define partition configurations considered typical for each processor model. Since the ***Multi-Image Table*** does consider partitioning effects, it should be the preferred source when generalizing on IBM System z capacity relationships. Note that the same SCP and workload is assumed to be running in every partition. zAAP, zIIP, IFL, and ICF partitions are not considered.

To derive capacity for any specific LPAR configuration, **zPCR**'s ***LPAR Configuration Capacity Planning*** function should be used to define that configuration. This function allows specification of the various SCP and workload categories for each partitions. In addition, specialty engines such as zAAP, zIIP, IFL, and ICF can be included in the configuration. Capacity results will likely differ (often considerably) from those obtained from the ***Multi-Image Table***. Since this **zPCR** function considers all the effects of partitioning, it should be considered the most reliable source of IBM System z capacity relationships.

LSPR data as of 07/22/2010

IBM System z Processor Families Supported

- **zEnterprise 196** (**z196**) Announced 07/22/2010
- **z10-EC** and **z10-BC**
- **z9-EC** and **z9-BC**
- **z990** and **z890**
- **z900** and **z800**

Workload Categories Included

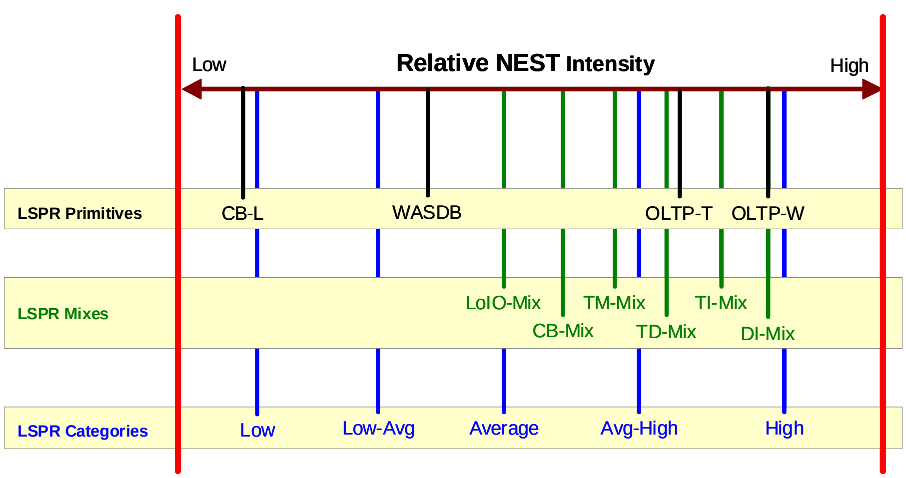
- **z/OS** **Low**
- **z/OS** **Low-Avg** (not part of LSPR, added in **zPCR**)
- **z/OS** **Average**
- **z/OS** **Avg-High** (not part of LSPR, added in **zPCR**)
- **z/OS** **High**
- **z/VM** **High/LV** (based on ***WASDB/LVm*** workload primitive)
- **Linux** **Low/L** (based on ***WASDB/L*** workload primitive)

Former zPCR Workload Mixes Convert to New Workload Categories

- **LoIO-Mix** **Average**
- **CB-Mix** **Average**
- **TM-Mix** **Avg-High**
- **TD-Mix** **Avg-High**
- **TI-Mix** **Avg-High**
- **DI-Mix** **High**

zPCR Workload Characterization for z/OS

Positioning New Workload Categories to Former zPCR Workloads



Overview of zPCR Function

zPCR Version 7.1, dated 07/22/2010

Capacity expectation for processor models can be obtained from **zPCR** in three ways:

1. ***LSPR Multi-Image Processor Capacity Ratios Table***

   This table assumes an average LPAR partition configuration considered to be representative for each processor model. Capacity can only be viewed for General Purpose CPs running the z/OS workload categories. The same workload is assumed to be running in every partition. **This table provides a generalized view of relative capacity for average LPAR configurations**.

2. ***LSPR Single-Image Processor Capacity Ratios Table***

   This table assumes a single partition on each processor model exploiting all the CPs (up to the limit supported by the SCP). Separate tables are available for each supported LSPR version, which include only the processor models that are published with that version. Capacity can be viewed for General Purpose CPs (z/OS, z/VM, and Linux) or for IFLs (z/VM and Linux only). **This table provides a simplistic view of relative capacity for single-partition configurations**.

3. ***LPAR Configuration Capacity Planning Function***

   This function is used to define a specific LPAR host processor and its entire partition configuration. Any legitimate partition configuration is supported.

   General Purpose CPs, zAAPs, zIIPs, IFLs, and ICFs (where they are supported) are configured to the LPAR host. General Purpose, IFL, and ICF partitions are defined. Logical CPs are assigned to partitions, which can be dedicated or shared, with a weight and capping assignment. z/OS, z/VM, VSE/ESA, Linux, and CFCC workloads can be assigned to General Purpose partitions. zAAP and/or zIIP logical CPs can be associated with General Purpose partitions running z/OS-1.6 or later. z/VM and Linux workloads can be assigned to IFL partitions. CFCC can be assigned to ICF partitions. **This function provides the most accurate view of capacity expectation, since it is specific to the exact LPAR configuration and to the software environment of each individual partition**.

Only the current LSPR version of z/OS, z/VM, and Linux can be viewed in ***LSPR Processor Capacity Ratios Tables***. Any of the z/OS workload categories can be selected for viewing. For z/VM and Linux, the published LSPR workloads can be viewed in the ***LSPR Single-Image Processor Capacity Ratios Table***. In addition, z/VSE and CFCC workloads are available to assign to partitions when using the ***LPAR Configuration Capacity Planning Function***.

With **zPCR**'s ***LPAR Configuration Capacity Planning Function***, various SCPs and their associated workload categories can be defined to partitions as follows:

General Purpose Partitions		
SCP	**Max LCP**	**Supported Workloads**
z/OS-1.11	80	**Low**, **Low-Avg**, **Average**, **Avg-High**, **High**
z/OS-1.10*	80	
z/OS-1.9*	64	
z/OS-1.8*	32	
z/OS-1.7*	32	
z/OS-1.6*	32	
z/OS-1.5*	16	
z/OS-1.4*	16	
z/VM	32	Low/LV, Low-Avg/LV, Average/LV, Avg-High/LV, **High/LV**
Linux	32	**Low/L**, Low-Avg/L, Average/L, Avg-High/L, High/L
z/VSE	4	Batch, Mixed, Online
CFCC	16	CFCC

z/OS is supported in General Purpose partitions only. The version of z/OS specified will only be used to control the number of logical CPs that can be assigned to the partition, including zAAPs and/or zIIPs. Regardless of the z/OS version chosen, the capacity results will be identical for any given workload since all the capacity data comes from a single LSPR table.

IFL Partitions		
SCP	**Max LCP**	**Selectable Workloads**
z/VM	32	Low/LV, Low-Avg/LV, Avg/LV, Avg-High/LV, **High/LV**
Linux	32	**Low/L**, Low-Avg/L, Avg/L, Avg-High/L, High/L

z/VM and Linux are supported on either General Purpose or IFL partitions. For z/VM, however, to run in an IFL partition, its guests must all be running Linux. For both of these SCPs, equivalents of the z/OS workload categories may be specified.

ICF Partitions		
SCP	**Max LCP**	**Selectable Workloads**
CFCC	16	CFCC

CFCC is supported on either General Purpose or ICF partitions. Only CFCC can run on ICF partitions.

Note: Workloads shown **Bold Blue** can be viewed in the ***LSPR Multi-Image Table*** or the ***LSPR Single-Image Table***. Workloads shown **Bold Brown** can be viewed in the ***LSPR Single-Image Table***.

Choosing a Workload Category

When using **zPCR**'s ***LPAR Configuration Capacity Planning*** function, a workload must be chosen for the SCP defined to each partition. For z/OS, multiple workload categories are provided. The ***DASD I/O Rate*** of a partition is often useful for choosing the most likely workload category to represent the production workload. The workload assignment will be based on (1) the ***LPAR Host*** processor model, (2) the partition ***DASD I/O Rate***, and (3) the partition ***Utilization***. The workload determination can be made by using **zPCR**'s ***Workload Selection Assistant***. When the three metrics are entered, a pointer will indicate the best workload category match. The workload determination can also be made manually (see Based on DASD I/O Rate, below).

Currently, no rule-of-thumb exists for the other SCPs such as z/VM, Linux, z/VSE, and CFCC. In **zPCR**, workload categories similar to those for z/OS have been defined for z/VM and Linux. Therefore, some judgment will be required. For a production workload whose internal characteristics are not easily determined, the ***Average*** workload category should normally be selected.

Based on DASD I/O Rate

The ***Average*** workload category is intended to represent production workloads where the partition's ***DASD I/O Rate per consumed MSU*** is less than 30. The actual calculation is the ***DASD I/O Rate divided by (MSU Rating times Partition Utilization)***. This category is generally representative of 60-70% of all production workloads

The MSU value in the calculation is intended to provide a generalized capacity value for each processor. However, the MSU values for the System z10, z9, z990, and z890 are discounted (providing improved software price/performance). Therefore, they must be adjusted to provide capacity consistency.

Processor Family	MSU Adjust Factor
z196	0.73
z10	0.73
z9	0.81
z990	0.90
z890	0.90
All others	1.00

In these cases, you need to divide the MSU value shown in **zPCR** by the ***MSU Adjust Factor*** to get the MSU value to be used for the ***Average*** workload test.

Example for a Partition	
LPAR Host processor model	**2084-301**
MSU rating (discounted)	**70**
Utilization	**64%**
DASD I/O Rate per Second	**1,395**

Calculations	
Adjusted MSU	70 ÷ 0.90 = **77.8**
MSUs Consumed	0.64 × 77.8 = **49.8**
DASD I/O Rate per Consumed MSU	1,395 ÷ 49.8 = **28.0**

This workload qualifies for the ***Average*** workload category since it demonstrates less than 30 DASD I/Os per Consumed MSU. The table below can be used to identify the workload category assignment based on the ***DASD I/O Rate per Consumed MSU*** that was computed.

Workload Category	DASD I/O Rate per Consumed MSU
Low	**N/A**
Low-Avg	**N/A**
Average	**Less than 30**
Avg-High	**Less than 40**
High	**Less than 50**

zPCR's ***Workload Selection Assistant*** automates the algorithm shown above, revealing the appropriate workload category choice with a pointer. To invoke this capability, from the ***Function Selection*** window or the ***Advanced-Mode Control Panel*** window, click on **Workload Select** under **CPcalculator** on the menu-bar.

When partitions are loaded into **zPCR** from RMF data or from an EDF file, the ***LPAR Host*** processor model and ***Partition Utilizations*** are already know. On the ensuing window, an input field is provided for the partition's ***DASD I/O Rate***. When entered, the ***Workload Selection Assistant*** is automatically invoked, and the appropriate workload category is filled in for that partition. If nothing is entered, the default workload category is assigned.

Note: The I/O rate for any production workload is only a superficial indicator of the production workload's actual hardware behavior. Therefore, there will be some cases where the workload category determined via the ***DASD I/O Rate*** test may not be the most relevant choice. Therefore, **zPCR capacity projections should always be considered with a ±5% margin-of-error**.

Based on CPU Measurement Facility Data

When a z/OS production workload is running on a z10 processor (or later), CPU Measurement Facility (SMF type 113; hardware counter) data can be captured for a partition. When used, this data will likely provide a more reliable way to choose the appropriate LSPR workload category that best represents that partition's production workload. When SMF type 113 data is captured, it must be reduced to an EDF file with the CP2KEXTR program running in a z/OS partition. A separate EDF must be created for each partition where SMF 113 data was captured.

With the EDF downloaded to a PC, **zPCR** can use it to create the entire configuration (similar to using RMF data). For each partition where SMF type 113 records exist, the appropriate workload category is automatically indicated. The selection currently does not replace the choice determined via the ***DASD I/O Rate*** method or the default assignment. Rather, it is displayed as a ***113 Hint***. If the ***113 Hint*** workload category is to be assigned to a partition in **zPCR** itself, it can be done manually from the ***Partition Definition*** window or the ***Partition Detail Report*** window. At some point in the future, after considerable experience has been gained, the ***113 Hint*** workload category will be automatically assigned to the partition.

z10 Upgrade Considerations

Capacity planning for z10 processor upgrades should be done in the same manner as for any other processor family. As for all **zPCR** studies, capacity projections should be considered with a ±5% margin-of-error.

When the following three conditions all exist, an additional step is warranted. These conditions are:

1. **The intended upgrade is for equal or near equal capacity**
   (i.e., 5% or less additional capacity)
2. **The current processor's utilization is high**
   (90% or higher)
3. **There is very little discretionary work**
   (i.e., 85% or more is high-priority must-complete work).

If you intend to commit to a z10 upgrade where the above 3 conditions all apply, contact:

zPCR@US.IBM.COM

Please provide a contact (with phone number if possible) and details concerning the planned upgrade. The capacity plan will be reviewed to determine if any alternative actions might need to be considered.

LSPR Workload Categories

LSPR data as of 07/22/2010

Low

The ***Low "relative NEST intensity"*** workload category represents light use of the processor's memory hierarchy. This category would be similar to past high scaling workload primitives. For more detail on how this workload category is defined, see Overview of LSPR Data.

Low-Avg

The ***Low-Avg "relative NEST intensity"*** workload category is defined to be midway between the ***Low*** and ***Average*** categories. In **zPCR**, this category has been added to the three published LSPR categories to provide additional granularity for capacity planning purposes. For more detail on how this workload category is defined, see Overview of LSPR Data.

Average

The ***Average "relative NEST intensity"*** workload category represents average use of the processor's memory hierarchy. This category would be similar to the ***LoIO-Mix*** workload used with previous versions of **zPCR**. ***Average*** is expected to be representative of the majority of production workloads. For more detail on how this workload category is defined, see Overview of LSPR Data.

Avg-High

The ***Avg-High "relative NEST intensity"*** workload category is defined to be midway between the ***Average*** and ***High*** categories. In **zPCR**, this category has been added to the three published LSPR categories to provide additional granularity for capacity planning purposes. For more detail on how this workload category is defined, see Overview of LSPR Data.

High

The ***High "relative NEST intensity"*** workload category represents heavy use of the processor's memory hierarchy. This category would be similar to lower scaling workloads such as ***DI-Mix*** used with previous versions of **zPCR**. For more detail on how this workload category is defined, see Overview of LSPR Data.

LSPR Workload Primitives (MVS)

FPC1 (OS/390)

FPC1 is an engineering and manufacturing batch jobstream, representative of scientific data development work. It includes jobs doing static analysis, dynamic analysis, computational fluid dynamics, nuclear fuel calculations, and circuit analysis. Being scientific work, it makes heavy use of floating point. While FPC1 is a processor intensive workload, it also places significant demands on external resources such as central storage, channels, and DASD.

Measurements are made with OS/390, DFSMS, JES2, and RMF. FORTRAN H Extended and MSC/NASTRAN™ software is also used by the jobstream. Performance data collected consists of the usual SMF data, including type 30 records (workload data), and RMF data.

The FPC1 job stream is replicated to assure a reasonable measurement period, and the job queue is pre loaded. Enough initiators are activated to ensure a high steady-state utilization level, approaching 100%. The measurement is started when the job queue is released, and ended as the last job completes. Each job in the job stream uses its own datasets in order to eliminate dataset contention.

Note: Since the LSPR is intended to characterize System/390 and z/Architecture processors with standard features, FPC1 has not been implemented to take advantage of an integrated vector facility where available.

ODE-B (z/OS)

ODE-B reflects the billing process used in the telecommunications industry. This is a multi-step approach which includes the initial processing of Call Detail Records (CDR), the calculation of the telephone fees, and the insertion of the created telephone bills in a database. The CDRs contain the details of the telephone calls such as the source and target numbers along with the time and the duration of the call. The CDRs are stored in flat files within a zFS file system. A feeder application reads the CDRs from the files, converts them into XML format and sends them to a queue. An analyzer application reads the messages from the queue and performs analysis on the data. During the analysis further information is retrieved from the relational database, and the same database is subsequently updated with the newly created telephone bill and new records for each call. The feeder and the analyzer applications are implemented as enterprise java beans (EJB) in IBM WebSphere Application Server for z/OS. Using the concept of multi-servant regions, which is unique to the z/OS implementation of WebSphere Application Server, the threads of the feeder and the analyzer applications are distributed over several java virtual machines (JVM). The WebSphere internal queuing engine is used as the queue for the message transport between the feeder and analyzer.

CB-L (z/OS); CBW2 (OS/390)

CB-L is a commercial batch workload with long job steps, reflective of large batch jobs with fairly heavy CPU processing. The job stream consists of 1 or more copies of a set of batch jobs. Each copy consists of 22 jobs, with 157 job steps. These jobs are more resource intensive than jobs in the CB-S workload (discussed below), use more current software, and exploit ESA features. See table 15 for a list of some of the performance metrics for the LSPR batch workloads. The work done by these jobs includes various combinations of C, COBOL, FORTRAN, and PL/I compile, link-edit, and execute steps. Sorting, DFSMS utilities (e.g. dump/restore and IEBCOPY), VSAM and DB2 utilities, SQL processing, SLR processing, GDDM™ graphics, and FORTRAN engineering/scientific subroutine library processing are also included. Compared to CB-S, there is much greater use of JES processing, with more JCL statements processed and more lines of output spooled to the SYSOUT and HOLD queues. This workload is heavily DB2 oriented with about half of the processing time performing DB2 related functions.

Measurements are made with z/OS, OS/390, DFSMS, JES2, RMF, and RACF. C/370, COBOL II, DB2, DFSORT, FORTRAN II, GDDM, PL/I, and SLR software are also used by the job stream. Access methods include DB2, VSAM, and QSAM. SMS is used to manage all data. Performance data collected consists of the usual SMF data, including type 30 records (workload data), and RMF data.

The CB-L job stream contains sufficient copies of the job set to assure a reasonable measurement period, and the job queue is pre loaded. Enough initiators are activated to ensure a high steady-state utilization level, approaching 100%. The number of initiators is generally scaled with processing power to achieve comparable tuning across different machines. The measurement is started when the job queue is released, and ended as the last job completes. Each copy of the job set uses its own datasets, but jobs within the job set share data.

CB-J (z/OS-1.6)

CB-J is a java commercial batch workload which reflects the production environment of a clearing bank that uses a collection of java classes working on a DB2 database and a set of flat files in z/OS. JavaBatch is a native, standard Java application that can be run standalone on a single JVM (Java Virtual Machine) or in parallel to itself on multiple JVMs. Each of the parallel applications instances can be tuned separately. All parallel applications are working on the same set of flat files and database tables.

The JavaBatch application is based on a Java-JDBC-framework of an external banking software vendor and has been enhanced and adapted using the WebSphere Application Developer tool. Various properties such as number of banks, number of accounts, and more, can be adapted for the specific runtime environment in a special properties file, by keeping the java application unchanged. The JavaBatch application allows a user to perform the following activities:

1. Initialize the working database
2. Create a set of flat files, each containing several hundredths to thousands of payments
3. Read the flat files, perform various syntax-checks and validation for each payment and store the payments to the working database
4. Read the payments from the database and route them to destination bank's flat files.

CB-S (z/OS-1.4); CB84 (OS/390)

CB-S is a commercial batch workload with short job steps, reflective of small batch jobs with fairly light CPU processing. The job stream consists of 1 or more copies of a set of batch jobs. Each copy consists of 130 jobs, with 496 unique job steps. Table 15 lists some performance metrics for the two LSPR batch workloads and can be used to compare the two. The work done by these jobs includes various combinations of compile, link-edit, and execute steps. Utility jobs, primarily for data manipulation, are also included.

Measurements are made with z/OS, OS/390, DFSMS, JES2, and RMF. Assembler H, COBOL/VS, PL/I Optimizing Compiler, and DFSORT software is also used by the job stream. Access methods include BSAM, QSAM, BDAM, and VSAM. SMS is used to manage non-system data. Performance data collected consists of the usual SMF data, including type 30 records (workload data), and RMF data.

The CB-S job stream contains sufficient copies of the job set to assure a reasonable measurement period, and the job queue is pre loaded. Enough initiators are activated to ensure a high steady-state utilization level, approaching 100%. The number of initiators is generally scaled with processing power to achieve comparable tuning across different machines. The measurement is started when the job queue is released, and ended as the last job completes. Each job in the job stream uses its own datasets in order to eliminate dataset contention.

TSO (OS/390)

The TSO on-line workload is representative of the work done by a typical OS/390 end-user community developing and testing programs interactively using ISPF/PDF. Workload activities include editing and browsing source data, compilation, execution, program testing, graphics, and Info/Management transactions. CLISTs are both explicitly and implicitly invoked.

Measurements are made with OS/390, DFSMS, JES2, RMF, VTAM, and TSO/E. The workload also uses ISPF/PDF, COBOL/VS and COBOL Prompter, PL/I Checkout Compiler, Script/VS, GDDM, and INFO/SYS program products. SMS is used to manage non-system data. RACF is used to control access to user datasets. Performance data collected consists of the usual SMF data, including type 30 records (workload data), and RMF data.

TSO is measured by logging on the number of terminals (end-users) required to reach the target utilization. Measurements are made at 70% and 90% processor busy. There are 25 different scripts, each consisting of a related set of activities in the form of TSO commands. The average think time is 15 seconds. A negative exponential distribution is applied to the think times so that half are 2 seconds or less, and 80% are 8 seconds or less. After a terminal completes a script, the entire process is repeated. TSO is measured as a steady-state workload over an elapsed period deemed adequate to produce a repeatable sample of work.

Historically, TSO measurements were made with simulated local VTAM terminals. Due to the greater number of terminals required on larger systems, measurements are made with simulated remote VTAM terminals for OS/390.

WASDB (z/OS)

WASDB is a WebSphere Application Server and Data Base workload run under z/OS. It reflects a new e-business production environment that uses WebSphere applications and a DB2 data base all running in z/OS.

WASDB is a collection of Java classes, Java Servlets, Java Server Pages and Enterprise Java Beans integrated into a single application. It is designed to emulate an online brokerage firm. WASDB was developed using the IBM VisualAge™ for Java and WebSphere Studio tools. Each of the components is written to open Web and Java Enterprise APIs, making the WASDB application portable across J2EE-compliant application servers.

The WASDB application allows a user, typically using a web browser, to perform the following actions:

- Register to create a user profile, user ID/password and initial account balance.
- Login to validate an already registered user.
- Browse current stock price for a ticker symbol.
- Purchase shares.
- Sell shares from holdings.
- Browse portfolio.
- Logout to terminate the user’s active interval.
- Browse and update user account information.

CICS/DB2 (OS/390)

The CICS/DB2 on-line workload was designed to represent customer's daily business by simulating the placement of orders and delivery of products, as well as business function like supply and demand management, customer demographics and item selling hit list information. The workload consists of ten unique transactions.

CICS is used as a transaction monitor system. It provides both an API for designing the dialogue panels and parameters to drive the interface to the DB2 database. The interface between the two subsystems is fully supported by S/390 and exploits N-Way designs. CICS functions like dynamic workload gathering and function shipping are not exploited in this workload. The CICS implementation uses an MRO model, which is managed by CP/SM. The number of AOR (Address Owning Region) and TOR (Terminal Owning Region) used, depends on the number of engines of the processor under test. The ratio between TOR and AOR is 1:3. The utilization of the TOR and the AOR regions is kept under 60%.

The application database is implemented in a DB2 subsystem. One of the major design efforts was to achieve a read-to-write ratio exhibited by OLTP customers. Several data center surveys indicate an average read-to-write ratio to be in the range of 4:1 - 6:1. The read-to-write ratio is an indication of how much of the accessed data are changed as well. For this CICS/DB2 workload implemented on a S/390 or z/Architecture system and using DB2 as database system, an approximation of the read-to-write ratio is the ratio of SQL statements performing 'read' operation, like select, fetch, open cursor to the 'write' SQL statements, like insert, update, delete.

To reduce the number of database locks and the inter system communication required for each database update and to preserve local buffer coherency in data sharing environments, DB2 type 2 indexes have been used. Additionally, row-level-locking has been introduced for some tables. Each table and index is buffered in separate buffer pools for easy sizing and control.

CICS (OS/390)

The CICS online workload consists of light to moderate transactions from diverse business applications, including order entry, stock control, inventory tracking, production specification, hotel reservations, banking, and teller system. The application programs are written in COBOL or Assembler. These applications are functionally similar, but not identical, to the IMS workload applications, described below.

For measurements on MVS/SP 4.2.0, the CICS workload contains six sets of 17 unique transactions. One set of transaction programs is run below the 16 megabyte line, and the other five are run in 32-bit mode above the line. For measurements on OS/390, there are twelve sets of transactions, with two sets running below the line and 10 sets above the line. Each set of transactions has different transaction names and IDs, and uses different program load modules, buffer pools, datasets, etc. The effect is that the OS/390/CICS system manages 102 or 204 transactions in each copy of the CICS workload.

CICS measurements are done in an MRO environment. For MVS/SP 4.2.0 measurements, a three region "MROplex", consisting of a TOR (terminal-owning region), AOR (application-owning region), and FOR (file-owning region), is run for each engine in the processor being measured. For OS/390, each MROplex consists of four regions: TOR, AOR, FOR, and a combined A/FOR, as this configuration results in CICS processing characteristics that are more typical of customer environments, and an MROplex is run for each one or two engines in the processor being measured. Each MROplex runs a copy of the workload. All CICS address spaces are non-swappable.

The CICS workload uses VSAM datasets, and data in storage is implemented with large buffer pools. Measurements are made with OS/390, DFSMS, JES2, RMF, VTAM, and CICS/OS/390. Performance data collected consists of CICS shutdown statistics and the usual SMF data, including type 30 records (workload data), and RMF data.

CICS is measured by logging on a predetermined number of terminals, each of which starts executing scripts consisting of end-user actions. Once the logons are complete, the average think time is adjusted to provide a transaction rate that causes the processor to reach the target utilization level. After appropriate stabilization periods, measurements are made at 70% and 90% processor busy. The average think time is approximately 16 seconds for the 70% run and 12 seconds for the 90% run, with a uniform distribution ranging from 75% to 125% of the average. CICS is measured as a steady-state system over an elapsed period deemed adequate to produce a repeatable sample of work.

DB2 (OS/390)

The DB2 workload consists of light to moderate transactions from two predefined, well-structured (not ad hoc query) applications, inventory tracking and stock control. IMS/DC is used as the transaction manager. The applications are functionally similar, but not identical, to two of the IMS/DLI and CICS applications. The DB2 workload contains sets of seven unique transactions, each of which has different transaction names and IDs. Conversational and wait-for-input transactions are not included in the DB2 workload.

The number of copies of the workload and the number of IMS Message Processing Regions (MPRs) configured is adjusted to ensure that the IMS and DB2 subsystems are processing smoothly, with no unnecessary points of contention. No Batch Message Processing regions (BMPs) are run. IMS and DB2 address spaces are non-swappable.

There are two DB2 databases, consisting of 11 tables for inventory tracking and 5 tables for stock control. Each table is in its own table space; partitioned and segmented table spaces are not used. Each table contains from 1 to 5 indexes, to ensure good performance, and is defined with “Lock Size ANY”. All DB2 referential integrity constraints are defined with the “Delete Restrict” rule. The DB2 workload has moderate I/O characteristics.

Measurements are made with OS/390, DFSMS, JES2, RMF, VTAM, IMS/ESA, and DB2. IMS coat-tailing (enabling reuse of a module already in storage) is not used since this activity is so sensitive to processor utilization, it could cause distortion when comparing ITRs between faster and slower processors. The DB2 LSPR workload does not take advantage of the DB2 sort assist feature found on the high-end ES/3090™ and ES/9000 processors, because the transactions do not happen to invoke sorts. Performance data collected consists of IMSPARS, DB2PM, and the usual SMF data, including type 30 records (workload data), and RMF data.

DB2 is measured by logging on the number of terminals required to reach the target utilization. Measurements are made at 70% and 90% processor busy. Each terminal starts executing scripts consisting of end-user actions. The average think time is approximately 4 seconds, with a uniform distribution ranging from 75% to 125% of the average applied. After the logons are complete, a stabilization period is used to ensure that the DB2 buffers are fully primed. Once the system reaches steady-state, the measurement is made for an elapsed time deemed adequate to produce a repeatable sample of work.

OLTP-W (z/OS)

OLTP-W is a web-enabled on-line workload, run under z/OS. It reflects a production environment that has web-enabled access to a traditional data base. For the LSPR, this has been accomplished by placing a WebSphere front-end to connect to the LSPR CICS/DB2 workload (described below).

The J2EE application for legacy CICS transactions was created using the CICS Transaction Gateway (CTG) external call interface (ECI) connector enabled in a J2EE server in WebSphere for z/OS V4.0.4. The application uses the J2EE architected Common Client Interface (CCI). Clients access WebSphere services using the HTTP Transport Handlers. Then the appropriate servlet is run through the webcontainer, which call EJB's in the EJB Container. Using the CTG external call interface (ECI) CICS is called to invoke DB2 to access the database and obtain the information for the client.

For a description of the CICS and DB2 components of this workload, please see the CICS/DB2 workload description further below.

OLTP-T (z/OS); IMS (OS/390)

OLTP-T is a traditional on-line workload run under z/OS. It consists of light to moderate IMS transactions from DLI applications covering diverse business functions, including order entry, stock control, inventory tracking, production specification, hotel reservations, banking, and teller system. These applications all make use of IMS functions such as logging and recovery. The workload contains sets of 12 (17 for OS/390 Version 1 Release 1 and earlier) unique transactions, each of which has different transaction names and IDs, and uses different databases. Conversational and wait-for-input transactions are included in the workload.

The number of copies of the workload and the number of Message Processing Regions (MPRs) configured is adjusted to ensure that the IMS subsystem is processing smoothly, with no unnecessary points of contention. No Batch Message Processing regions (BMPs) are run. IMS address spaces are non-swappable.

This IMS workload accesses both VSAM and OSAM databases, with VSAM indexes (primary and secondary). DLI HDAM and HIDAM access methods are used. The workload has a moderate I/O load, and data in memory is not implemented for the DLI databases.

Measurements are made with z/OS, OS/390, DFSMS, JES2, RMF, VTAM, and IMS/ESA. IMS coat-tailing (enabling reuse of a module already in storage) is not used; since this activity is so sensitive to processor utilization, it could cause distortion when comparing ITRs between faster and slower processors. Beginning with OS/390 Version 1 Release 1, measurements were done with one or more control regions. The number of data base copies, MPR's, and control regions (within the limits of granularity) are scaled with the processing power of a particular machine in-order to assure equal and normalized tuning. Performance data collected consists of IMS PARS, and the usual SMF data, including type 72 records (workload data), and RMF data.

IMS is measured by logging on a predetermined number of terminals, each of which starts executing scripts consisting of end-user actions. Once the logons are complete, the average think time is adjusted to provide a transaction rate that causes the processor to reach the target utilization level. After appropriate stabilization periods, measurements are made at 70% and 90% processor busy. The average think time is approximately 6.1 seconds (14 seconds for OS/390 Version 1 Release 1) for the 70% run, and 4.9 seconds (11 seconds for OS/390 Version 1 Release 1) for the 90% run, with a uniform distribution ranging from 75% to 125% of the average applied. IMS is measured as a steady-state system over an elapsed period deemed adequate to produce a repeatable sample of work

EAS-DB (z/OS); R/3-DB (OS/390)

EAS-DB is an "Enterprise Application Solution Data Base" workload. It reflects the z/OS database server (DB) portion of an Enterprise Application Solution. That is, the DB resides in a z/OS image while the application server and presentation server reside outboard. The EAS application used for this workload is the mySAP (previously known as SAP R/3) product produced by SAP AG using DB2 for z/OS. The workload is derived from the SAP AG defined Sales and Distribution (SD) workload. However it is not based on SAP AG certified benchmark results. Specifically the results show an mySAP DB in a 3-tier client/server configuration. The other two tiers, application server and presentation server, are not represented in this data.

The workload consists of interactive users repetitively creating a customer order with five items, creating a delivery of the order, displaying the order, changing the delivery, posting, listing orders, and creating an invoice. The steps involved in doing this are called dialogue steps (or dialogue screens to the presentation server) and are treated as transactions. There is a constant user think time of ten seconds. The third tier DB server doesn't see either users or dialogue steps. The work comes into the third tier from the application servers through DB2 Connect as remote SQL to DB2. Essentially DB2 is functioning as a SQL engine. Several pSeries servers function as application servers and simulate the presentation servers.

The software used is a typical complement of z/OS, DFSMS, JES2, RMF, DB2PM, and DB2. The specific software levels used are z/OS V1R6, DB2 V8, DB2 Connect V8, mySAP 4.70 DB, 6.40 kernel, and ECC 5.0.

As with all the LSPR workloads, the EAS-DB workload was measured holding the mix of applications constant on all processors, with only the amount of work executed adjusted for different processor capacities. While this provides valuable data to demonstrate relative processor capacities, it may not accurately reflect all workload growth scenarios. Workload growth may come in two flavors, "more of the same" or "adding new applications." The EAS environment is particularly prone to the latter case where exploiting new functions and features often drive the workload growth, and the capacity required to support this type of workload growth may not follow the ratios obtained based on homogeneous application growth. Therefore, it is highly recommended that capacity sizing for EAS application scenarios be done through your local Techline EAS Sizing Team.

LSPR Workload Primitives (VM)

CMS1 (VM/ESA)

The CMS1 on-line workload is designed to represent a VM/CMS end-user community. Processor time per command, I/Os per command, T/V ratio, and think time distribution are similar to those observed for actual VM production systems running large numbers of CMS users.

Each user runs as a separate virtual machine under VM, which IPLs CMS and enters a variety of CMS and CP commands. Each has access to its own read/write minidisks, and a shared read-only minidisk containing all of the data files required by the script activities. In addition, each user has access to shared minidisks containing the CMS system and containing the software products required by the workload. All user accessible software is installed as shared code where supported.

Software used includes z/VM or VM/ESA, TCP/IP VM, VS COBOL II, VS FORTRAN, IBM High Level Assembler, and OS PL/I. SCP exploitation includes the use of minidisk caching, spooling, paging, and virtual disks in storage. VM Monitor data is collected and post-processed by VMPRF.

CMS1 is measured by logging on the number of terminals (end-users) required to reach the 90% processor utilization target. There are 14 different scripts, each containing a related set of activities in the form of end-user commands. The average think time is 9 seconds. CMS1 is measured as a steady-state workload over an elapsed period deemed adequate to contain a repeatable sample of work.

Note CMS1 is no longer included with any LSPR tables in **zPCR**.

WASDB/LVm (z/VM)

The WASDB/LVm workload reflects a server consolidation environment where each server is running a full function application. For LSPR purposes, this was accomplished by taking the WASDB workload (described above under ***MVS LSPR Workloads***), and then replicating the Linux guest a number of times, based on the N-way of the processor being measured. Guest pair activity was then adjusted to achieve a constant processor utilization for each N-way. Thus, the ratios between processors of equal N-way are based on the throughput per guest rather than the number of guests.

See ***WASDB - WebSphere Application Server and Data Base*** for a detailed description of the WASDB workload.

LSPR Workload Primitives (Linux)

WASDB/L (Linux)

The WASDB/L workload reflects an e-business environment where a full function application is being run under Linux on System z in an LPAR partition. For LSPR purposes, this was accomplished by taking the WASDB workload (described above under ***MVS LSPR Workloads***), and converting it to run both application and data base server in a single Linux on System z image. The WASDB/L workload is basically the same as the WASDB workload on z/OS with the exception of being enabled for Linux on System z.

See ***WASDB - WebSphere Application Server and Data Base*** for a detailed description of the WASDB workload.

EAS-AS/L (Linux)

EAS-AS/L is an "Enterprise Application Solution Data Base" workload. It reflects the Application Server (AS) portion of an Enterprise Application Solution running in a Linux environment. That is, the AS resides in a Linux for zSeries or Linux for System z image while the database server and presentation server reside outboard. The EAS application used for this workload is the SAP product produced by SAP AG using DB2 for z/OS. The workload is derived from the SAP AG defined Sales and Distribution (SD) workload. However, it is not based on SAP AG certified benchmark results. Specifically, the results show an SAP AS in a 3-tier client/server configuration. The other two tiers, database server and presentation server, are not represented in this data.

This workload consists of interactive users repetitively creating a customer order with five items, creating a delivery of the order, displaying the order, changing the delivery, posting, listing orders, and creating an invoice. The steps involved in doing this are called dialogue steps (or dialogue screens to the presentation server) and are treated as transactions. There is a constant user think time of ten seconds. The AS resides physically between the presentation server and the database server. It contains all the business logic to interpret screen inputs, produce SQL requests, send them to the database server, interpret the results, and send updates to the presentation server. The AS in the SD workload typically requires much more CPU power than the database server. A separate System z server functions as the database server and its processing requirements are not included in these results.

The software levels used are SuSE Linux Enterprise Server 9 and SAP kernel 6.40 with 4.70 DB and applications.

As with all the LSPR workloads, the EAS-AS/L workload was measured holding the mix of applications constant on all processors, with only the amount of work executed adjusted for different processor capacities. While this provides valuable data to demonstrate relative processor capacities, it may not accurately reflect all workload growth scenarios. Workload growth may come in two flavors, "more of the same" or "adding new applications". The EAS environment is particularly prone to the latter case where exploiting new functions and features often drive the workload growth, and the capacity required to support this type of workload growth may not follow the ratios obtained based on homogeneous application growth. Therefore, it is highly recommended that capacity sizing for EAS application scenarios be done through your local Techline ISV Solutions Sizing Team.

LSPR Workload Primitives (VSE)

CICS (VSE/ESA)

The VSE/ESA CICS on-line workload consists of light to moderate transactions from diverse business applications including order entry, stock control, inventory tracking, production specification, banking, hotel reservations, and teller system. The application programs are written in COBOL or Assembler. These applications are functionally similar to those used for the OS/390 CICS workload, described earlier. However, the workload similarity ends here; the number of transactions managed by CICS and the rest of the software environment are extremely different. The VSE CICS workload consists of 17 unique transactions, using various combinations of CICS functions.

Two independent and equivalent CICS partitions are run in order to effectively utilize the processor. Each has access to its own datasets, to avoid potential dataset I/O contention. The CICS workload accesses 16 VSAM KSDS datasets, and has a moderate I/O load.

Software used includes VSE/ESA, CICS/VSE™, ACF/VTAM™, and VSE/VSAM. SCP exploitation includes the use of VSE/POWER and of ACF/VTAM, each in their own address spaces, allowing virtual storage constraint relief. Access methods used include SAM and VSAM.

CMF data, used to report CICS response time and total transaction counts, is logged and then processed by the CICSPARS post processing facility. Goal System's EXPLORE™ is also used to gather system performance data including CPU utilization, channel path utilization and DASD utilization.

CICS is measured by logging on a predetermined number of terminals, each of which starts executing scripts consisting of end-user actions. Once the logons are complete, the average think time is adjusted to provide a transaction rate that causes the processor to reach the target utilization level. After appropriate stabilization periods, measurements are made at 70% and 90% processor busy. The average think time is approximately 15 seconds for the 70% run and 11 seconds for the 90% run, with a uniform distribution ranging from 75% to 125% of the average applied. CICS is measured as a steady-state system over an elapsed period deemed adequate to produce a repeatable sample of work.

Note The VSE CICS workload is no longer included with any LSPR tables in **zPCR**. However, 3 z/VSE workload environments have been defined for use in **zPCR**'s ***LPAR Configuration Capacity Planning*** function. These workloads (**Batch**, **Mixed**, and **Online**) can be defined to any partition running on General Purpose CPs. They are based on equivalent z/OS workload capacity relationships for 1-way processors and N-way ratios for up to 4 CPs, intended to represent that those of the VSE Turbo Dispatcher.