

IBM Lotus Domino 6.5 with IBM Virtualization Engine technologies

• • • • • • • •

Richard Bassemir
IBM eServer Solution Enablement
September 2005



Table of contents

Abstract	3
Introduction	4
Scenario objectives	5
The scenario environment	5
Workload description	6
Running a capped Lotus Domino LPAR	7
Executing a capped LPAR scenario.....	7
Running an uncapped Lotus Domino LPAR	8
Executing an uncapped LPAR scenario.....	9
Changing to an uncapped Lotus Domino LPAR	11
Executing a capped-to-uncapped scenario.....	11
Changing memory allocation for a Lotus Domino LPAR	12
Removing memory: An enterprise scenario description.....	12
Enterprise scenario description: Adding memory.....	13
Running Lotus Domino with varying virtual processors	14
Summary	16
Resources	17
About the author	18
Trademarks and special notices	19

Abstract

This paper discusses how IBM® Lotus® Domino® is able to benefit from the IBM Virtualization Engine™ technologies offered with IBM eServer™ pSeries® systems, driven by IBM POWER5™ processors and running IBM AIX 5L™ Version 5.3. A model p510 system for a Lotus Domino mail server was used to test enterprise scenarios under a variety of conditions to observe the behavior of the server. Varying user workloads as well as the CPU resources showed not only that Lotus Domino behaves very well in this virtual environment, but also demonstrates how a company can exploit this technology to make more efficient use of its IT resources.

Introduction

The newest generation of pSeries servers using the IBM Power Architecture™ delivers outstanding innovation and performance to help businesses meet their on-demand requirements. The POWER5 processor represents the most advanced IBM 64-bit RISC chip with impressive performance when coupled with the AIX® 5L V5.3 operating system

One of the key capabilities introduced with this version of the operating system is the use of IBM Virtualization Engine technologies, which enable businesses to better control their system resources and increase the efficiency of those resources. The technologies give enterprises the ability to realize more value from their server investments.

There are numerous resources and Web pages containing information on the power of IBM Virtualization Engine technologies. This paper focuses on the use of the IBM Micro-Partitioning™ aspect of these technologies as it applies to Lotus Domino. References at the end of the paper will identify a few of the many additional resources that can be helpful. What follows is a basic explanation of Micro-Partitioning as it applies to the scenarios used in this white paper.

Micro-Partitioning allows the system's resources to be divided into pieces called processing units, which are smaller than a whole processor. For example, a single processor has 1.0 processing units. A logical partition can be defined with as little as 0.1 processing units (only one tenth of a physical CPU). With this ability, a logical partition (LPAR) can be given access to just enough CPU resources to do its job, while the rest of the CPU is free to be assigned to other needs.

User workloads change over time, and the beauty of Micro-Partitioning is that the LPAR can be dynamically changed without stopping or restarting workloads. If, during a peak workload, an LPAR is in need of more resources, they can be assigned. The system can even be set up to automatically use available resources without administrator intervention. This gives great flexibility to move CPU resources from LPARs with excess to LPARs in need.

Two terms used in this paper dealing with Micro-Partitioning are *capped* and *uncapped*. These terms refer to the capacity of an LPAR. If an LPAR is capped, then it cannot acquire any more resources than it has been assigned. If an LPAR is uncapped then it is allowed to take more of the available system resources when the need arises. The scenarios below will show the power of this ability in regard to running a Lotus Domino e-mail server.

Scenario objectives

The following scenarios were used to explore how companies might use the power of IBM Virtualization Engine technologies in a Lotus Domino environment to improve their resource utilization.

The scenario environment

To demonstrate the operation of Lotus Domino in an environment that takes advantage of virtualization, a model p510 two-way system was used. The system was set up with three logical partitions: LPAR1 and LPAR2 are the partitions that run single Lotus Domino 6.5 mail servers, and LPAR3 is the IBM Virtual I/O Server. Please note that LPAR2, the second Lotus Domino server, was not used in the analysis described in this paper.

When defining the partitions, the physical SCSI controller and Ethernet adapters are assigned to the Virtual I/O Server, because this server makes the Virtual SCSI and Ethernet devices available to the other logical partitions (LPAR1 and LPAR2).

This system has two physical processors (for a total of 2.0 processing units). The system has 8 gigabytes of memory. The following allocation was used when the LPARs were defined:

Partition	Name	Processing units	Memory	Physical adapters
LPAR1	dominoa	0.5	3.25	none
LPAR2	dominob	0.5	3.25	none
LPAR3	vioserver	0.5	1.0	SCSI controller, Ethernet

These initial assignments were dynamically varied during the testing (see Figure 1).

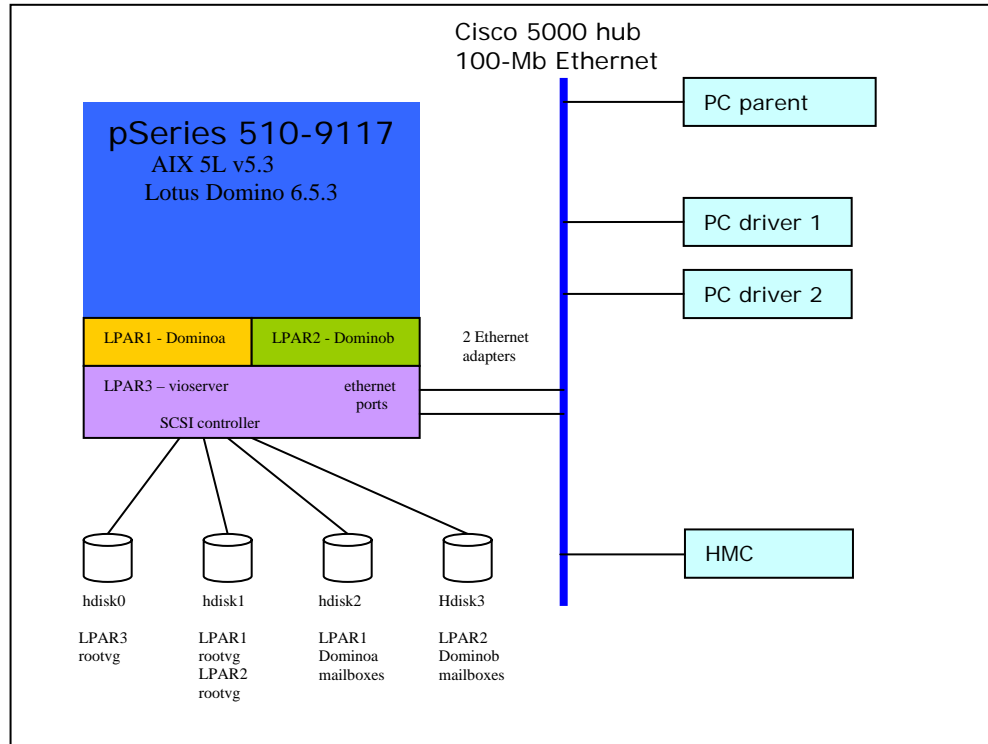


Figure 1: Lotus Domino Web Access simulation environment

Workload description

Each of the PC drivers was capable of simulating up to 1,000 Lotus Domino Web Access users. These users were started in staggered groups to avoid a sudden spike of user activity. Every 15 minutes each simulated user performed the following activities:

- Opened the mail database (documents averaged 10 kilobytes in size)
- Opened the current view
- Opened five documents (each might have up to 10-megabyte attachments)
- Categorized two of the documents
- Sent new mail memos and replies every 90 minutes with an average message and attachment size of 97 kilobytes
- Marked several documents for deletion
- Deleted documents marked for deletion
- Created one appointment (every 90 minutes)
- Scheduled one meeting invitation (every 90 minutes)
- Closed the view

Running a capped Lotus Domino LPAR

With Virtual Engine technologies, the enterprise has the ability to divide up the total number of available processing units into different LPARs to effectively use all the system's resources. For this scenario, the company is leasing a Lotus Domino LPAR and wants to limit the resources used by that LPAR to just the resources being specified in the lease.

Using the IBM Hardware Management Console (HMC), a capped LPAR is defined with a specified number of processing units that it cannot exceed.

Executing a capped LPAR scenario

The model p510 system is a two-processor system with 2.0 processing units to divide up between LPARs. The Virtual I/O Server LPAR, required to control all virtual resources, is given 0.2 processing units. Using the HMC, a capped Lotus Domino LPAR is defined with an entitlement of 0.5. The Lotus Domino server is being loaded with a workload of 500 Lotus Domino Web Access users to simulate the enterprise workload.

While the Lotus Domino server is running, and while the Web Access users are running the e-mail, workload resources are dynamically removed from the LPAR to determine the amount of resources needed to meet the user service-level agreement. For this enterprise, processors are run in the 80-90% utilization range. If the processors are pushed any higher, user response times are affected. If allowed to run any lower, valuable system resources are being wasted.

The graph below in Figure 2 shows the measured results using the `lparstat` command with a sampling every minute. The `lparstat` command is called from the AIX 5L v5.3 command line. The LPAR starts with an entitlement of .5 processing units. The first hour shows the Lotus Domino Web Access users ramp up to a total of 500 users. Then roughly every hour, 0.1 processing units are taken away from the LPAR. This continues until the LPAR is effectively using 100% of its entitled resources.

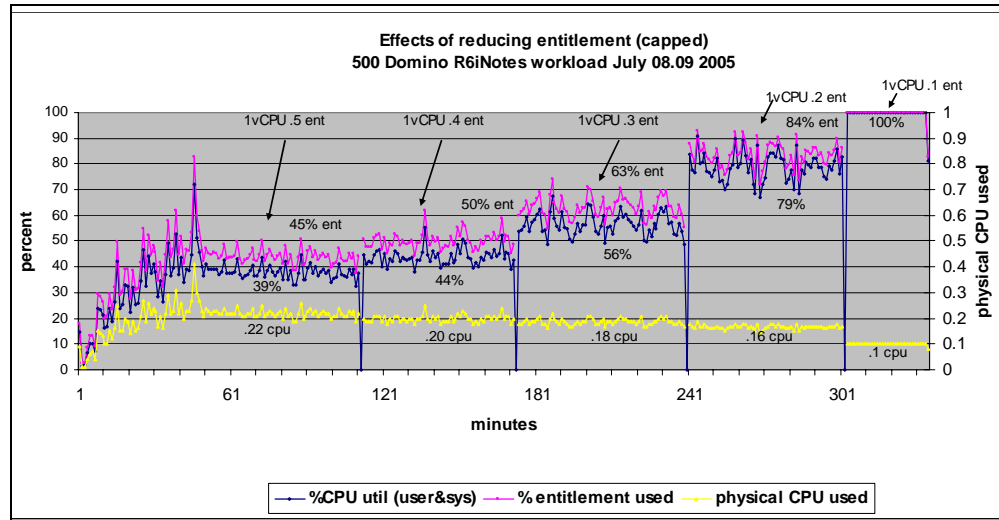


Figure 2: Dynamically reducing LPAR entitlement while there is a Lotus Domino workload

The chart shows how after the 500-user load was reached, the LPAR was less than 50% of its entitlement, indicating that resources are being unused. As resources are removed from the LPAR, the percentage of its used resources approaches its entitled resources. The result is more effective resource usage.

This particular workload shows that with an entitlement of 0.2 processing units, 84% of the LPAR allocated resources are being used. The LPAR is therefore using its entitled resources effectively, and the company has freed up 0.3 processing units for use elsewhere in the system. The LPAR can be redefined using the HMC to start with just 0.2 processing units so that the efficient use of resources will continue after any future restarts of the LPAR.

The yellow line in Figure 2 shows how much of the physical CPU resource in the system is being used. Notice that as entitlement is reduced, less physical CPU resources are being used. This is another indication how lowering the entitlement to make more effective use of the LPAR resources is freeing up physical CPU resources for use elsewhere in the system.

These changes in entitlement were done dynamically without interruption of the Lotus Domino Server workload. The data points at zero were placed there to illustrate when a change to the LPAR entitlement was made.

Running an uncapped Lotus Domino LPAR

In the prior scenario, the enterprise defined a specific level of system resources to assign to an LPAR, and that is an excellent way of controlling the use of system resources. In this next scenario, an enterprise that has an LPAR with a varying workload will be used. In this situation, an uncapped LPAR can automatically give the LPAR access to additional system resources to meet user demand.

IBM refers to this concept as *Capacity on Demand*. As the workload rises, eventually 80-90% of the CPU will be utilized, causing a greater demand for system resources. When the demand later falls, those resources are released.

By defining an LPAR as uncapped, the LPAR is able to obtain additional resources from an available pool of resources shared by multiple LPARs. The amount of resources placed in this pool is a function of the anticipated capacity required. This technique is a very powerful tool that allows a company to make efficient use of all the system's resources without dedicating resources to an LPAR that has only an occasional spike in workload.

Executing an uncapped LPAR scenario

This scenario begins the same as the previous example, with the model p510 system, a two-processor system with 2.0 processing units to divide up between LPARs. The Virtual I/O Server LPAR, required to control all virtual resources, is given 0.2 processing units. Using the HMC, an uncapped Lotus Domino LPAR is defined with an entitlement of 0.3, leaving 1.5 processing units to assign to other LPARs or leave in the shared resource pool.

A workload of 500 Lotus Domino Web Access users is applied to the Domino server to drive the LPAR to using 80% of its defined entitlement of 0.3 processing units. Once the system has stabilized, another 500 Lotus Domino Web Access users are applied for a total of 1,000 users. This addition simulates a spike in the LPAR workload. Because the 0.3 of a processing unit is not enough resource for the system, it automatically goes to the available pool of resources and applies that to the workload.

After running for four hours, 500 users are removed from the workload on the server, leaving the original 500-user workload and no longer requiring the resources that were taken on demand. Subsequently, those resources go back to the shared pool for use later or by another LPAR.

Figure 3 shows the behavior of the system during this test.

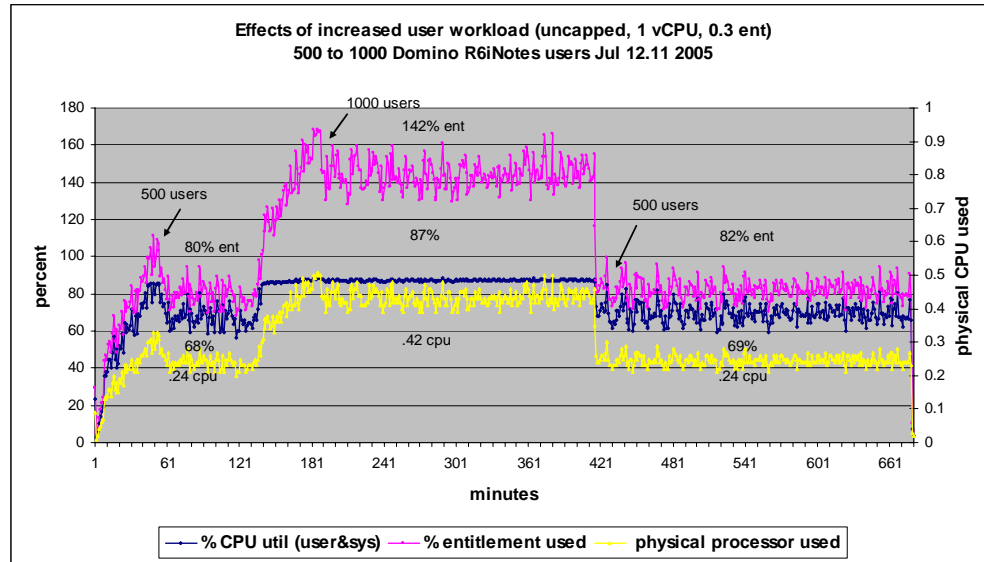


Figure 3: Varying workload on an uncapped Lotus Domino LPAR

The chart shows that, at the 1,000-user workload, the system was now using 142% of what it was entitled to use. Because the system was uncapped, it was allowed to take more processing units when required by the additional workload.

It is important to note that the CPU utilization reported by lparstat increased from 68% with the 500-user workload to 87% when the workload reached 1,000 users. Looking at just CPU utilization would give the false impression that the workload has doubled with less than a 20% increase in CPU utilization. Obviously, this is not true, because at 1,000 users, more resources were applied to the LPAR. Therefore, it is important to look at more than just the CPU utilization to understand how the LPAR is being used.

Note that the yellow line in Figure 3 shows that the increase in demand dictated how the LPAR used the system's physical processor. These increases are in 0.1 increments and done automatically by the system, and when the workload dropped, those physical resources were released by the LPAR.

What this scenario demonstrates is how a company can define an LPAR so that it can handle the peaks in workload without committing system resources that can only be used by that LPAR. Resources are taken from the shared pool and returned to the pool when no longer needed. Using this function, the hardware can be used much more effectively.

Changing to an uncapped Lotus Domino LPAR

In the next scenario, the enterprise starts out with a capped LPAR, but there is an unexpected change in the workload over time. As a result, the company finds the LPAR is consistently running at 80-90% utilization. The flexibility provided by IBM Virtualization Engine technologies provide the ability to dynamically react to these kinds of workload changes.

Executing a capped-to-uncapped scenario

Using the HMC, a capped Lotus Domino LPAR is defined with one virtual processor that has an entitlement of 0.3 processing units, with 1.5 processing units in the shared pool to use on demand.

A Lotus Domino Web Access workload is started and simulates 1,000 e-mail users on this LPAR. This workload is large enough to completely consume the LPAR's resources, and it can potentially impact the Web user experience.

Because the LPAR is capped, no additional resources are taken to meet the demand. However, given the flexibility of IBM Virtualization Engine technologies, the LPAR can be changed from capped to be uncapped.

Using the HMC, a dynamic change is made without interrupting the user workload, making this LPAR uncapped and allowing it to get access to resources in the shared processor pool. Then, after two hours, the workload is reduced to 500 users.

Figure 4 shows the behavior of the system during this test.

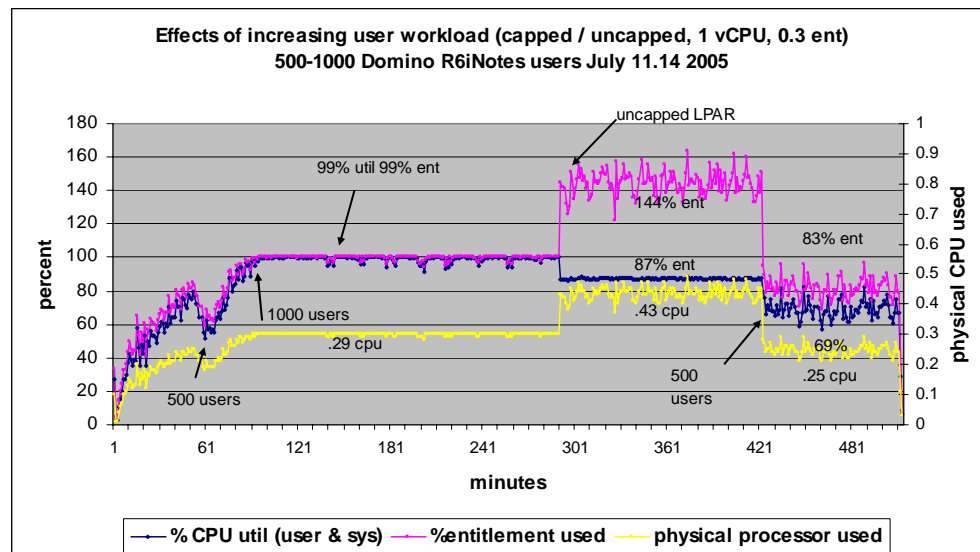


Figure 4: Dynamically changing a LPAR from capped to uncapped

The chart demonstrates that the 1,000 users saturate the LPAR. It is using all of its entitlement, and CPU utilization is at 99%. When the HMC is used to change the LPAR to uncapped, the system immediately applies additional processing units, as shown by the increase in measured utilization. At this point, the LPAR is using 144% of its entitled resources because it is uncapped.

Note that the CPU utilization drops to 87% and the physical CPU used increases to .43 processing units, all with no interruption to the user workload. It was not necessary to stop and restart either the Lotus Domino Server or the hardware.

When the workload drops down to 500 users, the resources applied to the LPAR are released back to the shared resource pool for use by other LPARs, and the physical CPU resources used drops to .25 processing units.

The company has been able to dynamically balance their system resources with no interruption to the user workload.

Changing memory allocation for a Lotus Domino LPAR

Processing units are not the only thing a user can control with IBM Virtualization Engine technologies. Using the HMC, it is possible to add and remove memory from an LPAR.

Removing memory: An enterprise scenario description

In this scenario, the enterprise has a Lotus Domino LPAR with one virtual processor that has an uncapped entitlement of 0.3 processing units and 3.25 gigabytes of memory. Because of other business requirements, a new LPAR is needed, and that LPAR has a requirement for more memory than is currently available on the system. The company needs 1 gigabyte of memory.

Using the HMC, a capped Lotus Domino LPAR is defined with one virtual processor that has an entitlement of 0.3. It has 3.25 gigabytes of memory, and the model p510 system being used has a total of 8 gigabytes. The Virtual I/O Server in its own LPAR is using .5 gigabytes of memory, leaving over 4 gigabytes of available memory.

A Lotus Domino Web Access user workload of 500 users is started, and after letting the system run for a period of time, the HMC is used to dynamically take 1 gigabyte of memory away from the LPAR. Then, after a couple of hours, the HMC is used to assign that memory back to the LPAR.

Figure 5 shows the behavior of the system during this scenario.

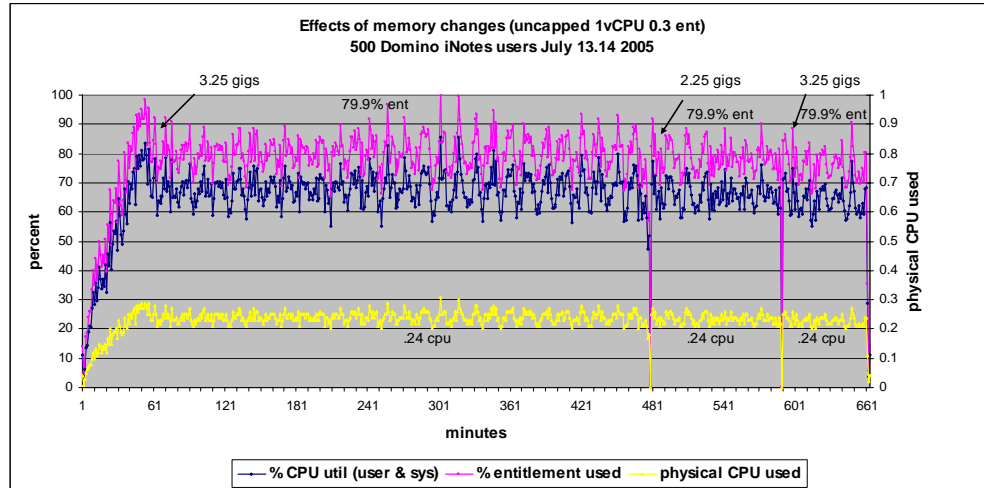


Figure 5: Reducing memory on a Lotus Domino LPAR dynamically

The chart shows how removing 1 gigabyte of memory had relatively no impact on the user workload, and the Lotus Domino server did not have to stop or restart. Likewise, adding the 1 gigabyte of memory back into the system was accomplished without any interruption.

This shows how Lotus Domino is able to handle the dynamic changes to memory allocation for this workload and system configuration. In this case, removing 1 gigabyte of memory still allowed Lotus Domino to service the 500-user workload. Obviously there are limits to how much memory Lotus Domino requires to run, and the company is best advised not to attempt to drop below the level on which its environment depends. Removing too much memory can cause the Lotus Domino application to stop running. To prevent the accidental removal of too much memory, the LPAR is defined with a minimum requirement for memory. Attempts to remove memory below that level are not allowed by the HMC.

Note: The data points showing utilization at zero are there only to indicate when the change to the environment was made. They do not indicate any outage in service.

Enterprise scenario description: Adding memory

As in the last scenario, the enterprise has a Lotus Domino LPAR with one virtual processor with an uncapped entitlement of 0.3 processing units and 3.25 gigabytes of memory. The model p510 system being used has a total of 8 gigabytes of memory, and the Virtual I/O Server in its own LPAR is using .5 gigabytes of memory, leaving over four gigabytes available.

Although the system is running fine, utilization is reaching a critical business limit and there is a desire to apply more memory to the Lotus Domino LPAR to reduce utilization.

A Lotus Domino Web Access user workload of 500 users is started, and after letting the system run for a period of time, the HMC is used to dynamically add 2 gigabytes of memory to the Lotus Domino LPAR.

The 2 gigabytes were then removed from the system to observe the ability of Lotus Domino to handle these dynamic changes. Figure 6 shows the behavior of the system during this scenario. Throughout the test the amount of physical CPU used was constant.

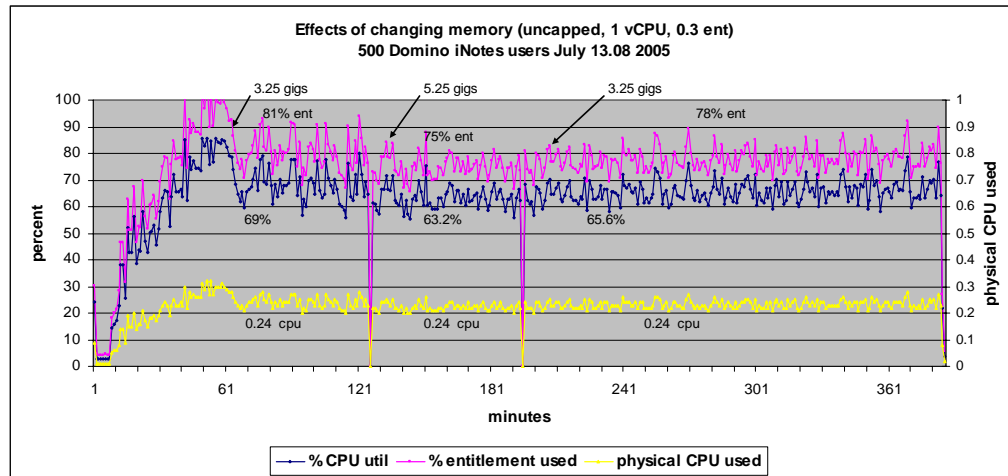


Figure 6: Adding memory on a Lotus Domino LPAR dynamically

The chart shows how upon reaching the 500-user workload the LPAR was using 81% of its entitlement, and the CPU was running at 69% utilization. After adding 2 gigabytes of memory, the percentage of entitlement and the CPU utilization dropped slightly. The lower CPU utilization indicates that the resources are being used less efficiently.

Why would a company want to do this? The system might have plenty of available memory, but not enough processing power. By adding more memory to the Lotus Domino LPAR, less processing units are required. Those processing units could then be applied to another LPAR in need of CPU resources. This is the flexibility that comes with IBM Virtualization Engine technologies.

Note: The data points showing utilization at zero are there only to indicate when the change to the environment was made. They do not indicate any outage in service.

Running Lotus Domino with varying virtual processors

In this scenario the enterprise has a Lotus Domino LPAR with one virtual processor with an uncapped entitlement of 0.5 processing units and 3.25 gigabytes of memory. The model p510 system being used has a total of 8 gigabytes of memory, and the

Virtual I/O Server in its own LPAR is using 0.2 processing units and 0.5 gigabytes of memory, leaving 1.3 processing units in the shared pool and over 4 gigabytes of available memory. This scenario will determine if it is better to run more virtual processors with smaller entitlement or fewer virtual processors with larger entitlement.

A Lotus Domino Web Access user load is started, and once the 1,000-user load is reached, the HMC is used to increase entitlement to something that would put the CPU in the 50% utilization range.

Virtual processors are then dynamically added using the HMC. The LPAR starts with one virtual processor, then, over time, is changed to two, four, and finally eight. Then, after an hour, the order is reversed to remove each of them.

Figure 7 shows the LPAR behavior during this scenario.

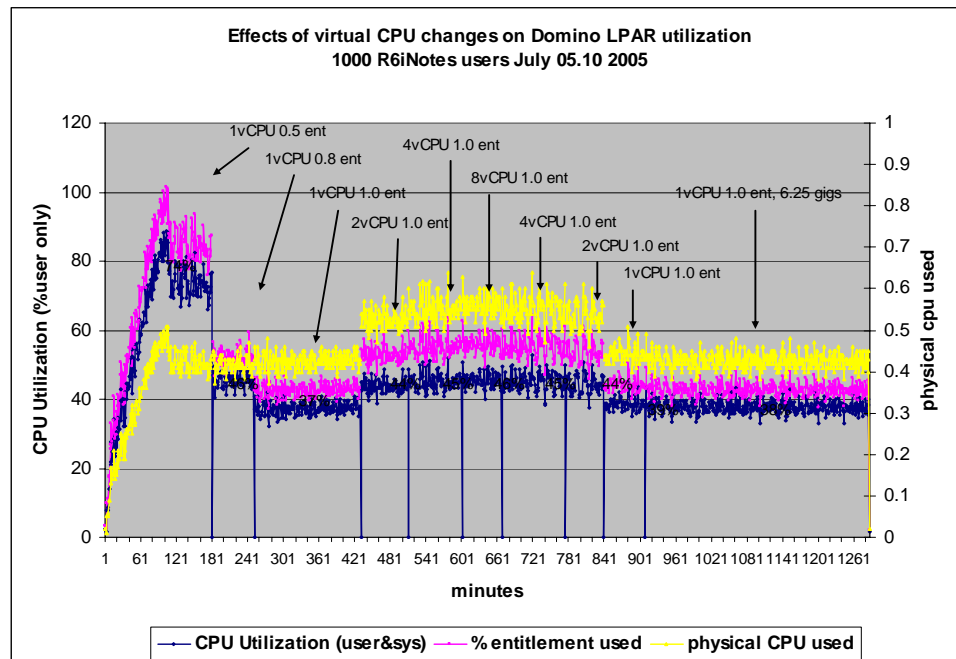


Figure 7: Changing the number of virtual processors on a Lotus Domino LPAR

The chart shows that adding virtual processors for this Lotus Domino Web Access workload did not result in any significant improvements. Instead, adding virtual processors actually resulted in the more physical CPU resources being used, which is not the best use of available resources.

The data shows that for this environment and workload, adding virtual processors increased both the CPU utilization and the percentage entitlement used. Therefore, it is better to use fewer virtual processors. The LPAR generally performs best if the number of virtual processors is close to the number of processing units available for that LPAR.

For this particular workload, the most effective use of the available resources would be a single virtual processor with 0.5 entitlement. Adding additional processing units is wasting valuable processing power.

IBM Virtualization Engine technologies allow the company to have the flexibility to dynamically tune their configuration and workload for the most efficient use of available resources

Summary

Partitioning provides the enterprise with greater flexibility when deploying multiple independent workloads on servers. With Micro-Partitioning, the company can manage the server for better availability and more efficient use of resources.

This paper has focused on a subset of those benefits by showing how Micro-Partitioning and memory allocation can be applied to a partition running Lotus Domino Web Access services. It also demonstrates that the Lotus Domino server is well suited for this technology. Resources were quickly added and removed from the Lotus Domino LPAR without disrupting the Lotus Domino Web Access user experience.

Businesses no longer have to size LPARs for peak workloads. Instead, the LPARs running Lotus Domino can be sized to meet a base level of workload demand and then make use of Capacity on Demand by using CPU resources in the shared pool.

With the dynamics in today's business environment, it is vitally important for companies to be flexible and responsive to changes in their IT environment. IBM Virtualization Engine technologies give enterprises the ability to be a business that responds on demand to changing business dynamics, and Lotus Domino adheres well to that environment.

Resources

These Web sites provide useful reference materials to supplement the information contained within this document.

- IBM UNIX® servers: eServer p5 and pSeries servers
ibm.com/eserver/pseries
- AIX 5L for POWER Version 5.3
ibm.com/servers/aix/os/53desc.html
- IBM Virtualization Engine technologies documentation: Introduction to Advanced POWER Virtualization on IBM eServer p5 servers
www.redbooks.ibm.com/abstracts/sg247039.html?Open
- Partitioning documentation: Partitioning Implementations for IBM eServer p5 servers
www.redbooks.ibm.com/abstracts/sg247039.html?Open

About the author

Richard Bassemir has worked at IBM for 27 years. During this time he has worked in mainframe logic design, storage controller simulation, manager of software tool system test, LAN systems certification testing and more recently in Solutions Enablement. For the past two years Rich has worked closely with IBM Lotus development to assure concurrency and availability of Lotus Domino on pSeries hardware. This has included testing and performance analysis to assure that Lotus Domino on the pSeries system is a world class solution.

Trademarks and special notices

© IBM Corporation 1994-2005. All rights reserved.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both: IBM, the IBM logo, eServer, pSeries, Lotus, Domino, Micro-Partitioning, POWER, POWER5, AIX, AIX 5L, Virtualization Engine, Redbooks, Power Architecture, and ibm.com.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.