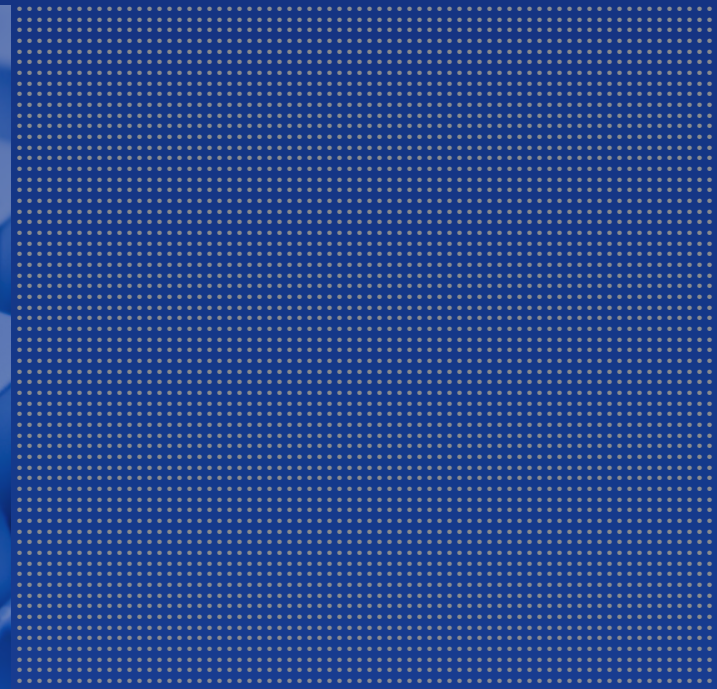


**A R G E N T**  
ENCYCLOPEDIA

# Monitoring Microsoft Hyper-V



## Contents

<b>Microsoft Hyper-V Monitoring Overview</b>	3
Introduction	3
Overall Health and Task Status	5
Where to Monitor Virtual Machine CPU Usage?	7
Monitoring Overall CPU Usage?	10
Monitoring Memory Usage?	12
Monitoring Network Performance?	15
Monitoring Disk Performance?	19
Hyper-V Event Log	22
<b>Appendix A - Microsoft Hyper-V Monitoring Overview</b>	24
Hyper-V Hypervisor Processor Counter Description	24

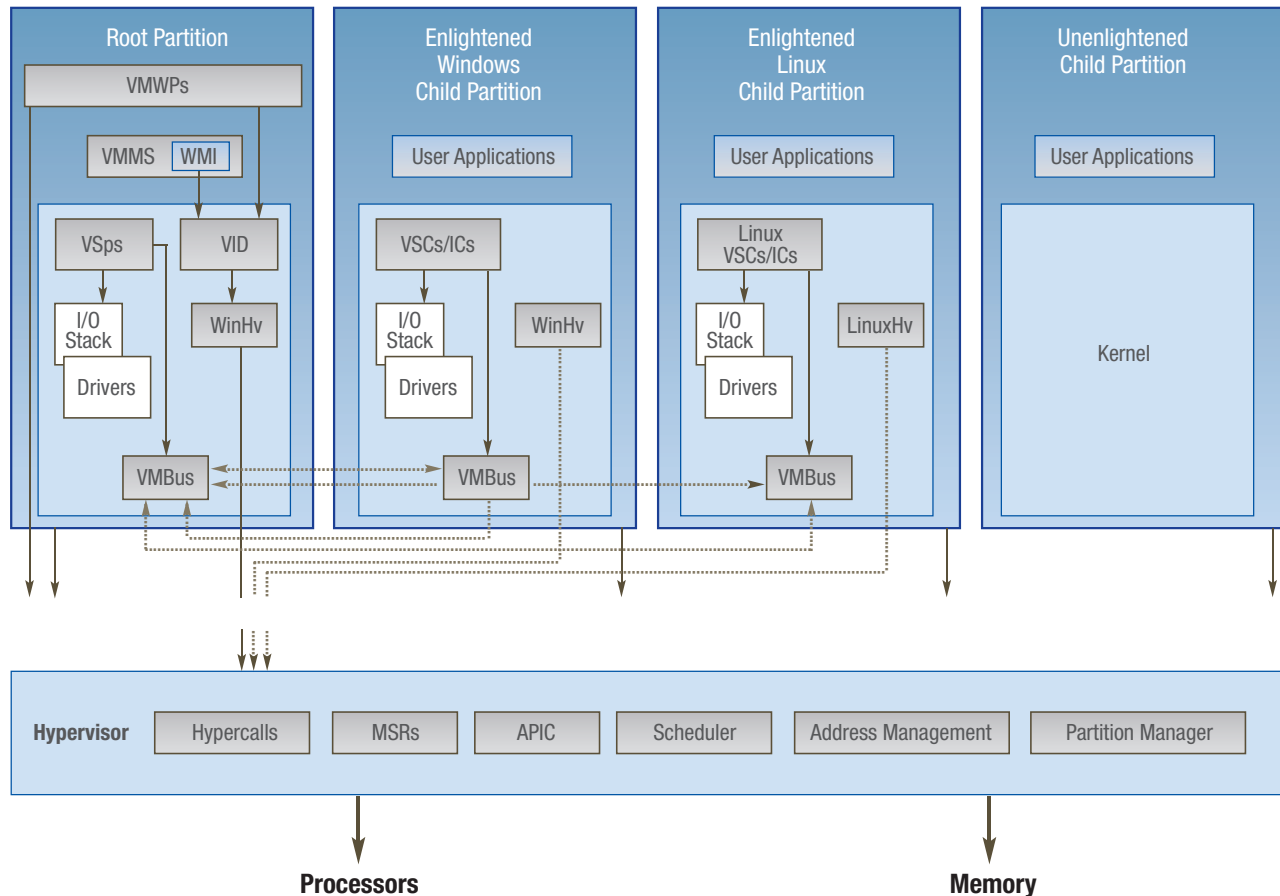
## Microsoft Hyper-V Monitoring Overview

### Introduction

Microsoft Hyper-V is a virtualisation platform like VMware ESX but will run only on 64bit hardware platforms, can run any guest server OS but has support for Windows Server 2008, Windows 2003 SP2, Windows 2000 SP4 and SUSE Linux 10 SP2 (enlightened partitions).

Hyper-V supports isolation in terms of a partition. A partition is a logical unit of isolation, supported by the hypervisor, in which operating systems execute. A hypervisor instance has to have at least one parent or root partition, running Windows Server 2008. The virtualization stack runs in the parent partition and has direct access to the hardware devices. The parent partition then creates the Child or Guest partitions which host the guest OS. A parent partition creates child partitions using the hypercall API, which is the application programming interface exposed by Hyper-V.

### HyperV High Level Architecture



A virtualized partition (guest) does not have access to the physical processor, nor does it handle its real interrupts. Instead, it has a virtual view of the processor and runs in *Guest Virtual Address*, which (depending on the configuration of the hypervisor) might not necessarily be the entire virtual address space. A hypervisor could choose to expose only a subset of the processors to each partition.

You have to understand a little of the Hyper-V architecture to know what needs to be monitored. Hyper-V has the following main components:

- Windows 2008 OS boots the system – this becomes the parent partition.
- VSC – Virtualisation Service Client is responsible for handling emulated devices, managing VM's, servicing I/O, and more.
- VSP – Virtualisation Service Provider resides in the parent partition and provides synthetic device support via the VMBus to Virtual Service Clients (VSCs) running in child partitions.
- Hypervisor is responsible for scheduling Virtual Processors, managing interrupts, servicing timers, and controlling other chip level functions.
- Device Drivers are part of the root and are also installed in guests as part of the Integration Services.

Virtual Devices are managed by the Virtual Motherboard (VMB). Virtual Motherboards are contained within the Virtual Machine Worker Processes, of which there is one for each virtual machine. Virtual Devices fall into two categories, *Core VDevs* and *Plug-in VDevs*. *Core VDevs* can either be *Emulated Devices* or *Synthetic Devices*.

- VMBus facilitates highly optimized communication between child partitions and the parent partition.
- Hypercall API - A DLL (named WinHv.sys) located in the parent partition Windows Server 2008 instance and any guest operating systems which are *Hyper-V aware* (in other words modified specifically to operate in a Hyper-V child partition). Allows the operating system's drivers to access the hypervisor using standard Windows API calls instead of hypercalls.

The rest of this document provides an insight into what counters can be used to provide Hyper-V system performance tracking and health status with the following main areas of interest.

- Overall Health and Task Status
- Virtual and Overall Processor
- Memory
- Network
- Storage
- Event Logs

## Overall Health and Task Status

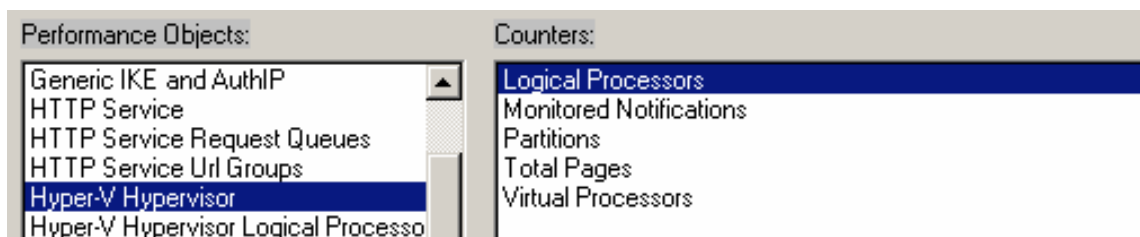
**Hyper-V Virtual Machine Health Summary** - This has two counters; **“Health Ok”** and **“Health Critical”**. If anything is Critical it means some resource (most likely disk) has been exhausted or other unrecoverable error has occurred.

### Performance Rule Is Broken If

Health Critical of Hyper-V Virtual Machine Health Summary GREATER THAN 0.000000

If you encounter a **“Health Critical”** you should take action to figure out what has happened.

**Hyper-V Hypervisor** - Use this counter set to understand how many Logical Processors the system recognizes (**Logical Processor**), the number of virtual machines running (**“Partitions”** – 1), and the total number of Virtual Processors (**Virtual Processors**). The logical processors (LP) are important because they are where all the work is done. They are a representation of the physical processor (core or CPU thread like HT and SMT). The virtual processors (VPs) tell you something about the guests and also if the system is overloaded.



You should make sure the VP to LP ratio does not exceed eight to one because Hyper-V does not currently support beyond this limit. Hyper-V does not set a hard cap so you can exceed it. Just understand you are in largely untested configuration and might see guest failures beyond 8:1.

### Performance Rule Is Broken If

Virtual Processors of Hyper-V Hypervisor As \$Var\_VP  
 Logical Processors of Hyper-V Hypervisor As \$Var\_LP  
 Expr\_VP\_to\_LP\_Ratio( \$Var\_VP / \$Var\_LP ) Greater Than 7.00

Some other current limits to be aware of is WS08 Hyper-V supports only 24 Logical Processors (LPs) and Windows Server 2008 R2 Hyper-V has a current stated limit of 32 LPs.

### Performance Rule Is Broken If

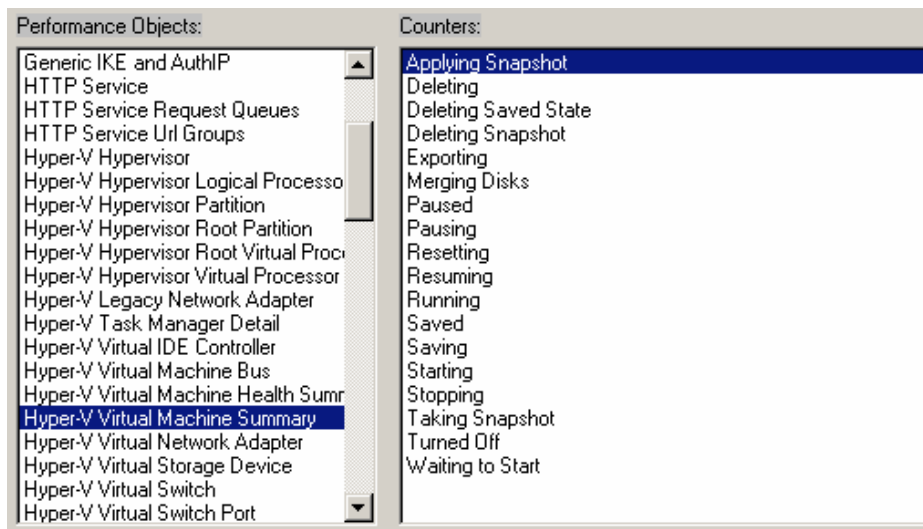
Health Critical of Hyper-V Virtual Machine Health Summary GREATER THAN 0.000000

The number of Partitions that exist can be tracked - This rule will allow the number of partitions on the system to be checked and will be broken when any new partition is added.

The last counter in the **“Hyper-V Hypervisor”** counter set that I use is the **“Total Pages”**. This counter gives an indication of how much Meta data memory the Hypervisor is using to manage the virtual machine.

Unfortunately this counter does not capture all the overhead because another component called the Virtual Interface Driver (VID) also has overhead to manage partitions; the **“Hyper-V VID Partition”** does this.

**Hyper-V Virtual Machine Summary** - State information on the number of VM's starting, stopping and resuming. Gives good insight into what system wide control functions are running.

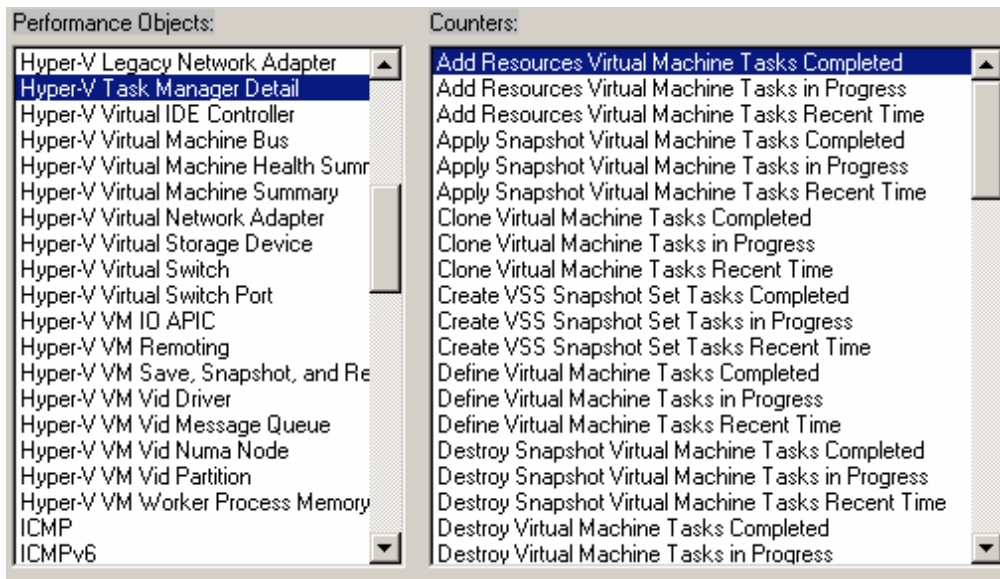


### Performance Rule Is Broken If

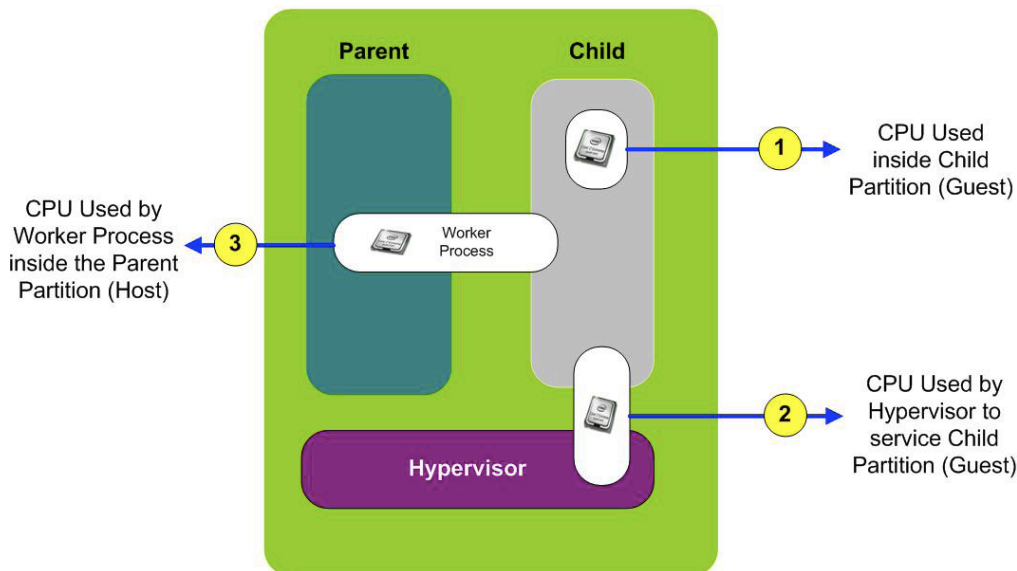
Deleting of Hyper-V Virtual Machine Summary GREATER THAN 0.000000

The Deleting Performance counter could be used to track when a machine is being removed.

**Hyper-V Task Manager Detail** - Gives detail and timing on changes to virtual machines like import, export, clone and any important VM tasks.



## Where to Monitor Virtual Machine CPU Usage?



When you view Task Manager in the parent partition (remembering that it is “just another partition on the system”) it knows nothing about CPU usage that is happening in the hypervisor or in the child partitions.

Virtual Machine CPU usage can be monitored in 3 main places:

1. The first, and most obvious, place that CPU gets used is inside the child partition. This information is displayed in the Hyper-V manager.

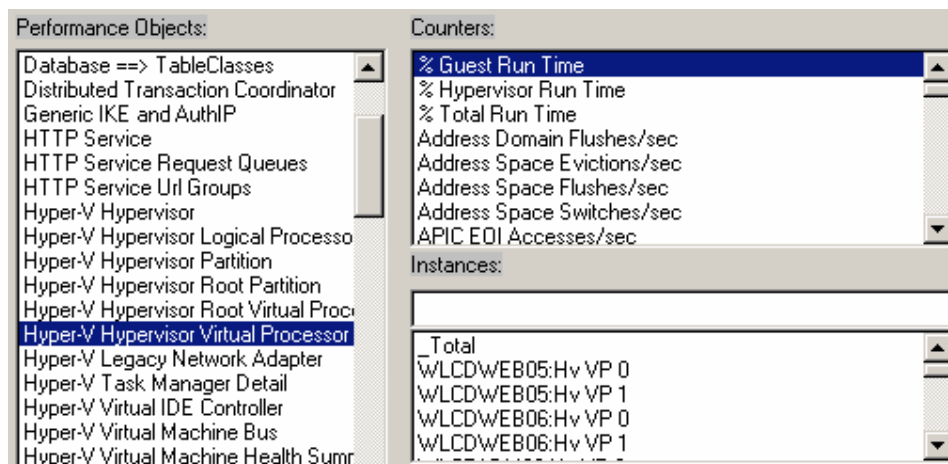
This is available from the Performance Monitor - look at the % Guest Run Time counter under the Hyper-V Hypervisor Virtual Processor section.

**%Guest Run Time** – For guest VM's this is the percentage of time the guest VP is running in nonhypervisor code on an LP or for the \_Total the total across all guest VP's. For the root this is the percentage of time the root VP is running in non-hypervisor code on an LP or for \_Total - the total across all root VP's. If you sum the \_Total for both the guest VP's and root VP's this will equal the % Guest Run Time \_Total of the Logical Processor counter set.

2. The next place that CPU is used is actually the time spent in the hypervisor handling intercepts, switching contexts, etc...

This is available from the Performance Monitor - look at the % Hypervisor Run Time counter under the Hyper-V Hypervisor Virtual Processor section.

**%Hypervisor Run Time** – For guest VM's this is the percentage of time the guest VP is running in hypervisor code on an LP or for the \_Total the total across all guest VP's. For the root this is the percentage of time the root VP is running in hypervisor code on an LP or for \_Total - the total across all root VP's. If you sum the \_Total for both the guest VP's and root VP's this will equal the % Hypervisor Run Time \_Total of the Logical Processor counter set.



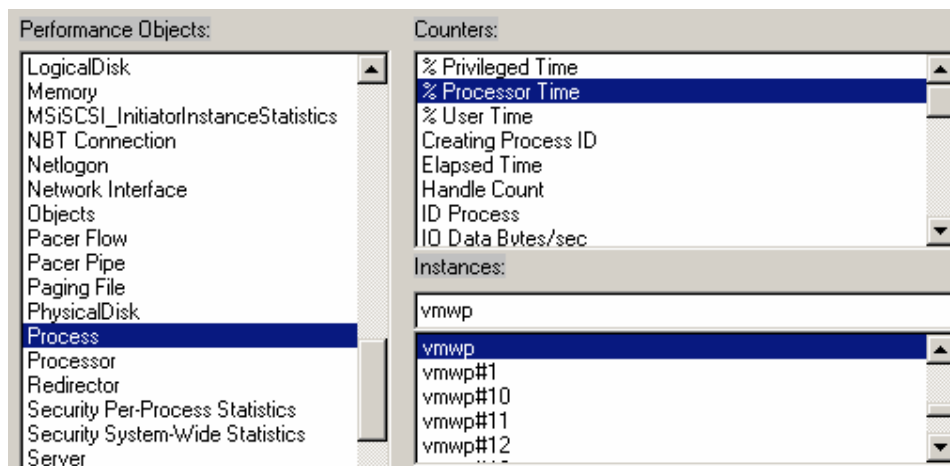


**%Total Run Time** – This is just a sum of %Guest Run Time + % Hypervisor Runtime on a per VP basis. If you sum the %Total Run Time across the Root Virtual Processor and Virtual Processor counter sets it will equal the sum of %Total Run Time from all the Logical Processor counters.

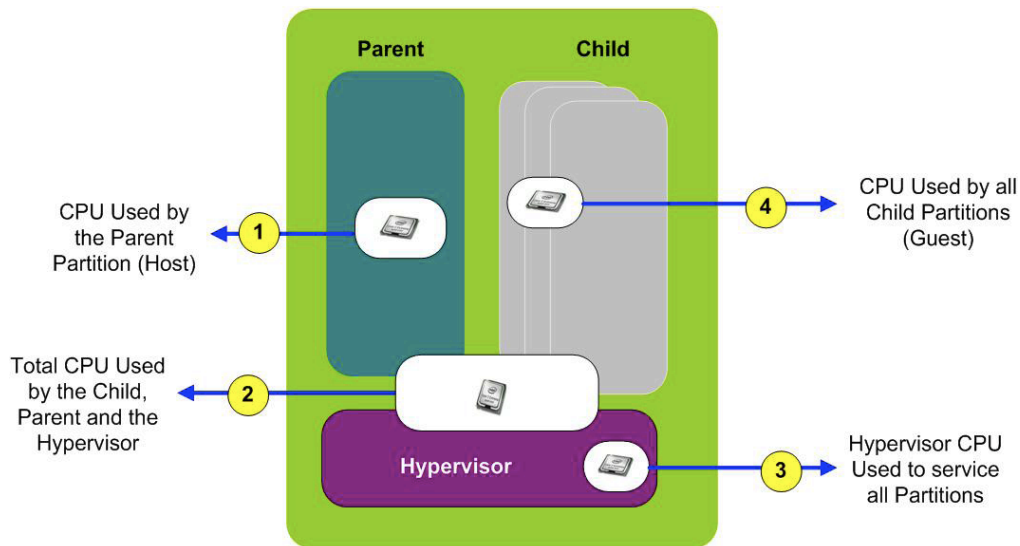
VP counters are prefixed with the name of the partition like this “WLCDWEB05:” followed by the VP id like this “Hv VP 0”. This makes it easy to identify which VP’s go with which partitions.

3. The final place that CPU gets used is in the worker process in the parent partition.

Virtual Machine Worker Processes are started by the VMM Service when virtual machines are started. A Virtual Machine Worker Process (named vmwp.exe) is created for each Hyper-V virtual machine and is responsible for much of the management level interaction between the parent partition Windows Server 2008 system and the virtual machines in the child partitions. The duties of the Virtual Machine Worker Process include creating, configuring, running, pausing, resuming, saving, restoring and snapshotting the associated virtual machine. It also handles IRQs, memory and I/O port mapping through a Virtual Motherboard (VMB).



## Monitoring Overall CPU Usage?

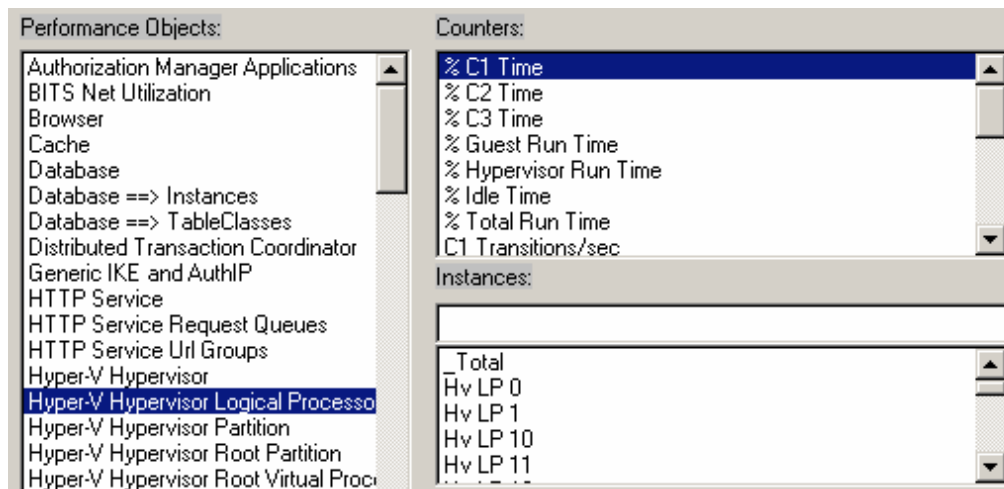


Overall CPU usage can be monitored in 4 main places:

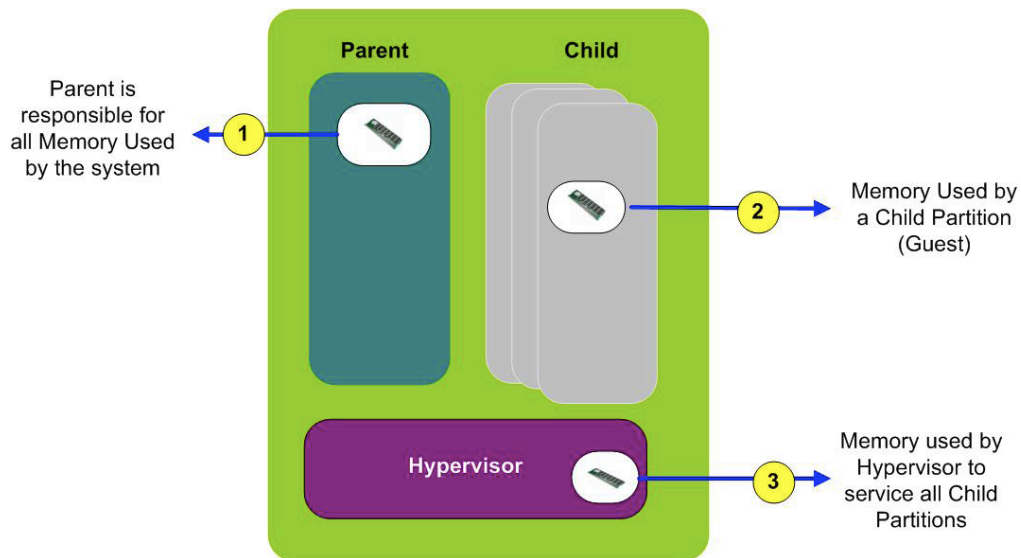
1. The first, and most obvious, place that CPU gets used is inside the parent partition. This is available from the Performance Monitor - look at the “\Processor(\*)\% Processor Time” monitor counter measures the processor utilization of the host operating system only. Remember the parent is involved in all IO. This means that when the parent CPU’s are saturated your whole system is likely saturated. In general you want to see the root CPU lower than 10% utilization and over 50% might indicate an issue.
2. To measure total physical processor utilization of the host operating system and all guest operating systems, use the “\Hyper-V Hypervisor Logical Processor(\_Total)\% Total Run Time” performance monitor counter. This counter measures the total percentage of time spent by the processor running the both the host operating system and all guest operating systems. Use the following thresholds to evaluate overall processor utilization of the Hyper-V environment:

- Less than 60% consumed = Healthy
- 60% - 89% consumed = Monitor or Caution
- 90% - 100% consumed = Critical, performance will be adversely affected

3. To measure total physical processor utilization used by the hypervisor to service the host operating system and all guest operating systems, use the “Hyper-V Hypervisor Logical Processor(\_Total)\% Hypervisor Run Time” performance monitor counter. This counter measures the total percentage of time spent by the processor running the hypervisor for both the host operating system and all guest operating systems.
4. To measure total physical processor utilization used by all the guest operating systems, use the “Hyper-V Hypervisor Logical Processor (\_Total)\% Guest Run Time” performance monitor counter. This counter measures the total percentage of time spent by the processor running the hypervisor for both the host operating system and all guest operating systems.



## Monitoring Memory Usage?



1. The **Memory** counter set allows you to monitor how much memory is being consumed in the root. The root is responsible for managing all memory in Hyper-V. When a VM starts you will see the “**Available Bytes**” go down by at least the amount of memory given to the guest plus around another 16 - 64MB for guests Meta data structures.

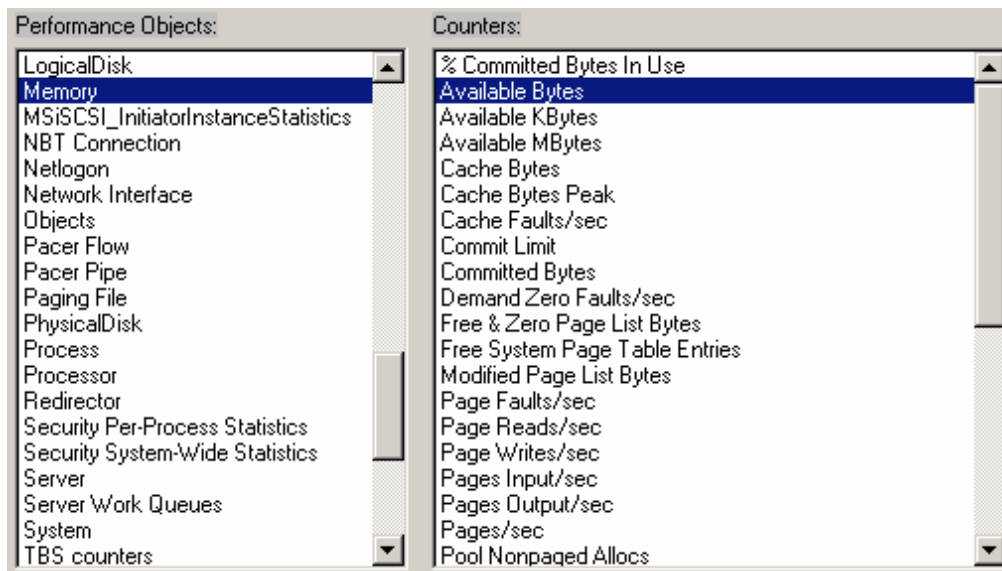
Most recommendations are to monitor the following counters:

- **Available Bytes** - This will give you an idea of how much memory is remaining for guests. There is a reserve of 256MBytes or 512MBytes that the root will always leave outside of guest memory. The exact amount varies but Hyper-V release. So if you find a time when a VM won't start it may be there are too few available bytes to satisfy the reserve.

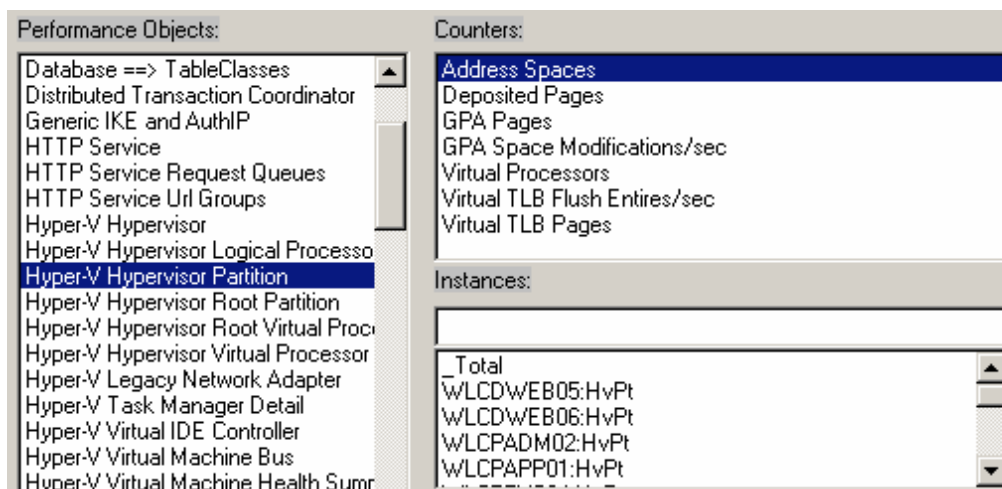
- 50% of free memory available or more = Healthy
- 25% of free memory available = Monitor
- 10% of free memory available = Warning
- < 5% of free memory available = Critical, performance will be adversely affected

- Pages / Sec - This is a measure of memory pressure since it tracks hard faults. Those are page faults that require a disk access. Usually the cause for the number to spike is when there are too few available bytes on the system and processes are competing with each other for physical RAM.

- Less than 500 = Healthy
- 500 - 1000 = Monitor or Caution
- Greater than 1000 = Critical, performance will be adversely affected



2. To measure total memory used by all guest operating systems, use the “Hyper-V Hypervisor Partition (\_Total)\GPA pages” performance monitor counter. This counter measures the total Guest Physical Addresses for all guest operating systems.

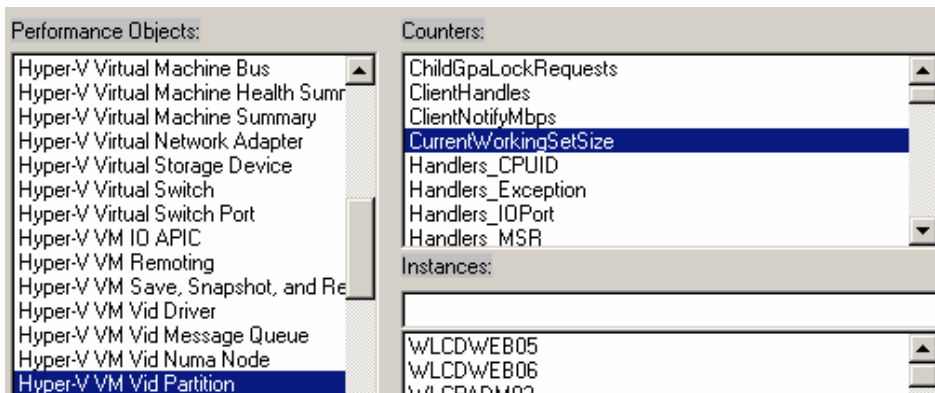


To measure memory used by a single guest operating systems, use the “\Hyper-V Hypervisor Partition (\*)\GPA pages” performance monitor counter.

This counter measures the total Guest Physical Addresses for those guest operating systems.

The \* in this case will be “Guest Name:HvPt” eg  
WLCDWEB05:HvPt

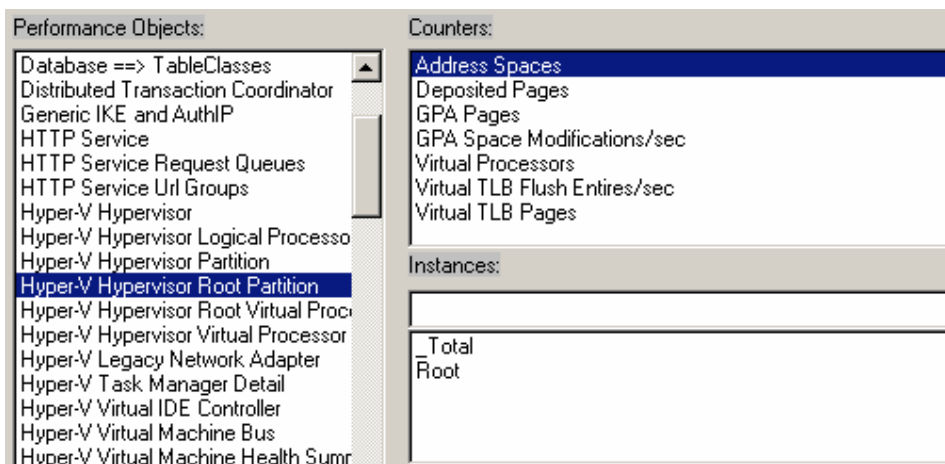
Hyper-V VM Partition is Similar to the “Hyper-V Hypervisor Partition States” but relative to the root’s view on the partitions.



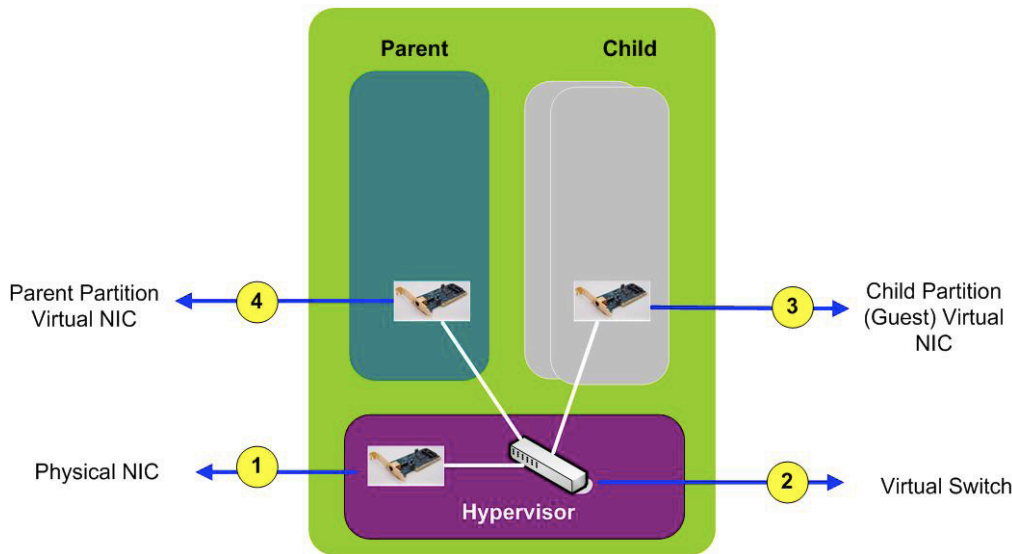
This has Counter set displays information about each child Partition.

3. “**Hyper-v [Root] Partition**” counters determine how much memory the Hypervisor is managing and using on behalf of a VM which includes the guest address space but not all the memory in the worker process and VID partition.

To measure total memory used by all guest operating systems, use the “\Hyper-V Hypervisor Root Partition (\_Total)\GPA pages” performance monitor counter. This counter measures the total Guest Physical Addresses for all guest operating systems.



## Monitoring Network Performance?



1. The network counters are useful for monitoring the overall net working performance on the system. The most important thing to generally monitor is the total throughput counters to make sure the NICs are not getting saturated. Once the NICs are saturated your overall system performance will be capped because no more web requests, remote storage requests, queries, etc can be received than what is currently being handled.

The counter set “**Network Interface**” gives the overall performance of physical device where as the other counter sets represent the activity of the virtual switches and network adapters in the VM’s. For the **Network Interface** the following are the top level counters to monitor:

- Bytes Total / Sec
- Output Queue Length
- Packets Outbound Errors
- Packets Receive Errors

Mostly you want to make sure the network is not saturating and that the error counts are low. If the error counts grow rapidly you might have too much load on the system or some problem in end- toend connectivity (including hardware).

**\Network Interface (\*)\Bytes Total/sec** – The percentage of network utilization is calculated by multiplying Bytes Total/sec by 8 to convert it to bits, multiply the result by 100, then divide by the network adapter's current bandwidth. Use the following thresholds to evaluate network bandwidth utilization:

- Less than 40% of the interface consumed = Healthy

---

- 41%-64% of the interface consumed = Monitor or Caution

---

- 65-100% of the interface consumed = Critical, performance will be adversely affected

**\Network Interface (\*)\Output Queue Length** – The output queue length measures the number of threads waiting on the network adapter. If there are more than 2 threads waiting on the network adapter, then the network may be a bottleneck. Common causes of this are poor network latency and/or high collision rates on the network. Use the following thresholds to evaluate output queue length:

- 0 = Healthy

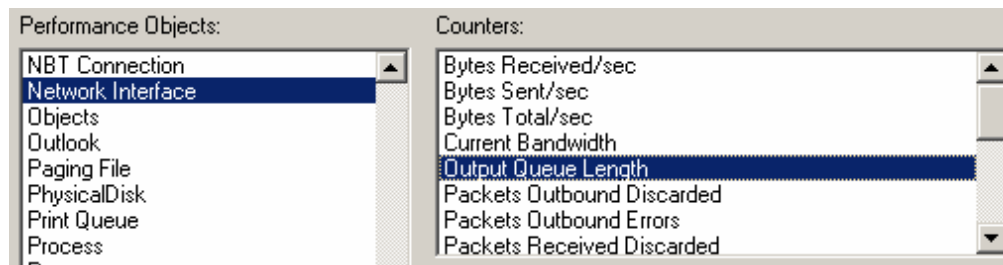
---

- 1-2 = Monitor or Caution

---

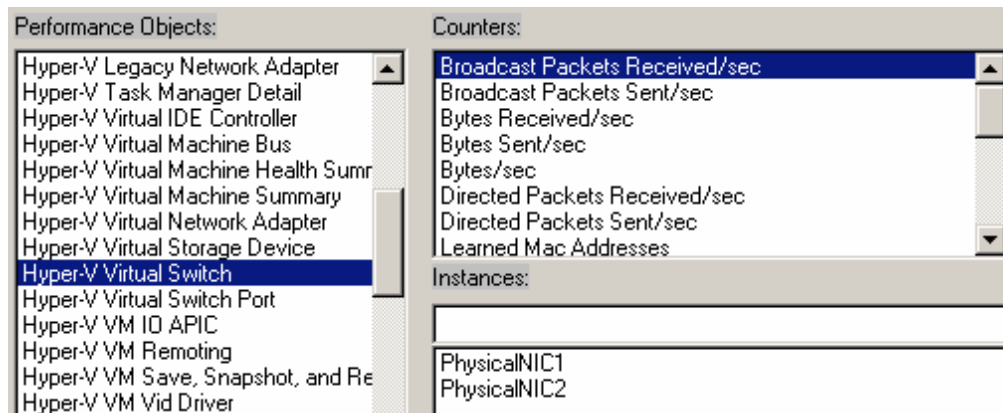
- Greater than 2 = Critical, performance will be adversely affected.

If an output queue length of 2 or more is measured, consider adding one or more physical network adapters to the physical computer that hosts the virtual machines and bind the network adapters used by the guest operating systems to these physical network adapters.





2. Hyper-V Virtual Switch - A virtual switch can attach to one and only one physical NIC. Each Virtual / Legacy NIC plugs into a virtual switch. This counter set gives details on what the switch is doing and flows of sends / receives.



NOTE: Hyper-V Virtual Switch Port - Similar to the virtual switch stats but details just a single “port” aka attached virtual NIC.

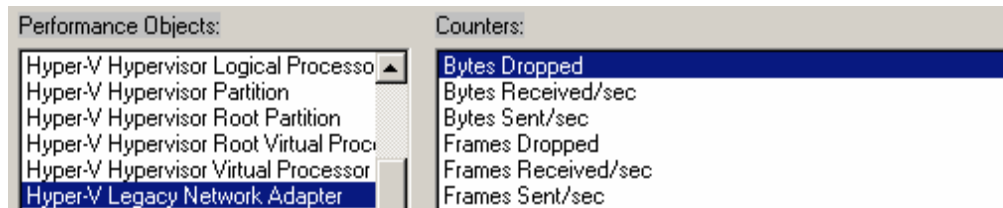
3. If a network adapter on the Hyper-V root partition is busy as indicated by the Network Interface performance monitor counters mentioned above (1), then consider using the “Hyper-V Virtual Network Adapter(\*)\Bytes/sec” performance monitor counter to identify which virtual network adapters are consuming the most network utilization.

The “**Hyper-V Virtual Network Adapter**” and the “**Hyper-V Legacy Network Adapter**” counter sets allow you to see how much ingress and egress a VM is doing.

There are two counter sets because there are two types of virtual network card you can assign to a VM. If you assign a Legacy Network Adapter then the counter set you should use is the “**Hyper-V Legacy Network Adapter**”. In general you should not use the network adapter type because it is not enlightened, creates a lot of CPU load in the root, and is generally slower than the Network Adapter. The challenge is you need the Legacy Network Adapter to get a VM working before installing Integration Services. Once your VM is working with Integration Services you should use the Network Adapter and the “**Hyper-V Virtual Network Adapter**” counter set. Keep in mind Windows Server 2008 and Windows Server 2008 R2 both have integration Services pre-installed.

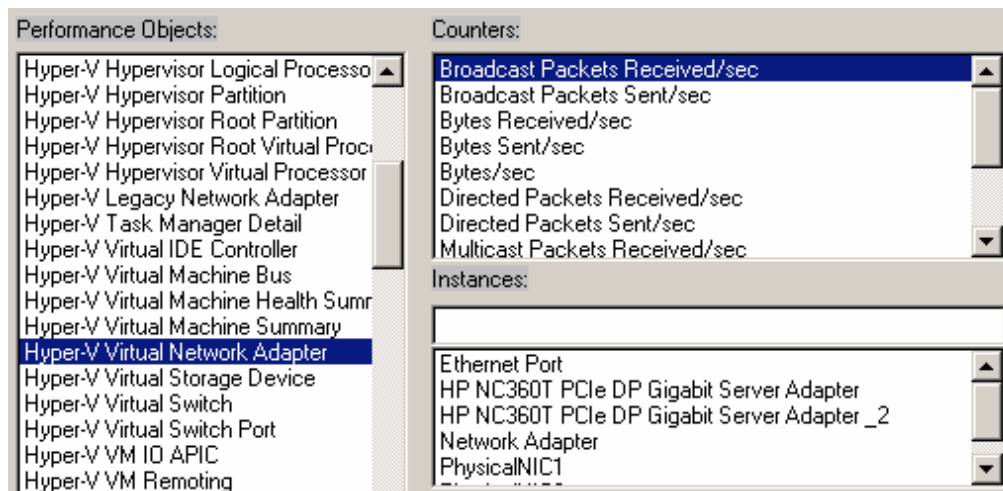
The “**Hyper-V Legacy Network Adapter**” counters that can be to monitor are:

- Bytes Dropped
- Bytes Sent / Sec
- Bytes Received / Sec

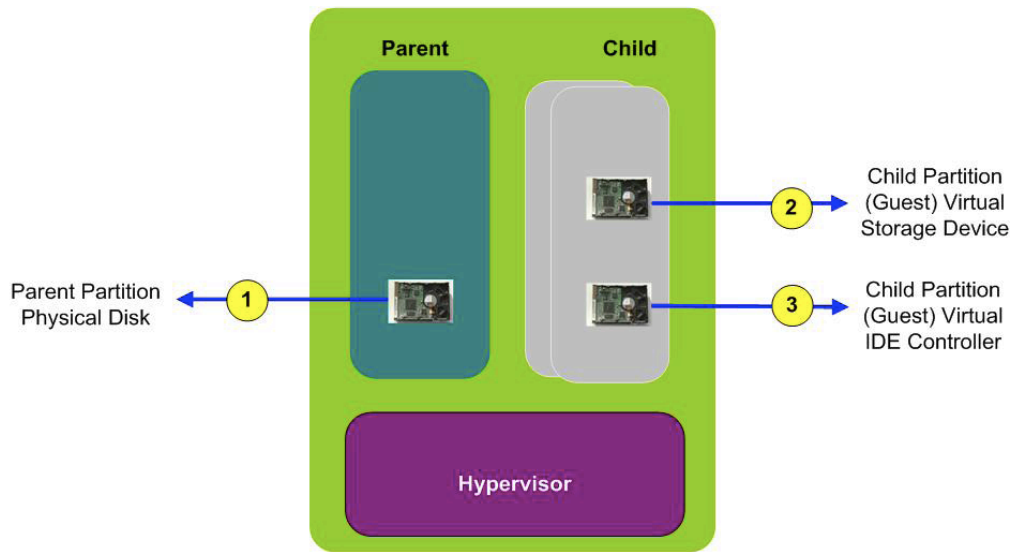


In the “**Hyper-V Virtual Network Adapter**” you should monitor:

- Bytes / Sec



## Monitoring Disk Performance?



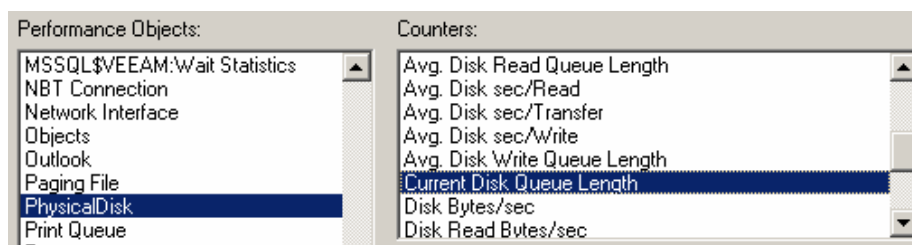
The storage counters are useful for monitoring the overall system disk performance as well as for each VM.

1. The **"Physical Disk"** set will give overall storage performance on the system. These can also be used inside the VM to monitor guest performance. The most useful are:

**"Current Disk Queue Length"** should be around two per drive. If you have a RAID 10 volume with 4+4 (total of 8 drives). Then a queue length of 16 is reasonable. A queue length of 32 might indicate this disk is saturated and is the bottleneck in the system.

**"Disk Bytes / Sec"** you should generally expect to see about 10MB/sec per drive which is a fairly safe number for most drives. So for the RAID 10 4+4 a throughput of around 80MB/sec is reasonable.

**"Disk Transfers / Sec"** and expect to see about 100 IO's per second (IOPs) per drive. For the RAID 10 4+4 around 800 IOPs for 8Kbytes reads and writes is reasonable. Generally writes are a bit lower.



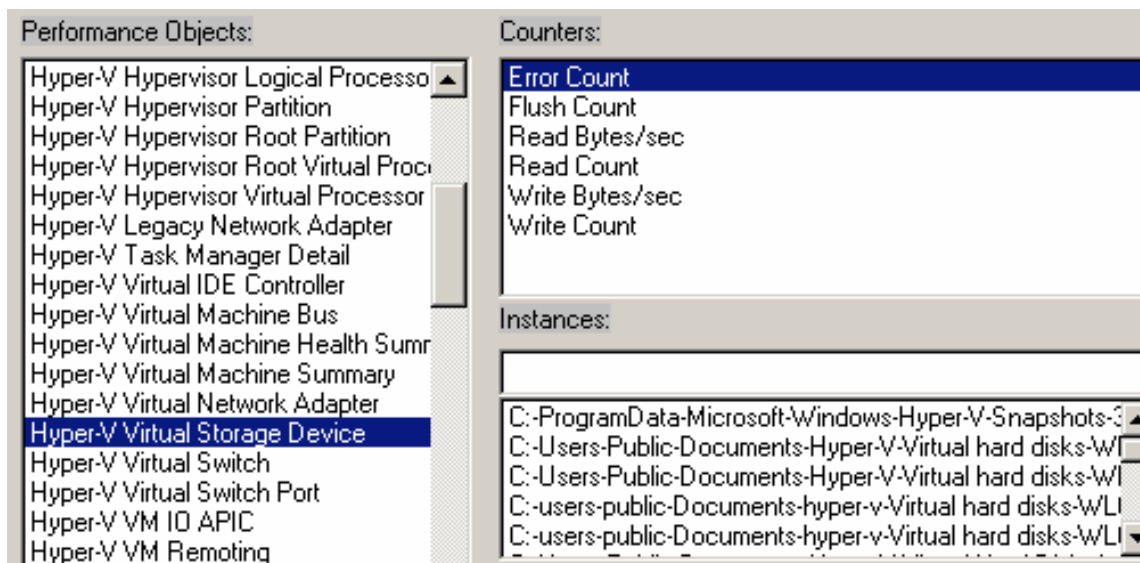
There are two Hyper-V storage counter sets because of how storage works in Hyper-V.

In Hyper-V there are two virtual storage buses for VM's. One is IDE and one is SCSI. The Virtual IDE counters show up in the **"Hyper-V Virtual IDE Controller"** counter set unless Integration Services are loaded and then you will see the activity for both virtual IDE and SCSI in the **"Hyper-V Virtual Storage Device"** counter set. If you don't have integration services installed the only the "Hyper-V Virtual IDE Controller" will show the VM disk activity.

2. For the **"Hyper-V Virtual Storage Device"** there are:

- Error Count
- Flush Count
- Read Bytes / Sec
- Write Bytes / Sec
- Read Count
- Write Count

The Error count should always be zero for the virtual storage device.

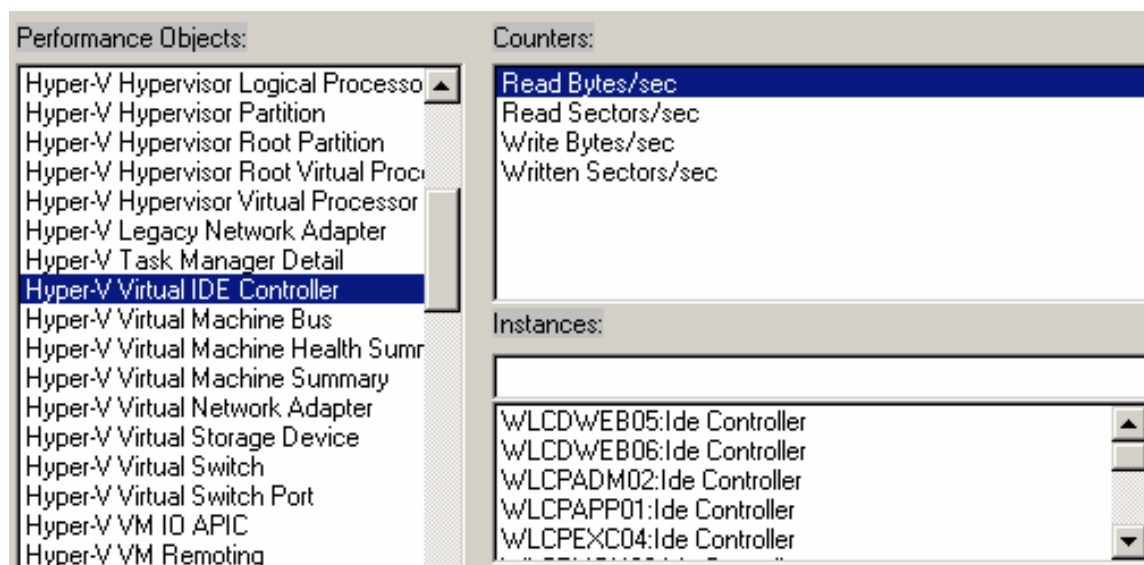


3. For the **"Hyper-V Virtual IDE Controller"** there are:

- Read Bytes / Sec
- Write Bytes / Sec
- Read Sectors / Sec
- Write Sectors / Sec

3. For the “**Hyper-V Virtual IDE Controller**” there are:

- Read Bytes / Sec
- Write Bytes / Sec
- Read Sectors / Sec
- Write Sectors / Sec



Hyper-V logs a lot of useful information if you need to diagnose a problem, all Hyper-V event logs are stored in the Event Viewer under “Applications and Services Logs”, “Microsoft”, “Windows”:



- This section is for anything that relates to virtual machine configuration files. If you have a missing or corrupt virtual machine configuration file - there will be entries here that tell you all about it.

- This section tells you about actions and changes that happen because of Hyper-V clustering.

- This section is used for hypervisor specific events. You will usually only need to look here if the hypervisor fails to start - then you can get detailed information here.

- This section is used by the image management service to log information about virtual hard disk operations - like creating, converting and editing virtual hard disks. If you have problems creating or editing a virtual hard disk - look here.

- **Hyper-V-Integration:**

This section is used to log events that relate specifically to integration services.

- **Hyper-V-Network:**

This section is used for events relating to virtual networks. You will see information about the creation and configuration of virtual networks here (as opposed to virtual network adapters).

- **Hyper-V-SynthNic:**

This is the section where information about virtual network adapters. You will see entries in here each time a virtual machine with virtual network adapter's powers up. You will also see entries here if a virtual machine fails to power on because of a configuration issue with its network adapters.

- **Hyper-V-SynthStor:**

This section is to do with virtual hard disks that are associated with running virtual machines (it is the storage equivalent of the SynthNic section).

- **Hyper-V-VMMS:**

This section is where the virtual machine management services files its events.

- **Hyper-V-Worker:**

This section is used by the worker process that is used for the actual running of the virtual machine.

Now - two pointers to keep in mind while looking at the Hyper-V event logs:

- If you do not know where to start, start with Hyper-V-VMMS. As the central management service it usually has something to say about everything.
- Learn to follow the trail of event logs. If a virtual machine fails to start because of a storage issue there will usually be events logged in the VMMS, Worker and SynthStor sections - and reading all of the events can usually provide better insight into the problem than just reading some of them.

## Appendix A - Microsoft Hyper-V Monitoring Overview

### Hyper-V Hypervisor Processor Counter Description

**%Guest Run Time** – This is the percentage of time guest code is running on an LP or for the `_Total` the average percentage across all LP. For example if you have 2LP and one VM running CPU tests you might see the value be 95% for LP(0), 0% for LP(1) and 47.5% for the `_Total`. For this you can see you VM is running on LP(0).

**%Hypervisor Run Time** – This is the percentage of time the Hypervisor is running on an LP or for `_Total` the average percentage across all LP. This is similar to % Kernel Run Time in the Processor counter set.

**%Idle Run Time** – This is the percentage of time the LP is waiting for work for `_Total` the average percentage across all LP. This is similar to % Kernel Run Time in the Processor counter set.

**%Total Run Time** – This is just a sum of %Guest Run Time + % Hypervisor Runtime. This counter can go over 100% just slightly (<0.5%). The problem has to do with how performance counters are computed. If you take the current time then value1 and later the end time and value2 this means value2 has the potential to increase between when end time was read and value2 is read. You would change it to get the start time then value1 and later value2 and end time. In this case the number would always be slightly less than 100.

**%C1 Time** – C1 is a power saving mode in a CPU. This counter keeps track of how often the process is able to enter a power saving state when idle. So %C1 Time is the percentage of time the LP is in the C1 state and for `_Total` the average percentage across all LP.

**%C2 Time** – Similar to %C1 Time, C2 is a deeper power state than C1.

**%C3 Time** – Similar to %C1 Time, C3 is a deeper power state than C2.

**C1 Transitions / Sec** – This is the number of times the LP has entered the C1 state in one second or for `_Total` the number of C1 transitions across all LP.

**C2 Transitions / Sec** – Similar to C1 Transitions / Sec, C2 is a deeper power state than C1.

**C3 Transitions / Sec** – Similar to C1 Transitions / Sec, C3 is a deeper power state than C2.

**Hardware Interrupts / Sec** – Number of hardware interrupts per second the LP is processing. `_Total` is the total for all LP. Hardware interrupts are delivered to the root VP's corresponding the LP on which it was received. For example a network card will create an interrupt when a packet is received.

**Total interrupts / sec** – Total number of interrupts of all kinds the LP is processing. For `_Total` this is the total number of interrupts happening on the system per second.

**Monitor Transition Cost** – This is a measure of the cost to enter the Hypervisor via an Intercept on a Logical Processor (LP). For `_Total` it is the total cost across all processors. Intercepts are like User mode to Kernel Mode context switches except here is User/Kernel Mode to Virtual Machine Monitor (VMM) aka Hypervisor mode. The smaller the better. The only real use it has is to figure out the relative performance of processors.

**Context Switches / sec** – These are the number of times a new Virtual Processor (VP) had been scheduled to a particular Logical Processor (LP). For `_Total` it is the total number of VP to LP switches. Ideal time context switches of around 1000 for a single guest running are not uncommon. This is due to the fact the VP will "Halt" and allow something else to run if it has no work to do.

**Scheduler Interrupts / sec** – These interrupts are sent by the Hypervisor scheduler from one Logical Processor (LP) to another to reevaluate their run list. The run list is the list of Virtual Processors (VP) waiting to run on a given LP. This is also a "wake-up" mechanism for an LP that might be sitting idle in a lower power state. `_Total` is the total number of scheduler interrupts happen per second across all LPs.

**Inter-processor interrupts sent /sec** – These interrupts are from one processor to another to get the processor to do memory coherency (like TLB, cache, ...). High counts > 20ish per Logical Processor (LP) can indicate lots of guest pages modification (like page access). `_Total` is the total number of Interprocessor interrupts (IPIs) sent per second.



**Inter-processor interrupts /sec** – This counter is the total number of Inter-processor interrupts (IPI) received per second of a given Logical Processor (LP). \_Total is the total number of IPI's received by all LP.

**Timer interrupts / sec** – There are a number of timers that the Hypervisor supports – APIC timer, PM Timer, ... This is the number of times an LP is interrupted to service a timer interrupt.

Note: ArgSoft Intellectual Property Holdings Limited has created this White Paper for informational purposes only. ArgSoft Intellectual Property Holdings Limited makes no warranties, express or implied, in this document. The information contained in this document is subject to change without notice. ArgSoft Intellectual Property Holdings Limited shall not be liable for any technical or editorial errors, or omissions contained in this document, nor for incidental, indirect or consequential damages resulting from the furnishing, performance, or use of the material contained in this document, or the document itself. All views expressed are opinions of ArgSoft Intellectual Property Holdings Limited. All trademarks are the property of their respective owners.