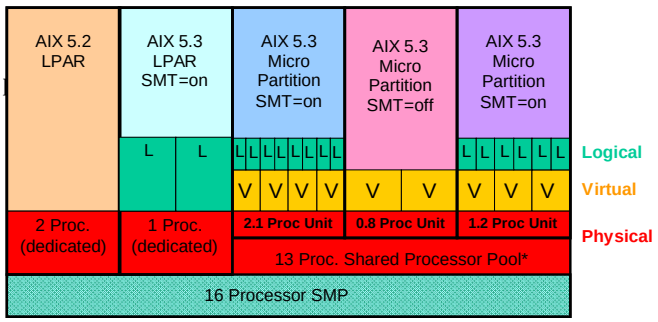
IBM System p5: Configuring Micro-Partition Processor Resources

Processor virtualization is a great way to get started with System p5 Advanced Power Virtualization (APV). APV processor virtualization can help you improve system utilization and reduce hardware/software expenses. This article provides an overview of the virtualized processor environment, LPAR configuration guidelines, plus tips on how to maximize processor utilization and minimize software licensing costs.

The Virtualized Processor Environment

System p5 servers can operate with a mix of dedicated processor LPARs and micro-partitions. The micro-partitions source their processing resources from a shared processor pool. The shared processor pool size is equal to the total number of active processors *less* the number of processors allocated to dedicated processor LPARs.



Think "PVL " P=Physical V=Virtual L=Logical (SMT)

The shared processor pool in this example has 13 processors (16 active - 3 dedicated = 11). The size of the shared pool will decrease/increase as dedicated processor LPARs are activated/deactivated.

Situated between the shared processor pool and the operating system we have processing units, virtual processors, and logical processors. You can think *PVL*, to remember the order of these items.

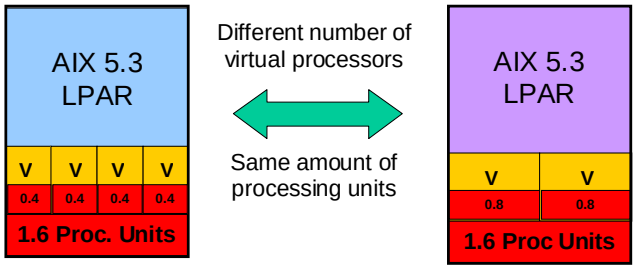
Processing units were introduced as a method to represent a portion of the shared processor pool to a micro-partition. Processing units can be configured in increments as small as 0.1. The processing units are presented to AIX through virtual processors. Virtual processors are always configured using whole numbers.

The presence of logical processors indicates that Simultaneous Multi-Threading (SMT) is enabled. SMT is the multi-threading feature of POWER5/5+ processors which allows AIX V5.3 to represent each micro-partition virtual processor or dedicated physical processor as two logical processors.

The Processing Unit & Virtual Processor Relationship

The number of virtual processors establishes the useable range of processing units accessible by an LPAR. Each virtual processor can represent between 0.1 and 1.0 processing units. For example, an LPAR with two virtual processors will be able to operate with a processing unit range of 0.2 to 2.0.

The system allocates processing units evenly across the virtual processors defined to an LPAR. For example, an allocation of 1.6 processing units to an LPAR with four virtual processors will result in 0.4 processing units being assigned to each virtual processor.



If the number of virtual processors is decreased to two, each virtual processor will receive 0.8 processing units. For a well threaded application, four virtual processors might be a better choice because four threads (eight with SMT=on) can be serviced at the same time. A single-threaded application would likely operate faster with the two virtual processor configuration because a single threaded application will run on just one virtual processor and would be receiving 0.8 processing units instead of 0.4

Article update: Starting with AIX 5.3 Technology Level 3, there is a new feature called processor folding. If you have virtual processors that are idle (fewer active threads/processes than virtual processors), the hypervisor will automatically notice this and redirect more CPU cycles to the active virtual processors. For more information on processor folding, see IBM redbook, Advanced Power Virtualization on IBM System p5, SG24-7940 at www.redbooks.ibm.com.

Configuring Micro-Partition Processor Resources

You designate an LPAR to run as a micro-partition by selecting the *Shared* option when the LPAR is created using the Hardware Management Console (HMC). Additional parameters that will need to be defined include:

- *Processing Units* (desired, minimum, maximum)
- *Capped/Uncapped*
- *Virtual Processors* (desired, minimum, maximum)

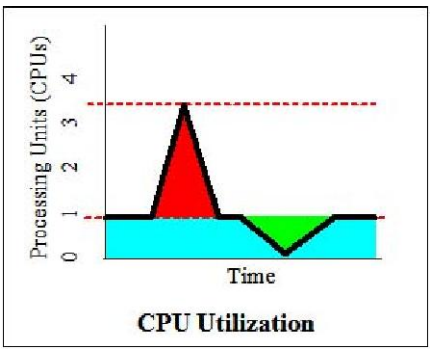
The *Desired Processing Units* parameter represents the guaranteed amount of processing resources that will always be available to a micro-partition. Micro-partitions operate in capped or uncapped mode. When a micro-parition is *capped*, the maximum amount of physical processor resources that it can access will be limited to the desired processing unit setting regardless of the number of virtual processors that have been configured (increasing the number of virtual processors simply decreases the processing units assigned to each one). Selecting *uncapped* mode allows an LPAR to access physical processor resources above its desired processor units setting. The maximum amount of uncapped processing units that can be accessed is gated by the desired number of virtual processors. Uncapped LPARs also have a priority setting which allows you to prioritize the allocation of extra processing cycles among all uncapped LPARs running in the shared processor pool.

The *Desired Virtual Processor* setting defines the number of virtual processors that will be created when the LPAR is activated. This setting establishes the maximum amount of processor resources that can be utilized by the LPAR. For example, an *uncapped* LPAR with two virtual processors can, at most, access two processing units. Recall that the desired processing units are distributed evenly across the virtual processors, hence the desired processing unit indirectly controls the amount of processing units assigned to each virtual processor.

The minimum/maximum settings (for both processing units and virtual processors), represent the extremes to which the desired values can be dynamically changed without stopping and then reactivating the LPAR. These dynamic changes can be made via the HMC GUI, the HMC command line interface, or by Partition Load Manager (PLM). In addition, the minimum processing units setting indicates the amount of processing units that must be available in the shared pool for the LPAR to start. Note that the maximum settings *do not* represent the amount of processing units that an uncapped LPAR can access. The maximum amount of uncapped processing units that an LPAR can access is limited by the desired virtual processor setting.

Guidelines for Setting Desired Parameters

Following is a diagram that shows the 24-hour processor utilization for an LPAR. Note that the processor utilization is represented in processing units and not as percentage.



This particular workload has a plateau of around 0.9 processing units, meaning on average, 0.9 processing units is adequate to run the application. Assuming the application has an average business priority level, we might choose to set the desired processing units to 0.9. This *guarantees* that the LPAR will always have access to 0.9 units (blue region). To satisfy the peak workload above 0.9 units (red region), this LPAR will need to be uncapped and will be a *user* of extra processing units during that portion of the day.

If this application has a high business priority, you might choose to set the desired processing units to 4.0. This will guarantee that processor demands (up to 4.0 processing units) will be met at all times with no dependence on the availability of extra processing units.

The sum total of desired processing units for all LPARs running in the shared pool can not exceed the physical number of processors allocated to the shared pool. To maximize the number of LPARs that can be running together, be careful not to overestimate the desired processing units setting.

Now we need to look at the desired virtual processor setting. Recall that the number of virtual processors establishes the maximum amount of uncapped processing units an LPAR can access. Our sample workload peaks at 3.5 processing units, so we need at least 4 virtual processors (3.5 rounded up to next whole number) to handle the peak workload. You can see that this application's processor needs can be satisfied with 4 virtual processors and a processing unit setting of 0.9 (uncapped) or 4.0 (capped).

All shared pool LPARs (capped and uncapped) can *donate* unused processing units to the shared pool for use by uncapped LPARs. Anytime the LPAR requires less than it's current allotment of processing units, the LPAR becomes a donor. In this example, with desired processing units set to 0.9, the green region designates donated processing units.

Optimizing "Desired" Settings

To optimize processor utilization for the shared processor pool, peak workloads (users) should be matched with idle workloads (donors) throughout the processing day. If the desired processing units are set too high for your collection of LPARs,

you will have too many donors and not enough users, resulting in less than optimal processor utilization. If the desired processing units are set too low for the collection of LPARs, you may have too many *users* and not enough donors to satisfy peak processing requirements.

Many software application licenses are based on the desired number of logical processors because they represent the maximum amount of processor resource the LPAR can access. To minimize software licensing costs, set the desired virtual processors parameter as low as possible, at the next whole number above the peak.

Key Learning points:

The *desired processing units* setting establishes a guaranteed amount of processing units that will be provided to an LPAR. The processing units will be evenly distributed across the virtual processors. LPARs can be *uncapped* to allow access to additional processing units.

The d*esired virtual processor* setting establishes the maximum amount of processing units that an uncapped LPAR can access, and is often the metric used for software licensing.

Additional information on this topic can be found in the *Advanced Power Virtualization on IBM System p5* Redbook (SG24-7940) available at www.redbooks.ibm.com.

Charlie Cler *has been with IBM for more than 21 years, working with the System p5 servers since their beginning 16 years ago. He currently supports customers in a system architect role. Charlie can be reached at cbcler@us.ibm.com.*

