The IBM PowerVP Investigation Guide

Table of Contents

Purpose of this Investigation Guide

Once you have PowerVP (Power Virtualization Performance) installed and monitoring your Power Systems, this investigation guide will help you better understand the information available and optimize the performance of your systems.

Please use the PowerVP Installation and User Guide to assist you with the prerequisites for using this investigation guide. That document will provide you with:

- *Power Systems hardware requirements for models that support running PowerVP*
- *Requirements for operation systems for AIX, IBM i, Linux, or VIOS*
- *Requirements for firmware levels*
- *Installation of PowerVP*
- *Configuration PowerVP for monitoring and customizing it for your environment*
- *Starting/stopping PowerVP monitoring*
- *Navigating within and between PowerVP displays*
- *High-level definitions of PowerVP functions*
- *Real-time monitoring and replay of a saved log*

This **Investigation Guide** will provide more detailed definitions of the monitored system resources and the performance metrics used. It will attempt to help you define criteria for your utilization thresholds with color illustrations. It will provide some best practices to help you interpret the results and optimize your system performance. You

are encouraged to read this investigation guide like a paper, and not just refer to parts of it as a reference.

PowerVP Synopsis

PowerVP (Power Virtualization Performance) is a new IBM licensed program product than can help you understand and monitor the performance of your IBM Power Systems.

Clients typically understand the performance of a given logical partition with the help of a comprehensive portfolio of OS-based performance tools from AIX, IBM i, and Linux. However, as the Power based systems have evolved, understanding the performance of the entire Power System as it hosts multiple logical partitions has become more complex with the age of virtualization and cloud computing. PowerVP was created to fill this gap as it monitors and illustrates the performance of an entire system (or frame). PowerVP will allow you to monitor overall performance and allow you to drill down into more detailed hardware and software views to help you identify and resolve performance issues and to optimize the performance of your Power system.

PowerVP includes agent components as well as a monitor. The system level agent can run in any partition (AIX, IBM i, Linux, or VIOS). This agent will collect a variety of information from the operating system and PowerVM interfaces to characterize the configuration and performance of the entire system. The GUI will display this data for the system level and hardware node level views. Optional partition-level agents can run in each partition (AIX, IBM i, Linux, VIOS). The GUI will display this data for the partition level drill-down views. The GUI can run on any network connected workstation and will provide you with an easy-to-use experience.

PowerVP illustrates Power Systems hardware topology in conjunction with resource utilization metrics to help you better understand the system. These resources include nodes, processor modules, chips, cores, Powerbus links, memory controller links, GX I/O bus, disk drives, Ethernets, etc. These resource utilizations are portrayed using a colorized heat technique. These colors and thresholds can be customized to suite a specific client's performance requirements in a meaningful way. For example, green may indicate normal, yellow can signal caution, and red may indicate that some resource is extremely busy and may require action. PowerVP also allows mapping between real and virtual processor resources. For example, click on a partition and see which physical cores are associated with that partition.

PowerVP is a real-time monitor that can collect and update the performance information as frequently as every second. The GUI allows you to view this data in real-time. The data can also be recorded in a repository for later playback. During playback, PowerVP provides DVR-like functions to play, fast forward, rewind, jump, pause, or stop. PowerVP utilizes a drill-down approach for performance analysis. The system-level view will illustrate overall system-level performance with processor modules and inter-node

links. Clicking on a specific hardware node will drill into that node showing the utilization of each core and intra-node links. A listing of each partition is present on both views showing the entitlement, utilization, and physical hardware mapping. Clicking on a specific partition will provide its detailed performance statistics including application efficiency. Collectively, these performance metrics can help you optimize performance with resource balancing, improved affinity, and application efficiency.

Many of the functions in PowerVP are suitable and have value for all users (clients, field support). Some of the more detailed metrics (such as CPI and bus utilizations) within PowerVP are aimed at more technical users.

Consider adding PowerVP to your performance toolbox to help simplify the management of your Power system's performance.

What gap does PowerVP fill?

Historically, clients have a detailed understanding of performance within a given partition (LPAR). Each operating system has a host of command line tools that help describe performance for the CPU, virtualization, I/O, etc. Each operating system has one or more monitoring tools that help you understand utilizations, resource consumption, and usage characteristics of your LPAR. This type of performance analysis allows you understand and optimize the performance of your applications running in that partition.

The adoption of virtualization has increased greatly in the last few years. Most systems typically have several partitions; some systems have hundreds of partitions. Many clients are moving towards cloud computing or (PoD) processing on demand which dynamically involves many systems with many partitions. Using traditional OS-based performance tools and monitors are still important for managing/optimizing application performance within an LPAR. However, there is an increased need to understand, manage, and optimize the performance of your overall system (or frame, CEC) or group of systems (or cloud, PoD). In doing this, clients don't want to monitor each LPAR individually and then have to piece this jigsaw puzzle together.

PowerVP fills this gap in providing an easy-to-understand overall view of performance. From a single partition, you can gather information for the entire system using new/enhanced interfaces to the PowerVM hypervisor. With these new interfaces, PowerVP is also able to illustrate resources and their utilization that normally don't appear in traditional performance tools within an LPAR. For example, knowing the utilization of individual processor cores and their mapping to LPARs can help you understand, manage, and balance your system resources. Another good example, knowing the utilization of internal buses (Powerbus, GX I/O bus, and the memory controller bus) can help you understand affinity and optimize performance. Also, PowerVM lets you consume all this information real-time.

Definition of general terms used with PowerVP:

The following terms are used within PowerVP and are described here as they should be interpreted when using PowerVP. They arranged in a logical reading order.

- **System:** A physical system is the entire Power System, including all resources for CPU, memory, storage, etc. This physical system may contain one or more partitions, or virtual systems. Some refer to this as a frame or a CEC. For the purpose of interpreting PowerVP, please do not interchange the terms system (physical) and partition (virtual system).
- **Partition**: A logical partition (LPAR) is division of a system's resources, such that it can run independently with its own operating system. A physical system can have one or more LPARs (virtual systems). These LPARs can be dedicated or shared (capped or uncapped). A hypervisor, like PowerVP, manages these partitions. Please refer to IBM Information Center for PowerVM for more descriptions of these related terms (virtual system, entitlement, hardware thread, VIOS, dedicated donate, processor folding, partition types, etc.).
- **Hardware node**: Except for the smallest Power Systems, there is a componentization of the physical system into books, drawers, or nodes. For example, Power 770/780 has up to four drawers, Power 795 has up to eight books.
- **Socket:** A socket is a physical connection on a Power System that connects a one processor module. These modules can be either an SCM (single chip module) or a DCM (dual chip module).
- **Processor Module:** A processor module is an orderable physical entity that connects to a socket. These processor modules can be in the format of an SCM or a DCM. With POWER7, these modules contain processor cores, caches, and other components. For POWER7, a DCM implies two processor chips.
- **Chip:** A processor chip is physical integrated circuit that contains processor cores and/or caches. POWER7 chips contain up to eight cores with on-chip L1, L2, and L3 caches. This document doesn't describe all Power System configurations; but there are significant differences between POWER4/5/6/7 chips as well as model footprints within those architecture families.
- **Core:** A processor core is a single physical processing unit. With POWER7, up to eight of these cores exist on a single chip. Each core can have up to four hardware threads dispatched to it simultaneously using SMT4. These hardware threads can be called logical cores. Many talk about a system with a total number of physical cores, for example, a 64-core system. LPARs can have an entitlement in terms of a number of cores.
- **CPU:** The term CPU is used to collectively refer to the CPU resources (core, socket, chip, system) for a given entity (partition, system) when discussing metrics like CPU utilization, CPU time, CPU cycles, etc. The term CPU is not used

explicitly as a specific resource name as it is often confusing. Some refer a CPU to a socket, some to a processor module, and some to a processor core.

- **Utilization:** Utilization is a base performance term that is the percentage of time that a resource is busy. It is normally in the form a percentage, typically from 0% to 100%. Of course, some shared LPARs may have a utilization of greater than 100% if it consumes more CPU resources from the shared pool than its entitlement states.
- **CPU utilization**: This term is much more complex than you might expect. It can refer simply to the percentage of time that the CPU resources are busy. However, with the advent of SMT levels (more than one hardware thread dispatched to a core), multi-core systems, and complex processor pipes, CPU utilization becomes more complicated. Each operating system may provide and interpret CPU utilization differently. AIX and IBM i provide utilizations that consider SMT levels and hardware thread dispatch conditions. From this, CPU utilization is rendered where a linear relationship is expected between system throughput and CPU utilization. Of course this comes with many assumptions (sufficient other resources for that workload to scale, only true for the actual workload used to tune the utilization while other workloads may scale significantly differently, and on and on). Linux operation systems currently provide CPU utilizations that are based more on occupancy (hardware thread occupying a given core). The more that you understand on this topic, the more you realize that other metrics are also needed to best understand your system/application (such as scaling characteristics, instructions consumed, run cycles consumed, contention issues, etc.).
- **Powerbus (W, X, Y, Z, A, B, C):** These are a set of links or buses within Power Systems. Within PowerVP, those links that are labeled W, X, Y, or Z are links within a hardware node; those links that are labeled A or B are links between hardware nodes. These Powerbus links carry data between a given chip and other resources outside that chip (cache, memory, I/O). PowerVP portrays these links and their utilizations. Having a higher Powerbus utilization implies that there is a higher rate of data transfer.
- **Memory Controller (MC):** These are a set of links that connect the memory to the socket. These MC buses carry data between the memory controller and the chip. PowerVP portrays these links and their utilizations. Having a higher rate of data transfer implies that there is a higher rate of data transfer.
- **I/O (GX) bus:** These are a set of links or buses within a Power System that connect the I/O subsystems to the chip. These links carry data for storage I/O and network I/O. PowerVP portrays these links, their utilizations, as well as their inbound/outbound data rate. Having a higher GX bus utilization implies that there is a higher rate of data transfer.
- **Cycles per instruction (CPI):** CPI is a standard measurement of application efficiency. It is the number of cycles consumed divided by the number of

(machine) instructions completed. Normally, a lower CPI is better than a higher CPI. A CPI can be measured for a given core, a processor module, a hardware node, or an LPAR with PowerVP. From an LPAR perspective, you can break down CPU utilization, into CPI components (e.g., load/store unit, floating point, global completion table).

- **CPI stack analysis:** CPU utilization can be broken down into CPI components. Load/Store Unit (LSU CPI) reflects the cycles consumed for accessing data (L1, cache, L2 cache, L3 cache, memory). Floating Point (FXU CPI) reflects cycles consumed on executing floating point. Global Completion Table (GCT CPI) reflects cycles consumed waiting on the global completion table for pipelining out-of-order instruction execution. PowerVP analysis typically focuses on LSU CPI.
- **LSU CPI stack analysis:** Normally the largest component of CPU utilization is the LSU CPI for OLTP applications. In other words, accessing data consumes a majority of the CPU resources. A characterization of the time accessing data from L1 cache, L2 cache, L3 cache, and memory; this also notes whether the accesses are for cache/memory for a given chip or for another chip on the same processor module or hardware node or distant hardware node.

When should I monitor with PowerVP?

You should proactively use your performance management tooling to understand your system(s) performance. It is best to have baseline information that reflects current performance levels. If you try to optimize performance later, you'll have a "before" baseline for your "after" improvement attempts. If you have a performance issue in the future, it is also good to have a "before/normal" baseline. Ideally you could run PowerVP all the time. Remember that you can only monitor (whether real-time or with play-back) information that was recorded with PowerVP as it cannot use historic data collected from other OS-based monitors.

Do I still need my OS-based performance tools?

PowerVP was created to compliment your current suite of performance tools in your toolbox. PowerVP focuses on new views that typically are not available with your OS-based performance tools. So, please use these tools together to monitor and optimize your system/application performance.

How should I set the color coded CPU utilization thresholds?

The utilization of a particular resource (core, disk drive, bus) simply indicates its level of being busy doing work. High or low doesn't necessarily imply good or bad.

If there is important work offered to your Power System, you would hope that it would be executed now; and processing that work will drive higher resource utilization. If there is spare resource utilization, you may wish that low-priority batch jobs will be able to take advantage of it; and processing that work will drive higher resource utilization.

If there is spare resource utilization, you may wish to be energy conscious and have some cores put to sleep; the result of this action will drive the utilization of the remaining resources to a higher utilization. The point here is that high utilization isn't necessarily a bad thing.

It is important to do sizing and capacity planning such that your system resources can handle the anticipated load as well as reasonable peaks in workload. In this planning, it is also important to plan some headroom (i.e., spare utilization). Part of the purpose of headroom, is to be able to do most of your work with CPU utilization level low enough not to have too much queuing time. The other purpose of headroom is to be able to handle workload peaks; perhaps during some of those peaks, you can deal with having additional response time due to the queuing multiplier effect. Best practices for headroom levels consider many factors (number of cores, resource type, partition type, partition size, etc). Please use the IBM Systems Workload Estimator to best size your new system, migration, or consolidation at www.ibm.com/systems/support/tools/estimator.

Therefore, you should customize these color-coded thresholds to meet the requirements of your business. You can set the number of thresholds, the utilization levels, and the colors. You might start with the default levels/colors and modify them to your customized environment. This customization will also help set your expectations and actions as to what to do when those levels are exceeded. For example if you see that your CPU utilization is generally red for an hour during your business day, do you: quickly make a change to increase the entitlement of a high priority LPAR? or consider a hardware upgrade to migrate to a newer/larger Power System in the near future? or consider activating some additional capacity on demand? or do you just know that you hit a red zone normally in the peak of your average day?

If I see red, do I have a performance problem?

Probably not, please reread the previous section. Red simply indicates a high utilization for a given resource.

How should I set the color coded link utilization thresholds?

The availability of the instrumentation for the Powerbus, MC bus, and GX bus is relatively recent. The default utilization thresholds/colors have been set as a starting point. We will continue to monitor these default thresholds and improve them using various workloads within IBM and from feedback from clients using PowerVP. High or imbalanced Powerbus utilization can be an indication that improvements can be made to affinity (see the affinity section).

How should I monitor my system with PowerVP?

It depends on the nature of your business and the health of your servers. You may wish to always record data and monitor it in real time, and then use the play back functions to do drill-down analysis as needed. To drill deeper, simply use the PowerVP

navigation (clicking and hovering) to look more closely at hardware nodes, bus utilizations, and partition details. Remember that you can only monitor (whether real-time or with play-back) information that was recorded with PowerVP as it cannot use historic data collected from other OS-based monitors. It is also possible to record PowerVP data without running the monitor GUI.

Some clients have talked about having the PowerVP system-level display projected on their wall or on the "big screen". They may customize PowerVP to alert them with particular colors for certain utilization levels.

How can I try optimization ideas?

You should get a good "before" monitoring interval prior to any changing of application or configuration. In doing so, note that day/time for the replay, or take screen shots, or note the CPU utilizations and CPI levels. For the LPAR drill-down panels, you can mark the bar charts with blue marks indicating the current levels. Then do the change to your application or configuration that you assume will provide an optimization. After this change settles down look again at the PowerVP data to confirm the performance level after the change. Typically one would want to keep the workload level equivalent to make appropriate comparisons. Now you can look for improvement indicators: CPU utilization reduction, CPI reduction, movement of the LSU CPI breakdown from right to left (remote memory to local memory, remote caches to local caches, L3 to L2, etc.), reduction of the utilizations of buses (Powerbus, MC bus, GX bus).

How can PowerVP help with DPO?

The Dynamic Platform Optimizer (DPO) can help optimize your virtualization configuration. Just like making any other change on your own, you can use PowerVP to help validate the performance benefit of DPO.

How do I map virtual partitions to physical configurations?

The new hypervisor interfaces created for PowerVP provide topology information. The illustrations on the PowerVP monitor show the specific existence and topology of cores, chips, processor modules, hardware nodes, and links. Each major view also has an LPAR section to show the virtual perspective by listing the partitions. PowerVP is able to help you map the virtual partitions to the physical configuration. By clicking on a dedicated partition, PowerVP highlights that partition in a unique color and also lights up the CPU resources (cores) in the same color. Then you can look at this mapping. Ideally, from the perspective of a given partition, the cores allocated would be grouped close together in the configuration to maximize the locality of data access. For shared processor partitions, PowerVP will map a given partition to a shared processor pool. Tasks in shared partitions can be dispatched to any core in the shared processor pool. If your server configuration has recently changed by adding partitions or changing the entitlement, you may want to consider running DPO to optimize the configuration for performance.

How can I tune for better affinity?

With IBM Power Systems, having good affinity is important for good performance. Servers that use a nodal design to increase their capacity (e.g., NUMA-like architectures) are especially sensitive to affinity considerations. Processor affinity suggests that your work be dispatched to hardware threads to the cores/chips/nodes with the highest likelihood of being close in proximity where your data is. Memory affinity suggests that memory allocated to your work be close in proximity to the cores processing your work. Ideally, your work would dispatched to the same core to optimize the chance of having a hot cache (vs. a dirty cache, or having to do remote cache accesses) or at least to the same socket or node to optimize the chance of having local memory accesses (vs. having to do remote/distant memory accesses).

Much of the CPU resource consumed for your application can be attributed to accessing data (that is, consuming cycles waiting for accesses from cache or memory). When accessing data, the goal is to consume as few cycles as possible. For POWER7, it is preferred to access data from this list in this order of preference: L1 cache, local L2 cache, local L3 cache, cache on another core on the same chip, cache on another chip on the same hardware node, cache on another hardware node, memory on your socket, memory on another socket on the same hardware node, memory on another hardware node. There might be a 1000 times difference in run cycles consumed from one extreme to another.

To improve your affinity you could try a number of things. Keep in mind that this may be an advanced technical topic. Application coding adjustments may maximize cache line optimization. Using pre-fetch or not may provide trade-offs between CPI, throughput, and response time. Using virtualization, such as dedicated partitions, may force better affinity. Using other OS-provided functions (RSET, subsystems, WPAR, affinity system values, etc.) may force better affinity. Many of these topics are discussed in papers at:
www.ibm.com/systems/power/software/aix/whitepapers/perf_faq.html or
www.ibm.com/systems/power/software/i/management/performance/resources.html
.

Can I impact the future direction of PowerVP?

IBM has responded to the performance management requirements for Power System clients and has delivered PowerVP to help meet those overall requirements. Remember, PowerVP is intended to be used along with the suite of all the existing performance tools. With this initial version of PowerVP, IBM welcomes feedback for this new monitoring tool to help guide future new enhancements. Please contact your IBM marketing representative to provide any feedback.