Parallels® Virtuozzo Containers

White Paper

Parallels Virtuozzo Containers for Windows Capacity and Scaling

www.parallels.com

Version 1.0

Table of Contents

Introduction 3
Resources and bottlenecks 3
Comparison of 32 bit and 64 bit Windows 4
-Comparison of 32 bit and 64 bit Windows 4
-Running 32 bit applications in 64 bit Virtual Environment 4
Resources monitoring 5
-Performance monitor 5
-CPU 5
-Physical Memory 6
-Application memory 6
-Server memory 6
-DIsk space 7
-DIsk I/O 7
-Network bandwidth 7
-Memory pools 8
Server consolidation with Parallels Virtuozzo Containers for Windows 10
-OS virtualization versus VM virtualization 10
-Choosing hardware for the server consolidation 11
References 11

Introduction

Parallels Virtuozzo Containers for Windows virtualization solution, among other goals, helps IT professionals to increase server utilization.

However, high resource utilization may be also associated with resource over-commitment, creating performance and even stability issues. This document should help Virtuozzo users to properly plan, setup and monitor their virtual infrastructure, providing efficient resource usage while meeting applications performance expectations and maintaining high services availability.

Resources and bottlenecks

Virtualization technologies enable efficient resource utilization by isolating and enabling multiple execution environments to use the same resources. Therefore, a server with multiple virtual environments runs more services, applications and serves more users at the same time than a typical standalone server. Some of the exhaustible resources on a server include:

- CPU
- Physical memory
- Disk space
- Disk IO
- Network IO
- System Kernel resources

Higher performance usually means higher hardware cost; so the resources are best utilized in case no resource becomes a bottleneck thus constraining utilization of other resources.

Some of these resources, such as system kernel resources, are platform specific. Proper planning and monitoring can be different depending on the platform chosen.

Parallels Virtuozzo Containers for Windows: 32 and 64 bit versions

Major vendors including AMD and Intel offer a broad range of x64 processors, which gives Virtuozzo users a choice between 32 and 64 bit versions of the software. Due to architectural differences of the operating system, these two may have different characteristics, making one version more suitable for a particular deployment. The information below should help you to make a proper choice between two platforms, as well as set proper expectations on the density and scalability of your virtualized environment.

COMPARISON OF 32 BIT AND 64 BIT WINDOWS

The table below highlights the difference between the memory limits of 32-bit and 64-bit versions of Windows 2003 server

General Memory Limits	32-Bit	64-Bit
Total virtual address space (for a single process)	4 GB	16 TB
Virtual address space per 32-bit process	2 GB (3 GB if system is booted with /3GB switch)	4 GB if compiled with /LARGEADDRESSAWARE (2 GB otherwise)
Virtual address space per 64-bit process	N/A	8 TB
Paged pool	470 MB	128 GB
Non-paged pool	256 MB	128 GB
System Page Table Entry (PTE)	332 MB (780 MB if LargeSystemCache flag not is set)	128 GB

Physical Memory and CPU Limits[1]	32-Bit	64-Bit
Windows Server 2003 Standard Edition	4 GB / 4 CPUs	32 GB / 4 CPUs
Windows Server 2003 Enterprise Edition	64 GB / 8 CPUs	1 TB / 8 CPUs
Windows Server 2003 Datacenter Edition	64 GB / 32 CPUs	1 TB / 64 CPUs

Table 1. Comparison of 32-Bit and 64-Bit Windows 2003

The scalability limitations applicable to the operating system are also applicable to virtualized environments. The major differences between these versions affect Virtual Environment density and scalability, including the amount of physical memory supported, size of virtual address space; size of paged, non-paged pools and System PTE size.

RUNNING 32 BIT APPLICATIONS IN 64 BIT VIRTUAL ENVIRONMENT

The support for running 32 bit Windows applications on 64 bit Windows is provided by the Windows on Windows 64 (WOW64) emulator.

The 32-bit applications running on Windows Server 2003 x64 Editions use the WOW64 emulator that allows the application to run exactly as if it were running on a 32-bit Windows Server 2003. While the emulation imposes a slight translation overhead, the effect is minimal, and many 32-bit applications will actually run faster on Windows Server 2003 x64 Editions because of other efficiencies such as faster file-system access.

[1] The data is applicable to R2 and SP1 versions; other versions may have different limits. See http://msdn2.microsoft.com/en-us/library/aa366778.aspx for more details.

There are two major limitations in this emulation:

- Kernel mode drivers must be written for x64 Windows. There is no support for 32-bit drivers, and applications that install or depend on specific 32-bit drivers will not run on x64 editions of Windows Server 2003. An example of such application may be Microsoft® Exchange Server 2003.
- Windows Server 2003 x64 Editions do not support DOS or 16-bit applications.

As a result, the majority of existing 32 bit applications built for 32 bit version of Windows will work on x64 Windows without any functionality or performance impact. Certainly, a virtualized 64bit environment can run 32 bit Windows applications as well as Windows itself can.

Resources Monitoring

As we discussed above, avoiding bottlenecks is important when we want to maximize the use of the hardware. This chapter discusses how to monitor the usage of different resources and how to predict the maximum number of users per server depending on the results of your monitoring.

PERFORMANCE MONITOR

Most of the monitoring values described are collected using Windows Performance Monitor, perfmon.exe. When Virtuozzo is installed, a new performance object called "Virtuozzo" becomes available. Many examples below refer to this performance object.

For more information on using the Windows Performance Monitor, please refer to Windows server 2003 documentation.

CPU

CPU performance is generally easy to monitor. We recommend that the average total CPU consumption on a server should not exceed 60-80%, with occasional peaks up to 100%.

The total server's CPU consumption is a sum of the CPU consumptions of all the VEs and the host system. Note however that the CPU consumption numbers from inside of a VE differ from real numbers. Say, 100% CPU usage in a VE means that VE is using 100% of its available CPU cycles (which may be only 20% from total server CPU power).

The following perfmon counters may be used in CPU consumption monitoring (all the measures here and below are taken from the physical server running Virtuozzo unless otherwise stated):

- Total CPU consumption on a server: **\Processor(_Total)\%Processor time**
- Total CPU consumption by all VEs: **\Virtuozzo(_Total)\VPS - %Processor Time**
- Total CPU consumption by a particular VE: **\Virtuozzo(VPS VEID)\VPS - %Processor Time**

By default, a number of Virtual CPUs in a VE is equal to the number of system CPUs (sockets, cores, hyperthreads). Note that for certain usage scenarios this configuration may be not optimal. Applications using CPU resources via a single thread generally perform better in a single CPU Virtual Environment. Also limiting the number of CPUs each VE can access can slightly increase overall system performance, because the system has less overhead for scheduling multiple tasks on multiple CPUs

Additionally, limiting the number of Virtual CPU for a VE efficiently limit the ability of the VE to consume server's CPU power. For example, one out of 4 VEs with 2 VCPUs each may do not experience any CPU performance issues on 8CPU physical box, even when other 3 use 100% of the CPU power available to them because those 3 VEs would be only able to occupy 6 physical CPUs in total.

You may limit the number of Virtual CPUs in a VE via Virtuozzo GUI tools or using command line (executing a command **vzctl set {VEID} --cpus *{number of VCPUs}* --save**).[2]

PHYSICAL MEMORY

Physical memory is probably the most important resource to use to plan your consolidation, it is the resource that commonly can be the bottleneck. There are two major considerations you should be aware of when planning your server consolidation. First, the amount of memory used by your applications that you are going to consolidate. This would give you an idea about possible consolidation ratio.

Another consideration is the server's memory. In addition to monitoring free available memory, we also need a reliable indicator which would tell us when the memory becomes a bottleneck.

Application memory

When planning the consolidation, you need to figure out how much memory applications are currently using. The performance monitor may help us here: the **\Process(*Process Name*)\Working Set** counter shows how much memory the process is currently using. **\Process(*Process Name*)\Working Set Peak** may be even more helpful when you want to be conservative and plan for usage peaks. The unit of the measurement is byte – so don't forget to divide by 1,048,576 since you need megabytes.

There are many different memory related counters. "Working Set" shows the perfect parameters, since it estimates the memory usage from a "RAM" perspective, eliminating page file and allocated but not used memory.

If you plan to allocate multiple applications you can add their working sets. The same pages are often used by multiple applications so adding their requirements will provide a safe or conservative estimation of usage.

Server memory

There are two important questions to consider when monitoring the memory usage on a Virtuozzo server:

- How much memory is still available? How many more VEs I can run on my server?
- In the current situation, is the memory a bottleneck?

The following performance counters may help you to answer these questions:

- Free memory available on a server: **\Memory\Available MBytes**
- Paging activity: **\Memory\Pages/sec**
- Paging file usage: **\Paging File\% Usage**

First counter shows how much memory is currently available on a server. Note however that this counter is not very straightforward. For example, with 600MB, you could potentially add 4 more applications requiring 180 MB each. Modern server Operating Systems have complex memory management algorithms, often using memory for caches when it is available and freeing it when applications demand it.

However, if the counter is typically rather low (often hitting the 4MB threshold, which immediately invokes memory freeing mechanisms) – it indicates that the system is on low memory almost permanently, likely producing high paging activity.

Additionally, two other counters may help you to check whether there is an issue with paging activity. The **\Memory\Pages/sec** counter shows how often the system needs to read something from disk in order to let process to use its memory, and the **\Paging File\% Usage** counter indicates how much of the paging file is in use. Occasional peaks of **\Memory\Pages/sec** are fine, while permanent activity

[2] Note that this functionality is available in Parallels Virtuozzo Containers version 4.0

over 20 pages/sec is a sign of the problem. Note that if you see 20 pages/sec because of 500 pages/sec peak that lasted several second – there is nothing to worry about. Only permanent paging activity should concern you.

As for the paging file usage, it is generally safe to keep it under 70%.

DISK SPACE

During the consolidation process, you consolidate the original server's hard drives as virtual disk images on a Virtuozzo server.

However, the server's disk space requirements depend on the real disk space used on the servers to be consolidated, rather than the disks size itself.

In Virtuozzo, the "compact" type Virtual disk type (used by default) has a size which is close to the real disk space usage, not to its limit. That means that if disk size (which is essentially amount of the disk space available for a VE) is 30GB, while the real usage is 10GB, the virtual disk image file size on the Virtuozzo server will be close to 10GB, not 30. When the usage increases, the file grows – automatically shrinking when the VE frees its disk space.[3]

Another disk savings comes from operating system and application templates. The Windows Server operating system and any application installed as a template will be shared, adding almost nothing to each VE. For Windows Server 2003, this means at least an extra gigabyte of the disk space saved.

Considering the mentioned above, in order to provide 30GB of the disk space for 20VEs, you don't have to pre-allocate 600GB of the disk space. Instead, you should rather calculate how much the disk space is currently used in those VEs. Let's say it is 10GB in average. It requires about 9GB of server's disk space per VE, since the templates are in use. Taking into account possible increase of the disk space in the future (let's say we assume further 100% growth), the total disk space requirements per server become (9GB + 9GB*100%) * 20 = 360GB

The following performance counters may be used for monitoring the disk space usage on a server and in a Virtual Environment:

- Free space on the server available for all VEs: **\LogicalDisk(D:)\% Free Space**
- Free space available in a VE: **\Virtuozzo(VPS (VEID)\VPS - Disk Space Free**

DISK I/O

Disk I/O is one of the vaguest parameters for monitoring. Megabytes per second can be meaningless. If you measure bandwidth in a linear reading it could be an order of magnitude different from a random sector bandwidth reading, and neither reading reflects real usage patterns well.

The best parameter to evaluate if the disk a bottleneck is a disk queue length – **\Physical Disk (Disk ID)\Avg**. **Disk Queue Length**, which indicates number of read/write requests queued to the disk. When the number exceeds the number of spindles * 2, that indicates that new requests are typically not served immediately, they wait.

Note that for RAID arrays the number of spindles for read and write operations may be different, so you may need to consider read and write queue length separately.

NETWORK BANDWIDTH

Generally, network bandwidth saturation is not an issue in a typical virtualized infrastructure (or, strictly saying, is not a virtualization specific issue). Typically Ethernet is already shared between multiple servers,

[3] With default settings, disk compression is performed automatically every 30 minutes when the free space exceeds 30% of current disk size.

and server consolidation on a single physical server does not affect the picture too much. However if performance of a single network interface becomes a bottleneck, you may consider the following options:

- Use a higher performance network interface (e.g. Gigabit Ethernet)
- Use network adapter teaming in a mode that increase the bandwidth
- Use a VE network adapter in the Bridged networking mode (instead of routed mode), which is more efficient for high throughput rate.

The following performance counters may be used for monitoring network bandwidth:

- Total traffic on physical adapter: **\Network Interface(*Name of the adapter*)\Bytes Total/sec**
- Total traffic of all VEs: **\Network Interface(SWsoft Virtual Network Adapter)\Bytes Total/sec**
- Total traffic of a particular VE: (*the counter is not available from outside a VE, in-VE counter should be used*)

MEMORY POOLS

Under certain usage scenarios the memory pools (as well as System Page Table Entry, or PTEs) may become a major density constraint in a 32 bit environment. Let's discuss in detail how memory pools are used and how a shortage of the memory pools may be diagnosed.

The Windows memory manager creates the following memory pools that the system uses to allocate memory: **nonpaged pool** and **paged pool**. Both memory pools are located in the system region of the address and mapped into the virtual address space of each process. The nonpaged pool consists of virtual memory addresses that are guaranteed to reside in physical memory. The paged pool consists of virtual memory that can be paged in and out of the system.

The handles for kernel objects, such as processes, pipes, access tokens and others are stored in the paged pool.

The nonpaged pool stores the system records, the limits and current values for its nonpaged pool, paged pool, and page file usage. For more information, see Memory Performance Information.

The System Page Table Entries area is used to map system pages such as I/O space, Kernel stacks, (which is 3 pages or 12KB per thread, and 12 pages or 48KB per GUI thread on x86 architecture) and memory descriptor lists.

The parameters for monitoring via the performance monitor are:

- Non-paged pool usage: **Memory\Pool Nonpaged Bytes**
- Paged pool usage: **Memory\Pool Paged Bytes**
- Free PTE entries: **Memory\Free System Page Table Entries**

Generally, up to 180 MB are safe thresholds for the nonpaged pool and 360 MB for the paged pool. Safe values for the Free Page Table Entries values are 24,000 or more. If the operating system runs out of space in one of those areas, programs may begin to encounter unexpected errors.

The more VEs you run on server, the more memory from these pools are in use. The heavier the applications are – again, the more memory is used from these pools.

It's difficult to predict how much of those resources would be used by different applications; however common rules are:

- Applications that work partially in the kernel space (using kernel drivers) tend to use more resources from these pools than their pure user-space counterparts.
- A few big applications will likely use fewer kernel resources than many little ones.
- Applications opening an enormous number of file handlers use a lot of kernel resources.

Here is an example. Two diagrams below display resources usage by the number of VEs running on a server. Small VEs with default set of services, one terminal server connection and a word processor opened were used in these measurements.

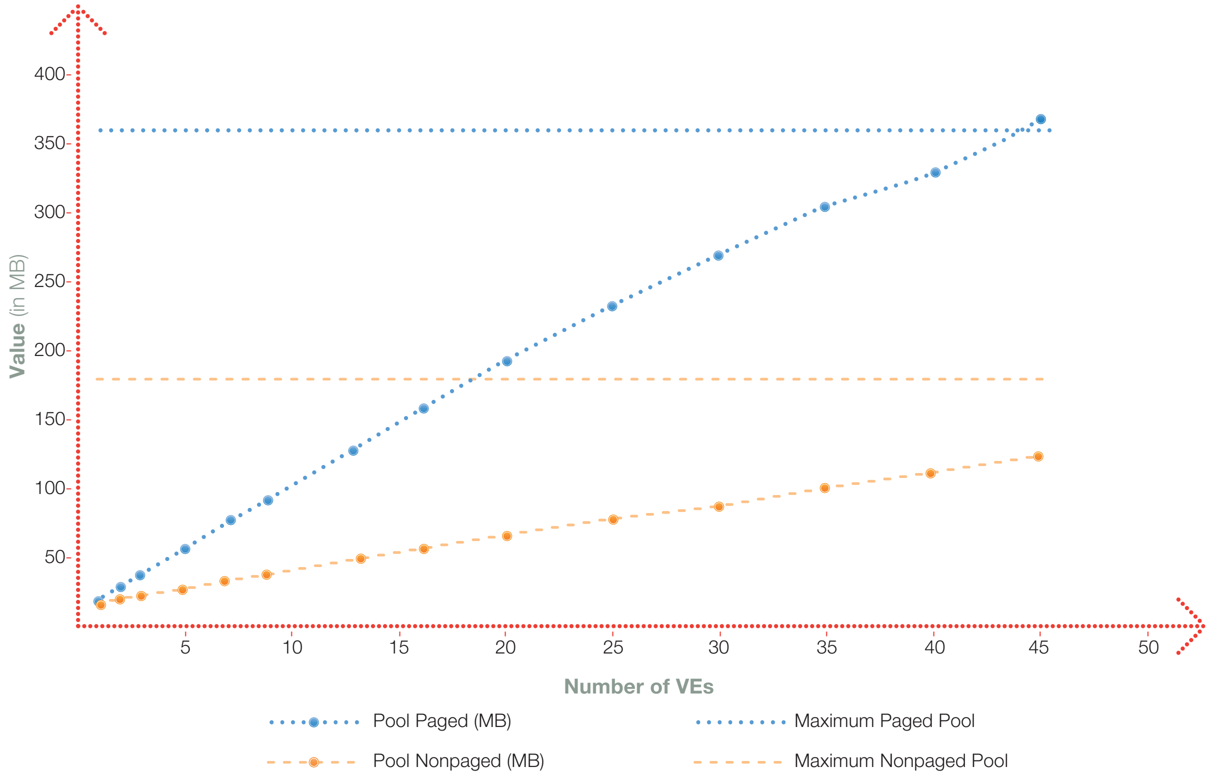

Figure 1. Paged/Nonpaged Pools usage by the number of VEs

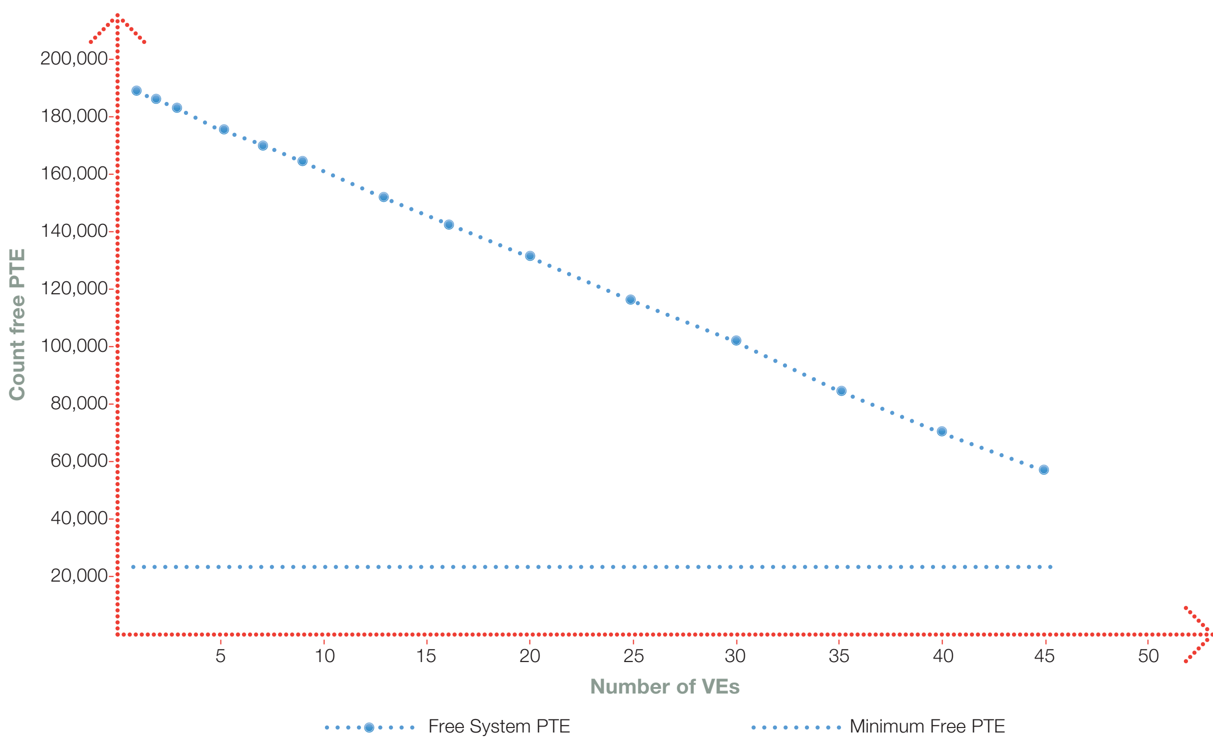

Figure 2. Free System PTE counter by the number of VEs

As you can see, for this particular type of use about 40 VEs per server would be a practical limit for the 32 bit architecture due to the Paged Pool exhausting, regardless of the amount of physical memory and other resources available.

Many applications typical for enterprise server consolidations demand more of these resources, so practical density limit for a 32bit platform may become 10-20 instances per server.

For 64 bit edition of Windows, there are no practical constrains for these resources.

Server consolidation with Parallels Virtuozzo Containers for Windows

OS VIRTUALIZATION VERSUS VM VIRTUALIZATION

In the operating system virtualization approach, you consolidate applications and operating systems instead of consolidating hardware. Let's discuss how the different architecture affects the consolidation ratio and what are the best practices for choosing hardware for server consolidation.

Imagine the following consolidation scenario.

You have a thousand hypothetical physical servers, with 2GB of physical memory each, running Windows server family operating system. The servers are dual 1GHz CPU, with average CPU utilization 10%. A hypothetical application is using 300MB of memory out of 2GB available; the rest is used by the operating system.

These data reflect results of technology research companies, claiming that in non-virtualized server environments average resources utilization is 10-15 percents.

As reference hardware for our virtualization platform, we have chosen quad core dual 3GHz CPU servers, with 8GB of memory available.

Let's calculate the consolidation ratio we can expect with VM solution.

First, we need to define the amount of memory we want to dedicate for the virtual guests. As we know, the application footprint is 300MB. The operating system needs to have some memory for the kernel, also efficient file system access requires enough memory for the file system cache – so it is unlikely we could strip it down to less than 768MB. Later the memory paging activity in a VM will show if the decision was wise – let's hope that it was.

Each VM requires 768MB of RAM. Considering that we have 8GB of physical memory, we reserve 7GB for use of by VMs. Even assuming that we can have 30% of memory over commitment, this would give us (7GB + 7GB*30%)/768MB – i.e. about 12 VMs per server. 8 cores of CPU give us an amazing 24GHz of computing power, but here we can't fully benefit from it – we are memory bound. Let's hope that increased CPU power would help our application performance, instead of serving excessive paging activity (remember, we decreased VM memory during the consolidation)

Adding more memory, we can do more. 16GB would give us numbers close to 24:1

Now let's try to figure out the density numbers with Virtuozzo.

We should take some part of those 8 gigabytes of memory for the operating system – it serves all the applications, so let's give it a lavish 2GB. It leaves 6GB to the applications. Each application will be running in separate virtual environment, which has its own memory footprint. Measurements show that it may be anywhere from 30 to 50 megabytes, so let's assume it 40.

So the total footprint is 340 MB. However, we didn't consider the recourses sharing yet. A part of this footprint is coming from the shared libraries (dll), or application own images, and will be shared among multiple VEs on the same server, eventually increasing the application's memory footprint. Another part is rarely used and may be safely stored in the swap file. Very conservative sharing assumptions will bring the footprint back to the original 300MB (in real life it would be closer to 250MB, but let's stay on a safe side, we don't want our applications to suffer).

Now let's calculate the consolidation ratio. 6GB/300MB give us 20 VM per server, with again, plenty CPU power per application.

With more memory, we increase the consolidation ratio. 16GB of RAM, with 3GB given to the operating system leaves 13GB for our applications, or about 43 VEs per server – and still plenty of CPU power for good applications performance. However make sure you reviewed the considerations about kernel memory above. 32bit windows may be running out of the system resources – that's where 64bit version comes to the rescue.

CHOOSING HARDWARE FOR THE SERVER CONSOLIDATION

The 64bit version of Virtuozzo may provide almost unlimited scalability, and run at least several hundred Virtual Environments on a server. However, the hardware price may be just not justified.

First, quad socket servers from leading hardware vendors may be almost twice as expensive as their dual socket counterparts, and the CPU scalability of an operating system is not unlimited – that means that the host OS can't fully benefit from this added computing power. Also, the scalability of 32bit version of Windows is limited, due to the kernel constrains discussed above. Thus, the 32bit version can't fully benefit from an erroneous amount of memory, because of architectural limitations. Also, the PAE access used to address memory above 4GB is associated with some extra overhead.

As a result, for typical consolidation scenarios sticking with mid-range x86-64 servers would probably justify the hardware cost the best. Recent case studies research shows that Virtuozzo customers typically deploy Virtuozzo for Windows on dual socket dual or quad core servers with 4-12 GB of RAM for 32bit version having average VE density around 20 VEs per server, while 64bit version tends to be deployed on slightly more powerful hardware with dual or quad core and 8-32GB of memory, with VE density between 30 and 50 VEs per server.

References

Ruling out memory-bound problems http://technet.microsoft.com/en-us/library/7a44b064-8872-4edf-aac7-36b2a17f662a.aspx

Benefits of Microsoft Windows x64 Editions http://download.microsoft.com/download/9/e/9/9e9b9c02-4e68-46d5-83e3-86258e9e780c/Benefits_of_Windows_x64.doc

Analyzing performance data http://technet2.microsoft.com/windowsserver/en/library/392ddbd6-da23-4fa5-bf3f-690d6b83bc5a1033.mspx?mfr=true

Solving performance problems http://technet2.microsoft.com/windowsserver/en/library/39bcc83c-bb42-46db-b9dd-e0c8a11bfb1b1033.mspx?mfr=true

Memory pool troubleshooting http://kb.swsoft.com/article_206_1675_en.html

Virtuozzo for Windows update VZU35137 http://kb.swsoft.com/en/1722

Using performance monitor http://technet2.microsoft.com/WindowsVista/en/library/5e9c3c1e-6b6b-4140-8dd6-c304c7a2d3651033.mspx?mfr=true

Evaluating Memory and Cache Usage http://www.microsoft.com/technet/prodtechnol/windows2000serv/reskit/prork/prec_evl_yrcn.mspx?mfr=true