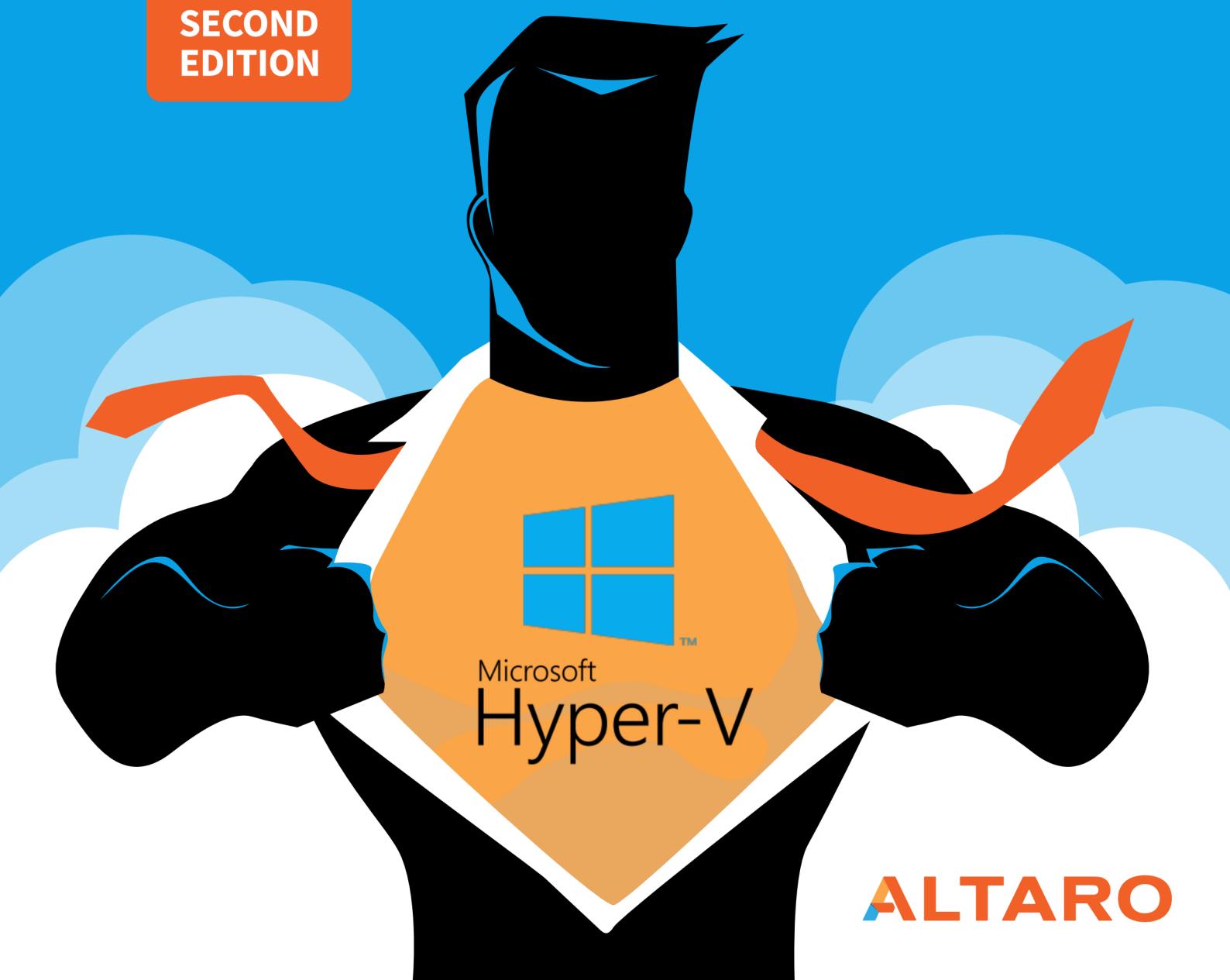


SUPERCHARGING HYPER-V PERFORMANCE

FOR THE TIME-STRAPPED ADMIN

SECOND
EDITION



ALTARO

CONTENTS

| | |
|-------------------------------------------------------------|----|
| INTRODUCTION | 4 |
| PART 1: DIAGNOSING AND REMEDIATING PERFORMANCE ISSUES | 6 |
| Containers – A new way to do infrastructure..... | 7 |
| Tool: Windows Performance Monitor 101 | 8 |
| Tool: PAL is your friend | 10 |
| Tool: Azure Log Analytics (ALA) | 15 |
| Finding runaway servers through Resource Metering..... | 18 |
| Finding Storage performance problems with Storage QoS..... | 19 |
| Finding problem VMs with Windows Admin Center (WAC) | 19 |
| Finding resource problems with Systems Insights..... | 20 |
| Monitoring performance issues with containers..... | 22 |
| Finding and remediating Storage Performance issues | 23 |
| Finding and remediating CPU performance issues..... | 23 |
| Finding and remediating memory performance issues..... | 23 |
| Finding and remediating network performance issues..... | 25 |

| | |
|-------------------------------------------------------------------------------|----|
| PART 2: PLANNING IS BETTER THAN REMEDIATION | 26 |
| Step 0. Planning to run workloads in the cloud | 26 |
| Step 1. Planning Hosts to minimize the risk of performance issues | 28 |
| Step 2. Host Patching | 32 |
| Host Drivers | 33 |
| Step 3. Planning VMs to minimize the risk of performance issues | 34 |
| Step 4. Planning Storage to minimize the risk of performance issues | 35 |
| Step 5. Planning Networking to minimize the risk of performance issues | 38 |
| Step 6. Planning Management to minimize the risk of performance issues | 41 |
| TAKE AWAY ACTION ITEMS: | 43 |
| ABOUT THE AUTHOR: PAUL SCHNACKENBURG | 44 |
| ABOUT ALTARO | 46 |

INTRODUCTION

The aim of this second edition eBook is to help time-constrained and resource-strapped Hyper-V administrators make the most of their existing infrastructure as well as give guidance as to what to look for when planning new hardware. Through practical hands-on tips as well as background information, you will learn how Hyper-V (and virtualization in general) affects performance and how to find issues in storage, CPU, memory and network components.

This is followed by a look at planning hosts, VMs, storage, networking and management for maximum performance. For this second edition we've expanded the coverage to Windows Server Containers, Windows Admin Center (WAC), System Insights in Windows Server 2019, Storage Spaces Direct (S2D) / Hyper-Converged Infrastructure (HCI) and where you can include Azure in your infrastructure.

You can use this eBook in two ways:

- 1.** If you're familiar with performance monitoring and management in Hyper-V you can **use the Table of Contents to jump directly to the page that covers a particular topic**. If you're in the planning phase for a Hyper-V deployment you can skip straight to the planning chapter.
- 2.** Alternatively, if you are relatively new to the idea of monitoring servers for performance it's better to work through the book from start to finish – this should give you a good grounding in the field of performance monitoring and management.

This book should be useful whether you have a small environment with a handful of Hyper-V hosts, whether they are clustered or not, using only WAC with Hyper-V Manager and Failover Cluster Manager for operations. It should also serve you well for larger environments managed through System Center Virtual Machine Manager (VMM) and monitored through System Center Operations Manager (SCOM) or third-party tools.

A quick note about Windows Server versions, this eBook covers Windows Server 2012/2012R2/2016 and 2019. These are the traditional flavor of server, with 5 years standard + 5 years extended support after they've been released, now called Long Term Servicing Channel (LTSC). There's also a corresponding totally free (GUI less) flavor of each release called [Hyper-V Server](#).

Parallel to the LTSC releases Microsoft also releases Semi Annual Channel (SAC) versions of Windows Server twice a year to businesses with Software Assurance (SA). These releases (Server Core flavor only) are only supported for 18 months and are useful for agile business scenarios where the latest features are more important than longevity – for your production Hyper-V hosts we recommend the traditional Windows Server releases.

PART 1: DIAGNOSING AND REMEDIATING PERFORMANCE ISSUES

It's easy to think in today's virtualized, cloudy world that performance monitoring and management doesn't matter. One approach I hear frequently from fellow IT Pros is to simply size every server VM the same (1 vCPU, 1 GB of virtual memory is common) and only change those values upwards if users complain. The other tactic that I've heard is giving each VM 4 virtual processors and 4 GB of memory, which in most cases should lead to happy users, but wasted hardware resources.

A more practical approach is using common sense and application workload sizing guides to design your VMs for your workloads and then knowing how to track down performance issues when they do appear. To do this you need to understand how virtualization, specifically Hyper-V, works.

I've heard it said a few times by engineers at Microsoft that the job of Hyper-V is to lie to Windows.

If you take a Windows Server 2019 (and earlier) and you enable the Hyper-V role, what happens is that a small layer of hypervisor code "slips in" under the installed OS, turning it into a guest VM. This "first VM" is special because it's the parent partition which hosts drivers, negating the need for Hyper-V specific drivers. Normal Windows Server certified drivers work fine for Hyper-V. By contrast, vSphere doesn't use a parent partition and thus needs to have drivers specifically written for it.

Hyper-V "lies" to the parent partition. As an example, on a very large system, not all CPUs are exposed to the root partition (another name for the parent partition), simply because it doesn't need them. And of course, Hyper-V is "lying" to each guest VM about what resources they have access to. This means it's important to use a performance monitoring tool that's not being lied to for tracking down issues.

Say that you have indications (through user complaints or monitoring tools) that a particular workload or VM(s) is having performance issues. The first step is to ascertain if the issue is ongoing, if it only happens at certain times, or at random times. It's also important that you know your environment. In the old world, each server had its own OS and application(s) and probably its own storage, making the task of tracking down the area to start looking for a performance issue much easier.

Today most servers are virtualized, so the performance issue could be lack of resources assigned to the VM. It could also be a “noisy neighbor” VM on the same host or it could be in the shared storage or somewhere in the complex networking system.

So, knowing your environment means understanding how each of your Hyper-V hosts connects to backend storage, how storage is allocated and its performance characteristics, and how the hosts and VMs are connected networking-wise.

You also need to know the characteristics of your workloads in each VM as well as the processors and memory assignment for the problem VMs and other VM(s) on the same host(s). If you're taking over an existing environment there's an [excellent free script](#) which will inventory your stand-alone or clustered hosts and present reports on any issues found.

Task Manager is the first tool most people think of when it comes to performance issues. It is completely useless in a virtual world. In a guest VM it's being lied to by Hyper-V and if you run it on the host, remember Hyper-V is lying to the parent partition as well. So, your first choice should be Performance Monitor – not Task Manager.

CONTAINERS – A NEW WAY TO DO INFRASTRUCTURE

For a long time now, virtual machines have been the go-to tool for IT to run infrastructure and for good reason – it gives us workload isolation and portability (Live Migration), along with easier backup and better hardware utilization. But each VM is a full virtual server with a full OS. The virtual disk files are large, ranging from tens of Gigabytes to hundreds. Each VM needs to be backed up, protected from malware and managed with monitoring and software patches.

A container is a virtual copy of the OS itself.

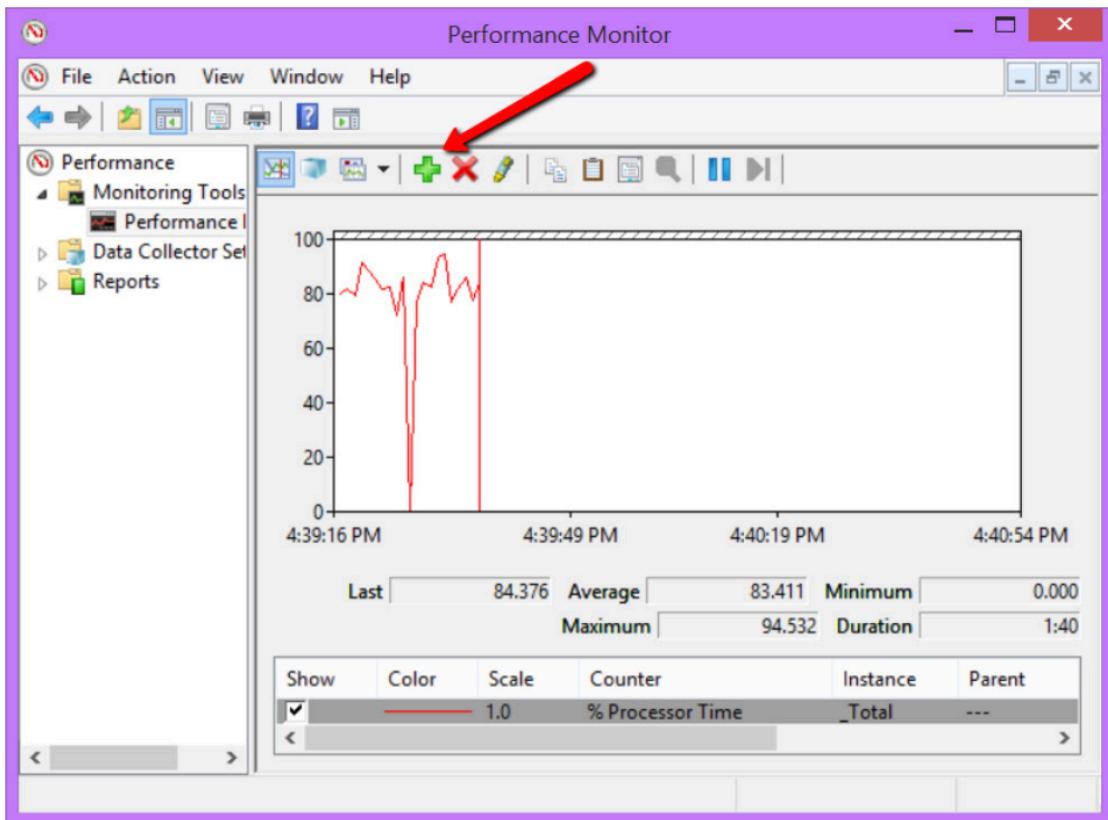
They're much smaller and they start in a few seconds. They're generally much shorter lived than VMs and are spun up and down on a regular basis through a container orchestrator such as Kubernetes. Containers solve several problems but the biggest one is probably "Well, it works on my machine". Differences in platforms between the developer's laptop and production servers no longer matter because containers automatically include everything required to run the application.

On Windows there are two types of containers, [Windows Containers and Hyper-V Containers](#). The former work like their Linux counterpart and separate each copy of the OS – these are great for testing and development work and for deployment on your own hosts where you can be reasonably sure that other containers on the same hosts aren't running malicious code. If, however, you're deploying in a multi-tenant environment, you'll want to use Hyper-V containers which take advantage of security hardened, VM isolation technology to separate each container. The choice of container type is done at deployment time; developers who write code to run in containers don't need to choose as they work identically. The Hyper-V isolated container also means that a Windows Server 2019 host can run both Windows and Linux containers on the same host.

Azure has several services for containers, such as [Azure Container Instances \(ACI\)](#) which lets you run containers without having to worry about the underlying infrastructure and pay per second of usage, [Azure Container Registry \(ACR\)](#) for securely storing container images and automatically replicating them to other regions and [Azure Kubernetes Service \(AKS\)](#) for managing clusters of container hosts and groups of containers.

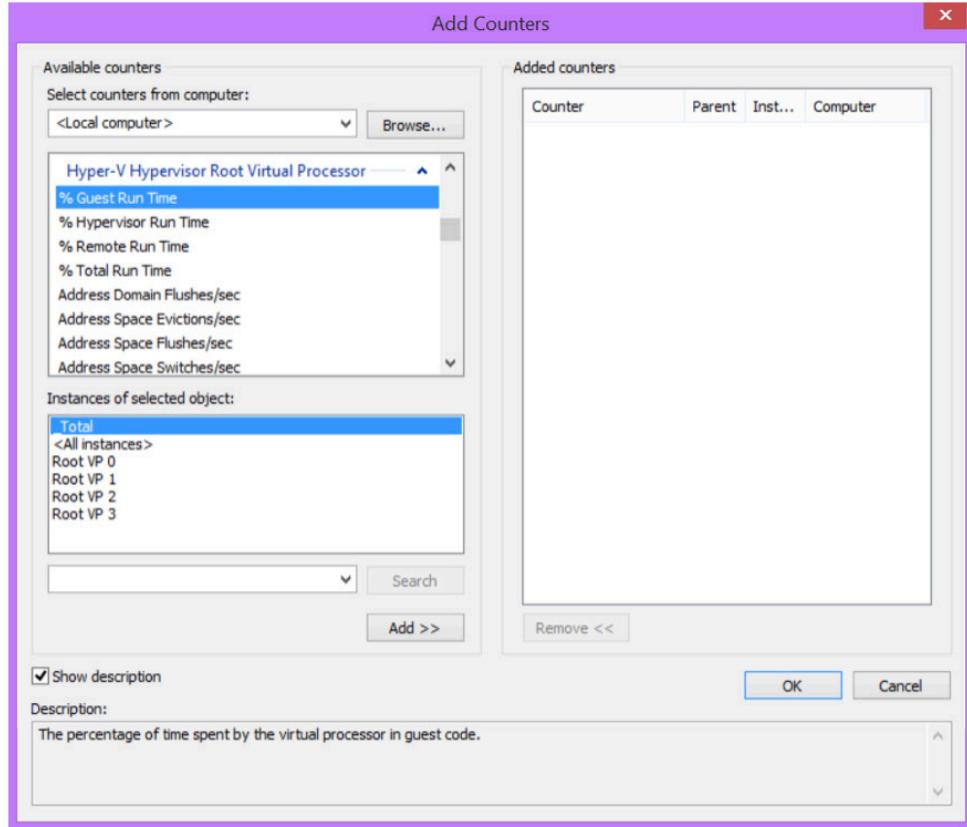
TOOL: WINDOWS PERFORMANCE MONITOR 101

This excellent tool has been in Windows since NT 3.1 (for you young ones: that was the hottest OS on the planet just after the dinosaurs died out!). And since the first version of Hyper-V, it's had specific ways of monitoring that are aware of Hyper-V and thus can see beyond the virtualization layer. If you're tracking down an issue in a VM you should run Performance Monitor **on the Hyper-V host**.



Add counters button in Performance Monitor

When you start Performance Monitor (Start – Run – Perfmon), click on **the green plus sign**. As you can see in the Add Counters dialog box it can run against your local host or against a remote host; in the middle area on the left you pick the object to monitor (Hyper-V Hypervisor Root Virtual Processor in the screenshot). Each object then has multiple counters (% Guest Run Time) and each counter has one or more instances (_Total as well as four Virtual Processors). Once you've picked one or more instances of each counter you need, click Add to start monitoring these particular assets.



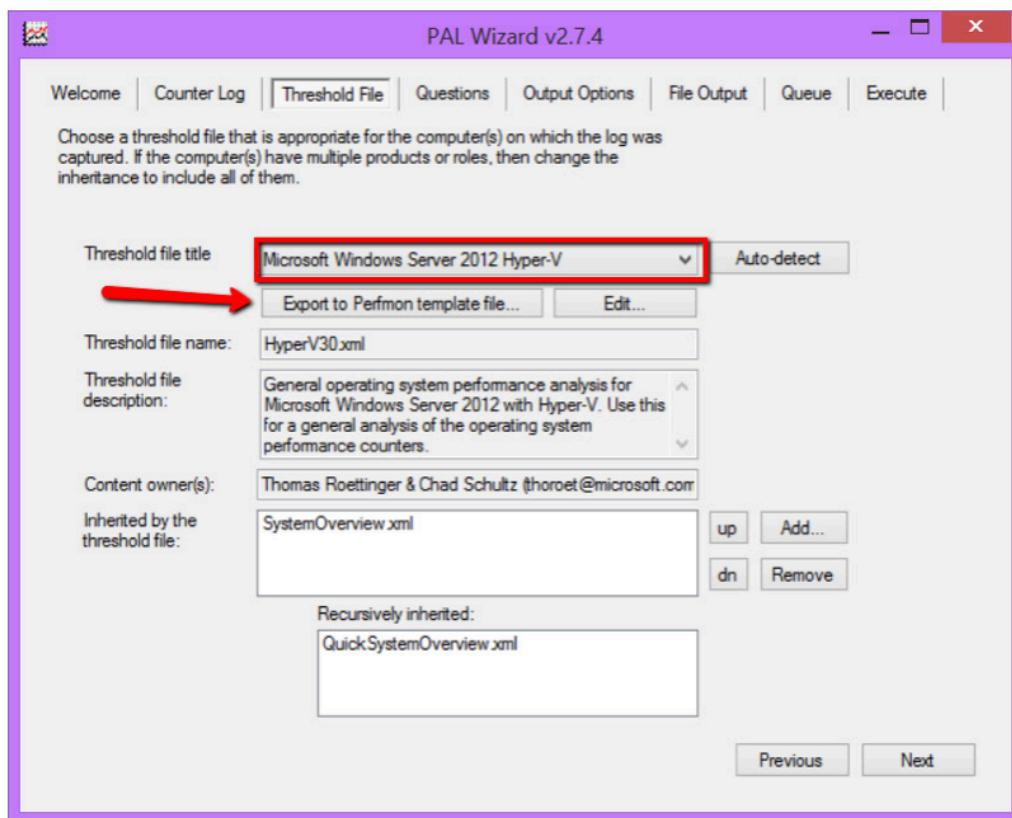
Adding counters in Performance Monitor

This will give you a real-time chart of all the counters you have chosen to display, which can be handy if the problem is happening right now. For most problems, you need to monitor for longer periods of time. This is done using Data Collector Sets; these can be set to gather data for hours or even days. To display the collected data, you click on the View log data in Performance Monitor. This also allows you to select the time range to show. You can highlight a particular graph line with the highlight button (Ctrl + H).

TOOL: PAL IS YOUR FRIEND

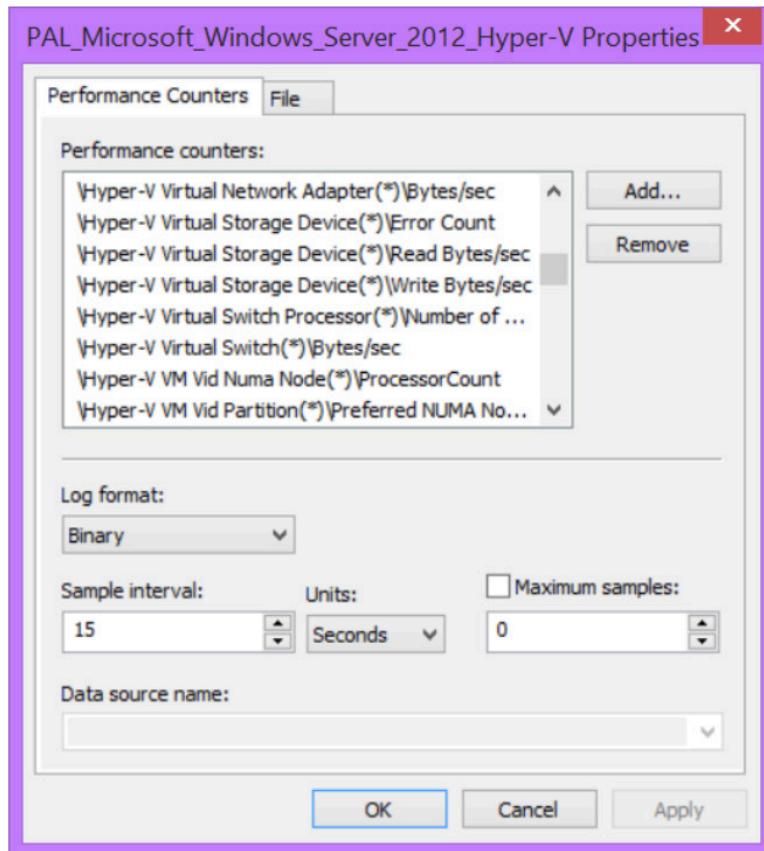
Once you have a feel for Performance Monitor and what it can do, the next step is becoming familiar with [Performance Analysis of Logs \(PAL\)](#), which takes a lot of the guess-work out of knowing which counters to use and which values mean there's an issue and which ones are "normal". PAL is free and builds on Performance Monitor (for gathering the logs) but helps tremendously in the analysis phase.

Once you have downloaded and installed PAL itself and the prerequisites (.Net Framework 3.5 and Chart Controls) run PAL. A core concept in PAL is the use of threshold files and there are supplied templates for several different Microsoft workloads. There isn't a more recent threshold file for Hyper-V than 2012 but this doesn't make it less useful – the settings work fine on Windows Server 2016 and 2019. In the following screenshot, the Hyper-V threshold file is loaded:



Choosing a threshold file in PAL

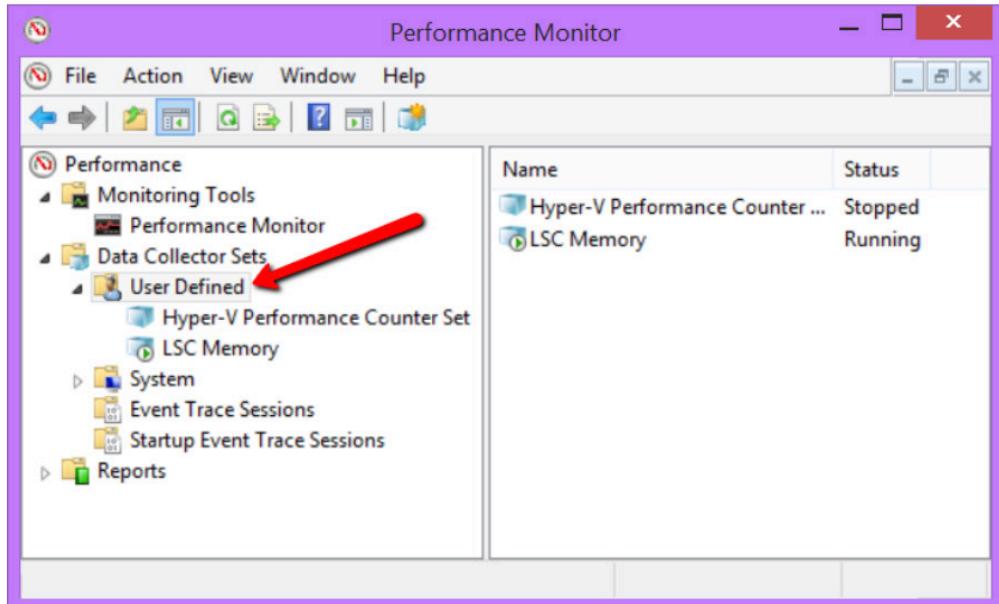
This is a sample of the Performance Counters that are included in the Hyper-V template:



Counters defined in the PAL Hyper-V template

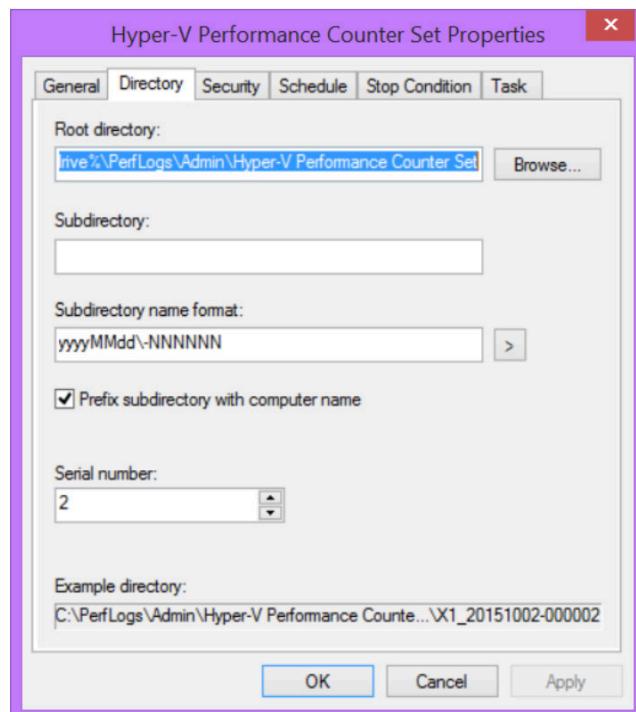
After installation, these are your next steps to start getting some useful information out of PAL:

1. Use the Export button to export the threshold file to a Perfmon template file. In Performance Monitor, expand Data Collector Sets – User Defined, right click and select New Data Collector set.



Loading threshold template in Performance Monitor

2. Give the set a name and select Create from a template and click **Next**. On the next page, select **Browse** and pick the threshold file that you saved and click **Finish**. Highlight your new set and click the green **start** button to capture a log during the peak load time on one or more of your systems. By default, the log data will be saved to C:\Perflogs.



Choosing a folder to save performance log files

Be sure to check that you have ample disk space on your drive – these files can get very big if you capture several hours or days. If you need to save logs on a different drive, right-click on your set and go to **Properties** and then to the **Directory** tab. Here you can set another drive to save the log file to.

- After you have captured the performance data, go back to PAL. On the **Counter Log** tab, load the .blg file created by Performance Monitor. Then, go to the **Questions** tab and enter the relevant information about your host, such as OS and amount of memory. On the **Output Options** tab, select an analysis interval. If you leave it at **Auto**, PAL will divide your analysis into 30 parts so if you captured for 3 hours, each slice will be six minutes long.

The **File Output** tab lets you define where to save files, the **Queue** tab lets you see the PowerShell script that PAL will run, and the Execute tab allows you to add more jobs to the queue or start the process. The end result will be an HTML report with green (OK), yellow and red color coding for problem areas.

| Alerts | | | | | | |
|-------------------------------------------|--|-------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|---------|---------|
| Time Range | | Condition | Counter | Min | Avg | Max |
| 2015.08.23-22:37:01 - 2015.08.23-22:37:16 | | Greater than 1 error per second on this virtual storage device | \V\x1\Hyper-V Virtual Storage Device(C:\-Program Files-Microsoft Learning-20411D-LON-SVR1-Virtual Hard Disks-20411D-LON-SVR1-Encrypted_7B5B492E-953E-44A8-80CA-F464532CD7A7.vhdx)\Error Count | 9 | 9 | 9 |
| | | Greater than 1 error per second on this virtual storage device | \V\x1\Hyper-V Virtual Storage Device(E:\Win2016TP3\VM-Win2016TP3\Virtual Hard Disks-Win2016TP3.vhdx)\Error Count | 9 | 9 | 9 |
| | | Greater than 3 error per second on this virtual storage device | \V\x1\Hyper-V Virtual Storage Device(C:\Users-Paul-Downloads-10514.0.150808-1529.TH2_RELEASE_SERVER_OEMR6_X64FRE_EN-US.ISO)\Error Count | 22 | 22 | 22 |
| | | Value should be below 20,000 per logical processor | \V\x1\Hyper-V Hypervisor Logical Processor(_Total)\Context Switches/sec | 73,641 | 73,641 | 73,641 |
| | | Value should be below 20,000 per logical processor | \V\x1\Hyper-V Hypervisor Logical Processor(Hv LP 2)\Context Switches/sec | 22,108 | 22,108 | 22,108 |
| | | Value should be below 20,000 per logical processor | \V\x1\Hyper-V Hypervisor Logical Processor(Hv LP 0)\Context Switches/sec | 21,738 | 21,738 | 21,738 |
| | | More than 20% privileged (kernel) mode CPU usage | \V\x1\Processor(_Total)\% Privileged Time | 21 | 21 | 21 |
| | | More than 20% privileged (kernel) mode CPU usage | \V\x1\Processor(3)\% Privileged Time | 21 | 21 | 21 |
| | | More than 20% privileged (kernel) mode CPU usage | \V\x1\Processor(1)\% Privileged Time | 26 | 26 | 26 |
| | | More than 20% privileged (kernel) mode CPU usage | \V\x1\Processor(0)\% Privileged Time | 21 | 21 | 21 |
| | | Less than 10% free disk space | \V\x1\LogicalDisk(Harddisk4\Volume5)\% Free Space | 7 | 7 | 7 |
| | | Less than 5% free disk space | \V\x1\LogicalDisk(G):\% Free Space | 2 | 2 | 2 |
| | | Less than 10% free disk space | \V\x1\LogicalDisk(C):\% Free Space | 5 | 5 | 5 |
| | | Greater than 25 ms physical disk READ response times | \V\x1\PhysicalDisk(3 E):\Avg. Disk sec/Read | .054 | .054 | .054 |
| | | Greater than 25 ms physical disk WRITE response times | \V\x1\PhysicalDisk(0 C):\Avg. Disk sec/Write | .032 | .032 | .032 |
| | | Greater than 25 ms logical disk READ response times | \V\x1\LogicalDisk(F):\Avg. Disk sec/Read | .054 | .054 | .054 |
| | | Greater than 25 ms logical disk WRITE response times | \V\x1\LogicalDisk(D):\Avg. Disk sec/Write | .032 | .032 | .032 |
| | | More than 1000 hard page faults per second - this only indicates 4 MB/s or more of disk activity which may or may not be related to page file(s) | \V\x1\Memory\Pages/sec | 2,746 | 2,746 | 2,746 |
| | | Greater than or equal to 64 kB IO sizes. Generally, the larger the IO size, the more data can be transferred per second, but the response times are longer. | \V\x1\LogicalDisk(C):\Avg. Disk Bytes/Write | 482,478 | 482,478 | 482,478 |
| | | Less than 4 MB of free disk space | \V\x1\LogicalDisk(Harddisk4\Volume6)\Free Megabytes | 3,305 | 3,305 | 3,305 |
| | | Less than 500 MB of free disk space | \V\x1\LogicalDisk(Harddisk4\Volume5)\Free Megabytes | 26 | 26 | 26 |
| | | Less than 500 MB of free disk space | \V\x1\LogicalDisk(G):\Free Megabytes | 82 | 82 | 82 |

An example PAL report

PAL also helps you with baselining, which is a critical part of your performance analysis.

When things are going well, users are happy, and the fabric is providing a good level of performance; that's the time to create a baseline. If you later encounter performance related issues, gather some logs and compare against the good baseline, this should quickly demonstrate where the problem lies. In other words, if you don't know what normal looks like, it's hard to spot abnormal.

PAL works on Windows Server 2012 / 2012 R2 / 2016. As it requires .Net 3.5 I couldn't install it on a preview of Windows Server 2019 because the .Net 3.51 installation fails – something that should be fixed before the final version is released.

TOOL: AZURE LOG ANALYTICS (ALA)

One way to do performance monitoring without having to install tools on premises (apart from agents in the monitored servers) is to use a SaaS solution – [Azure Log Analytics](#) (the artist formerly known as Operations Management Suite, OMS).

This service gathers data from a variety of Azure resources (such as your new VM), Windows and Linux VMs running in Azure, any other cloud or your on-premises clusters. Non-Azure machines can be connected directly with the [Microsoft Monitoring Agent \(MMA\) on Windows](#) or the [OMS Agent for Linux](#), alternatively you can use the [OMS Gateway](#) to funnel traffic from non-internet connected servers to ALA. If you have [System Center Operations Manager](#) it can also be used to send data to ALA. Azure VMs just need to be connected to ALA (Workspace Data Sources – Virtual machines – Connect).

Advanced settings

4sysops

Refresh Analytics Save Discard

| COUNTER NAME | SAMPLE INTERVAL | |
|------------------------------------------|-----------------|--------|
| LogicalDisk(*)% Free Space | 10 seconds | Remove |
| LogicalDisk(*)\Avg. Disk sec/Read | 10 seconds | Remove |
| LogicalDisk(*)\Avg. Disk sec/Write | 10 seconds | Remove |
| LogicalDisk(*)\Current Disk Queue Length | 10 seconds | Remove |
| LogicalDisk(*)\Disk Reads/sec | 10 seconds | Remove |
| LogicalDisk(*)\Disk Transfers/sec | 10 seconds | Remove |
| LogicalDisk(*)\Disk Writes/sec | 10 seconds | Remove |
| LogicalDisk(*)\Free Megabytes | 10 seconds | Remove |
| Memory(*)%\ Committed Bytes In Use | 10 seconds | Remove |
| Memory(*)\Available MBytes | 10 seconds | Remove |
| Network Adapter(*)\Bytes Received/sec | 10 seconds | Remove |
| Network Adapter(*)\Bytes Sent/sec | 10 seconds | Remove |
| Network Interface(*)\Bytes Total/sec | 10 seconds | Remove |
| Processor(_Total)\% Processor Time | 10 seconds | Remove |
| System(*)\Processor Queue Length | 10 seconds | Remove |

Adding Performance Counters in ALA

ALA monitors many different types of data from event logs / syslogs but for our purposes we're going to focus on the [performance counters for Windows and Linux](#). You configure your ALA instance to collect particular counters (a default set is automatically installed), you can then configure [visualizations in a dashboard](#) of the data you need, as well as [set alerts](#) if a particular counter goes over a value for a set time. ALA can also monitor performance for [MySQL and Apache HTTP server on Linux](#) and any server application (SQL, Exchange, etc.) that has performance counters on Windows.

```
Perf | where CounterName == "% Processor Time" and ObjectName == "Processor" and InstanceName == "_Total" |  
summarize AggregatedValue = avg(CounterValue) by bin(TimeGenerated, 1m), Computer | render timechart
```

► RUN

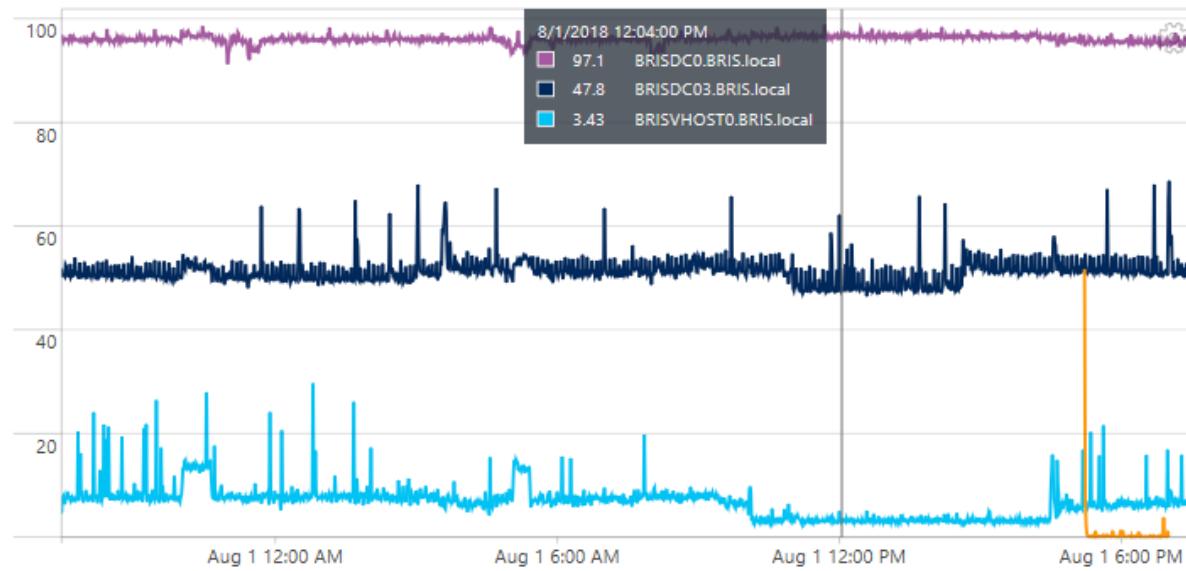
Advanced Analytics

00:00:05.398

4K Results

Chart

Table



SELECT ALL SELECT NONE

AltaroPerf

BRISDC0.BRIS.local

BRISDC03.BRIS.local

BRISVHOST0.BRIS.local

Azure Log Analytics Query based dashboard

ALA is a great way to centralize your performance counter data logging and a very powerful way to analyze, visualize and alert on performance data.

ALA promises Near Real Time performance data collection (10 second intervals) but do some network data throughput testing on your internet connection before configuring a large number of VMs to upload data to the cloud.

FINDING RUNAWAY SERVERS THROUGH RESOURCE METERING

Another helpful addition to Hyper-V in Windows Server (2012 and later) is Resource Metering. It's intended for service providers to track resource usage of each VM (the data follows the VM, even if it is Live Migrated to another host) but can be a life saver if you're trying to track down which VM is using more than its fair share of storage, networking and compute.

These are the steps to follow:

1. To enable it open PowerShell as Administrator and run:

Get-VM | Enable-VMResourceMetering

This will enable metering for all VMs on that host. Alternatively, you can define metering for a particular VM with **Get-VM VMName**.

2. Once the data has been collected use:

Get-VM VMName | Measure-VM

This display shows the gathered information which covers CPU, memory, disk and network usage. If you suspect that a particular VM on a host (or in a cluster) is a “runaway”, using metering for 10 minutes or so and then displaying the results usually pins down the offender.

```
PS C:\> get-vm "Win19b17713" | Enable-VMResourceMetering
PS C:\>
PS C:\> get-vm "Win19b17713" | Measure-VM
VMName      AvgCPU(MHz)  AvgRAM(M)  MaxRAM(M)  MinRAM(M)  TotalDisk(M) NetworkInbound(M) NetworkOutbound(M)
-----      -----
Win19b17713  28          2048       2048       2048       130048        2                  0
```

Resource Metering

FINDING STORAGE PERFORMANCE PROBLEMS WITH STORAGE QOS

In Windows Server 2016 there's a [central storage QoS engine](#) that lets you set policies for VMs or groups of VMs to control their IOPS usage. It also allows you to monitor all VMs across a cluster to see their IOPS usage, through the Get-StorageQoSFlow cmdlet. VMM lets you create and assign QoS policies using a nice UI.

```
PS C:\> Get-StorageQoSFlow | Sort-Object StorageNodeIOPs -Descending | ft InitiatorName, @{Expression={$_.InitiatorNodeName.Substring(0, $_.InitiatorNodeName.IndexOf('.'));Label="InitiatorNodeName"}, StorageNodeIOPs, Status, @{Expression=$_.FilePath.Substring($_.FilePath.LastIndexOf('\')+1);Label="File"} } -AutoSize
```

| InitiatorName | InitiatorNodeName | StorageNodeIOPs | Status | File |
|---------------|-------------------|-----------------|--------|----------------------------|
| SCVMM1711 | HV1RTM | 1408 | ok | SCVMM_PREVIEW_1711_VHD.VHD |
| Win10WAC | HV1RTM | 979 | ok | WIN10WAC.VHDX |
| S2DTestVM | hv2rtm | 872 | ok | 2016.VHDX |
| S2D Test vm2 | HV3RTM | 0 | ok | 2016.VHDX |

Get-StorageQoSFlow report on a S2D cluster

FINDING PROBLEM VMs WITH WINDOWS ADMIN CENTER (WAC)

Windows Admin Center is an all-in-one, web-based, management interface for Windows Server. It marries the best of all the different MMC consoles for managing storage, networking, the registry, event logs, files, services, roles and features, updates and more. It also gives you a single UI to manage clusters, including Hyper-V clusters and [Storage Spaces Direct \(S2D\) clusters and HCI clusters](#). It's extensible with several third-party plug-ins available for monitoring hardware (Fujitsu, DataOn) with more coming. Best of all, it's free, comes in the box for Windows Server 2019 and you can [download it](#) for free for older OSs.

For a Hyper-V host, WAC gives you an overview of processor and memory usage which is very useful as a point-in-time indication if there are any issues. You can also dig into individual VMs and see an overview of their performance. For S2D clusters on Windows Server 2019 (only), WAC provides another benefit – historical tracking of performance and capacity. This lets you investigate performance issues such as when a drive started running slow, which VM used the most memory last week or what's my hit rate for reads and writes for my S2D cache? WAC gives you a UI for this and for scripting you can use [new cmdlets](#) for performance health checks, such as Start-ClusterPerformanceHistory, followed by Get-ClusterPerformanceHistory to give you output that you can use to plot trends.

Due to the usefulness of WAC for 2012 or later Hyper-V environments for overview performance monitoring (not to mention all the other benefits it gives you), my recommendation is to install it today.

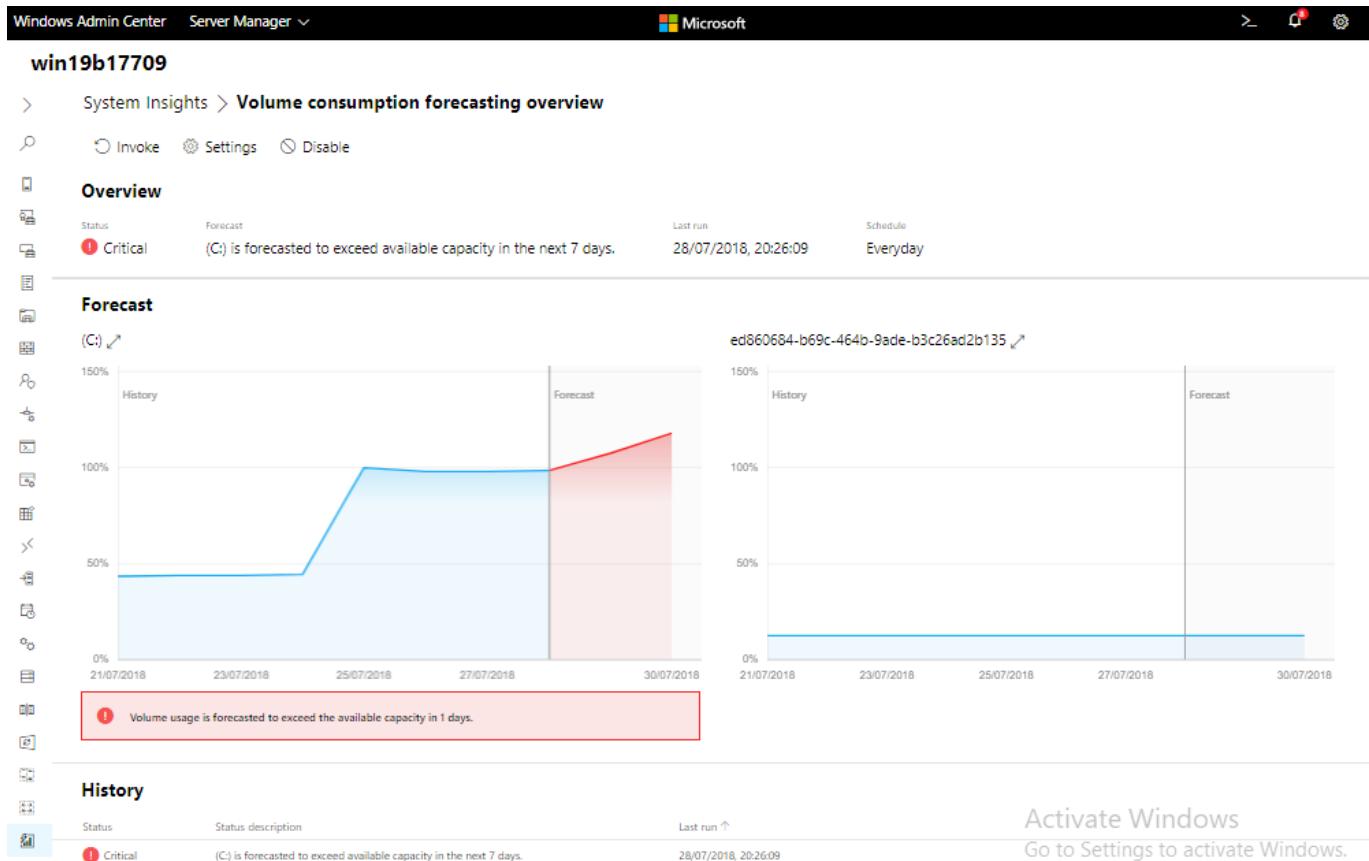
FINDING RESOURCE PROBLEMS WITH SYSTEMS INSIGHTS

If you have systems running Windows Server 2019 (this capability won't be backported to earlier versions of Windows Server), you can add a new tool to your performance management toolbelt: [System Insights](#). This uses predictive analysis of past usage to predict future resource constraints. Currently four areas are investigated; CPU capacity, networking capacity, total storage consumption and per volume consumption. The framework is extensible so expect new areas to be added in the future. There is no reliance on cloud connectivity, all data storage and analysis are local to each server, up to one year's data is kept and you can view the results of the last 30 predictions. WAC will show you the most saturated network first for networking, or the most stressed storage volume when analyzing multiple items.

| System Insights | | | | | PREVIEW ⓘ | | |
|-------------------------------------------------------|--------------------------------|-----------------------------------------|---------------------------------------------------------------------|----------------------|-----------------------|--------|--|
| <input type="radio"/> Invoke | <input type="radio"/> Settings | <input checked="" type="radio"/> Enable | <input type="radio"/> Disable | 4 items 1 selected X | <input type="radio"/> | Search | |
| Capability name ↑ | State | Status | Status description | Last run | | | |
| CPU capacity forecasting | Enabled | Ok | CPU usage is forecasted to remain within the available capacity. | 28/07/2018, 20:25:53 | | | |
| Networking capacity forecasting | Enabled | Ok | Network usage is forecasted to remain within the available capa... | 28/07/2018, 20:26:00 | | | |
| Total storage consumption forecasting | Enabled | Ok | Disk usage is forecasted to remain within the available capacity. | 28/07/2018, 20:26:04 | | | |
| Volume consumption forecasting | Enabled | Critical | (C:) is forecasted to exceed available capacity in the next 7 days. | 28/07/2018, 20:26:09 | | | |

System Insights Capabilities in WAC

You can use [new cmdlets](#) such as Get-InsightsCapability, Invoke-InsightsCapability and Get-InsightsCapabilityResult to script the analysis across machines (schedule it for off-hours as it can be resource-intensive). You can also use the extension for WAC to invoke and manage Systems Insights on a server.



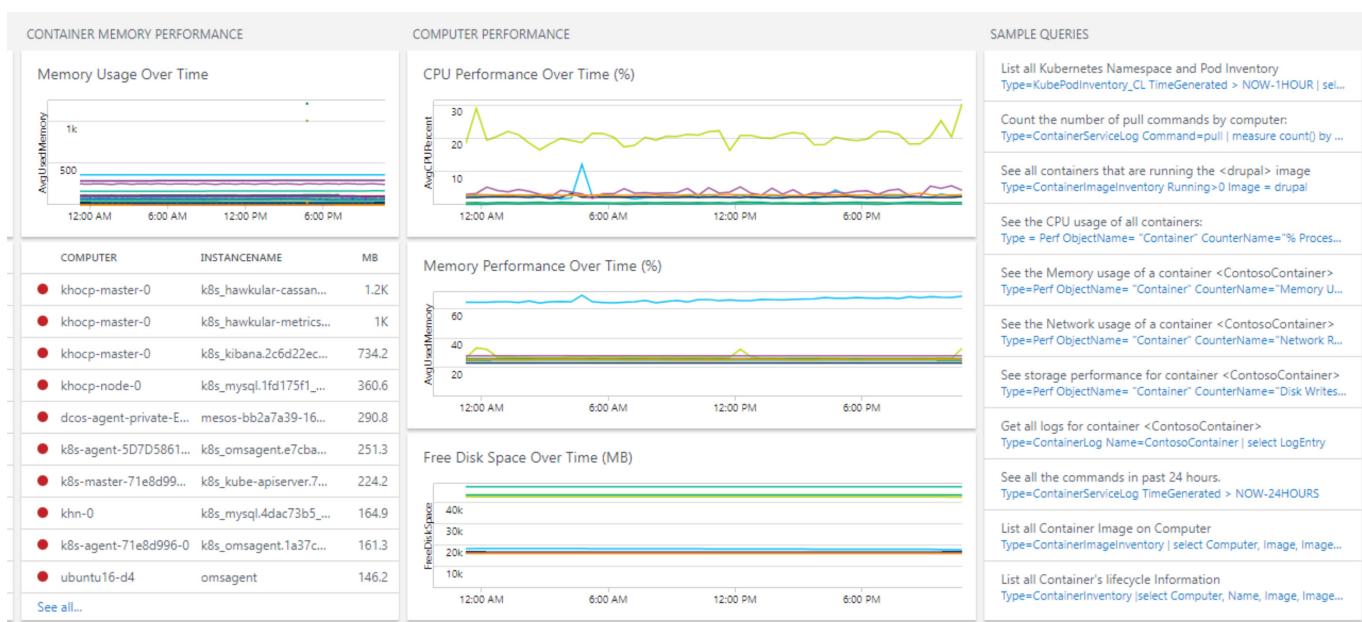
System Insights disk forecast

If the system predicts a resource to be close to hitting a limit you can also [add PowerShell scripts](#) that'll take automatic actions to remediate a problem, for instance, it can extend a volume or run disk cleanup to increase the available disk space.

While System Insights is a great technology (as long as you're running Windows Server 2019) that'll make it easier to predict when you may need to add more capacity to your clusters, it won't help you pinpoint a specific VM that's having or causing trouble so knowing how to use the other tools we show you in this book to track down culprits is still crucial.

MONITORING PERFORMANCE ISSUES WITH CONTAINERS

Containers bring with them different issues for performance monitoring. Unlike VMs, they live for short amounts of time and if there's a problem with a container, it's just killed off (often automatically by an orchestrator such as Kubernetes or [Service Fabric Mesh](#)) and redeployed. The basic concept (taken from DevOps) is treating your infrastructure like cattle, not like pets. Nevertheless, you need to monitor containers and their hosts to keep an eye on performance, one good solution is [Azure Log Analytics \(ALA\)](#) with the Container Monitoring Solution, which uses the [Microsoft Monitoring Agent \(MMA\)](#) on Windows and the [OMS Agent for Linux](#). The hosts and containers can be located on-premises, in Azure or any other cloud.



Azure Log Analytics Container monitoring

For each container you can see network, processor, disk and memory usage. For the entire cluster you can see running, stopped and failed containers. You can drill into the logs from each container and see the processes running in each container as well as CPU and memory performance for each node. Note that AKS has a [slightly different way](#) to connect to Azure Log Analytics for monitoring. Microsoft has some [guidelines](#) for performance tuning Windows Server containers.

FINDING AND REMEDIATING STORAGE PERFORMANCE ISSUES

It's vital that your backend storage can deliver enough performance to each VM, whether that storage is a single host with internal disks, a Storage Spaces Direct cluster or a SAN.

If you suspect that storage is the culprit (and honestly, if you start here you'll be right 90% of the time) there are good Performance Monitor counters to employ.

Use **LogicalDisk Disk Transfers/sec** for the backend storage disk. Note that this will work for local storage as well as block storage that shows up as local drives (iSCSI or FC SAN). If you have virtual disks on SMB file shares you'll need to use Performance Monitor on the physical file server that hosts the shares. If a particular VM is hogging storage IO, you should use storage QoS policies to rein it in (see Planning Storage).

FINDING AND REMEDIATING CPU PERFORMANCE ISSUES

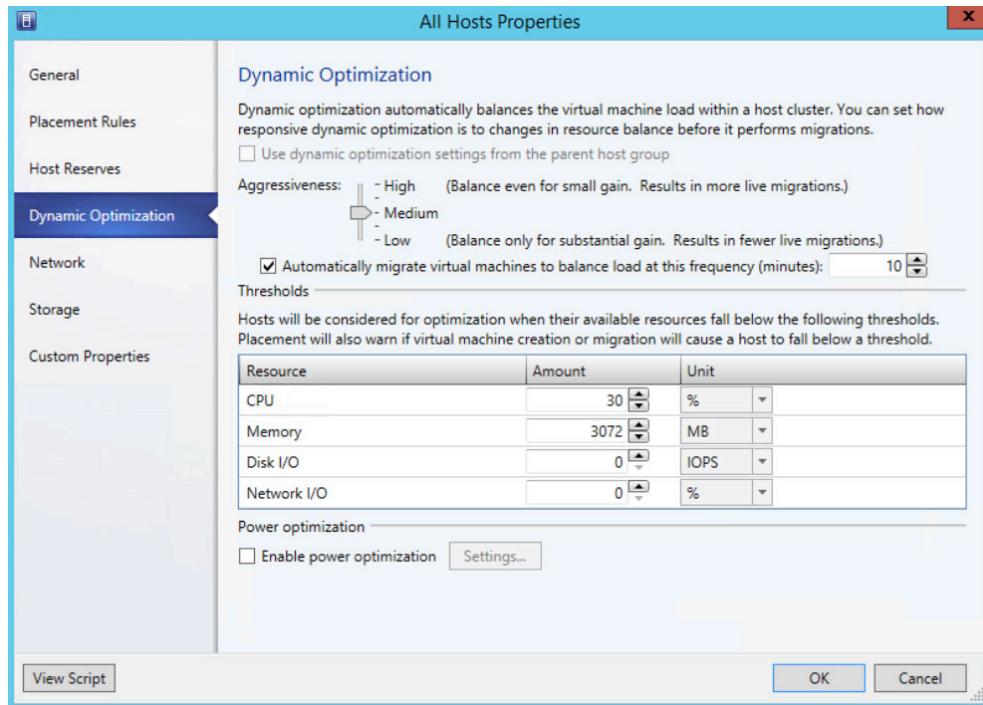
It's rare that processor-related performance issues are the root cause of slow VMs. The exception is misbehaving applications; occasionally you will have a hung process or a poorly written program that hammers the virtual CPU(s).

A good counter to use here is **Hyper-V Hypervisor Logical Processor\% Total Run Time** which is the CPU load from all guests and the hypervisor itself for a given host. You can gather this on a per LP basis or as an aggregate across all LPs. Use **Hyper-V Hypervisor Virtual Processor\% Guest Run Time** to track the CPU load on a per VM basis to find the culprit.

FINDING AND REMEDIATING MEMORY PERFORMANCE ISSUES

There are a number of aspects to memory-related performance problems. The power of [dynamic memory](#) (introduced in Hyper-V 2008 R2) takes a lot of the guess work out of establishing how much memory a particular workload requires. Note that some

applications, particularly those that do their own memory management, don't play well with dynamic memory. An example is [Exchange Server](#). **Memory\Available Mbytes** is your first stop to see if the host is running out of memory. To see if dynamic memory is working hard to balance the load amongst the VMs monitor **Hyper-V Dynamic Memory Balancer\Average Pressure**. It should stay under 80 on a healthy host.



Setting Dynamic Optimization in Virtual Machine Manager

In a clustered environment that's managed by VMM you should definitely enable [Dynamic Optimization \(DO\)](#). Right-click on **All Hosts** in the **Fabric** pane. Select **Properties** and then the **Dynamic Optimization** area. Configure the appropriate frequency and aggressiveness (how often VMM checks to see if VMs should be moved to balance the hosts and how serious it is about making sure all cluster hosts are evenly balanced).

FINDING AND REMEDIATING NETWORK PERFORMANCE ISSUES

Networking performance problems are more commonly encountered because of incorrect configuration. There are a lot of options for configuring your host network connectivity. Teamed or single NICs? [Switch Embedded Teaming \(SET\)](#) in Windows Server 2016 or later? Dedicated network links for each type of traffic or converged networks using *Quality of Service* (QoS) or *Datacenter Bridging* (DCB) for carving up the bandwidth for each flavor? All these different types of network setups make it difficult to give general performance sleuthing tips.

Nevertheless, aiming for **Network Interface (*)\OutputQueue Length** less than 1 is a good rule of thumb. Comparing **Network Interface (*)\Bytes Sent /sec** and **Received/sec** against the **Network Interface (*)\Current Bandwidth** is another good place to start to see if network links are saturated. Remember that RDMA-enabled NICs won't show up in normal performance monitoring and neither will SR-IOV network links.

PART 2: PLANNING IS BETTER THAN REMEDIATION

The first half of this book has covered several ways to track down performance issues. But it is better to build or expand your environment with the right gear in the first place to minimize performance problems.

Here we're going to look at how to plan your hosts, VMs, storage, networking and management to end up with a smoothly running, reliable and highly performant fabric on which your business can depend.

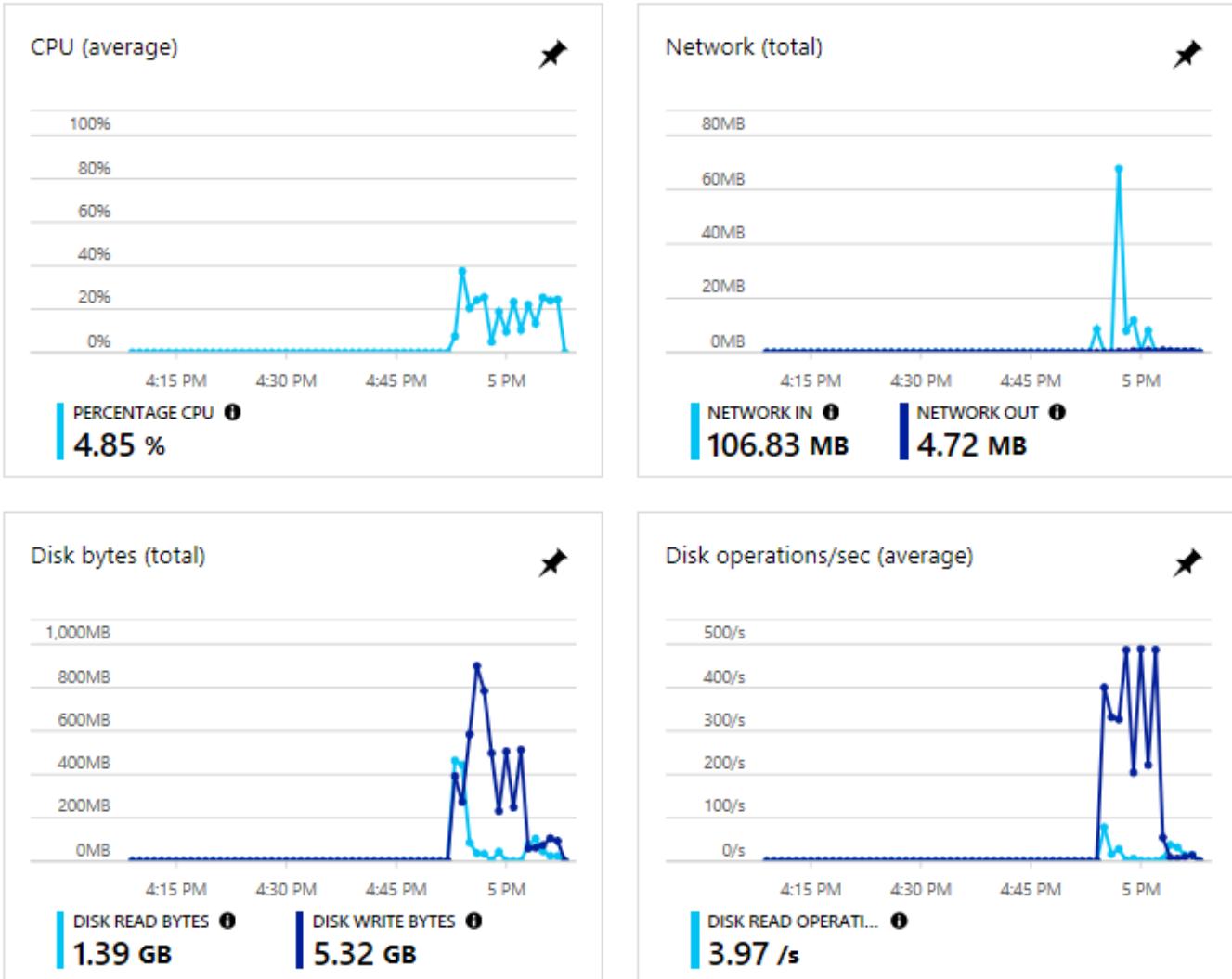
For a comprehensive checklist for performance configuration in Hyper-V, check out [this list](#) on [Altaro's Hyper-V Hub](#).

STEP 0. PLANNING TO RUN WORKLOADS IN THE CLOUD

One valid option today is to outsource the entire hardware layer of your virtualization platform. Run your test / dev or production VMs in Azure (all of Azure after all runs on Hyper-V). Instead of paying a large chunk of money every five years for new physical servers, simply pay per minute for what you use and let Microsoft worry about faulty network equipment, broken disks, etc.

Running a VM in Azure is simple, just follow this [two-part guide](#). Once it's up and running you can see performance information for it at a glance on the Overview pane. It shows hourly data for CPU, Network and Disk by default, but you can look at data from 6 hours up to 30 days. If you enable performance diagnostics when you create a VM (or afterwards by adding the [extension](#)) data will stored in a separate storage account for trending analysis.

Show data for last: 1 hour 6 hours 12 hours 1 day 7 days 30 days



VM performance overview in Azure

For more in-depth performance information you should use [Azure Log Analytics](#) (ALA) as discussed above.

If you're considering moving some of your existing workloads to Azure (rather than creating new VMs), include [Azure Site Recovery](#) (ASR) in your planning. This service can be used for DR, it gathers disk writes to your physical / VMware / Hyper-V machines on premises and uploads them to Azure where you only pay for storage (not running VMs) until a calamity hits your datacenter, when you can simply start the VMs in the cloud. ASR can also be used as a migration tool, the first 31 days (per VM) of usage is free so you can upload a VM to the cloud and then, when the disks are in synch, switch over to the VM in the cloud.

STEP 1. PLANNING HOSTS TO MINIMIZE THE RISK OF PERFORMANCE ISSUES

A few words on planning the setup of your hosts before we dive into performance details. You can run Hyper-V on a single host which can fit some scenarios such as very small businesses or branch offices.

If you don't have the budget for the shared storage required by clustering, an option can be [Hyper-V Replica](#).

This fantastic technology is built into Windows Server 2012+ and lets you replicate running VMs to another Hyper-V host, either across the datacenter or, for true Disaster Recovery purposes, another, distant datacenter. In a small environment you can still provide decent availability by replicating VMs from one host to another in the same datacenter. It isn't clustering, and failover isn't automatic if the primary host fails for some reason, but it doesn't require shared storage.

In larger scenarios you'll want to plan for clustering your Hyper-V hosts for High Availability (HA). For this there needs to be some form of shared storage to house the VMs virtual disks and configuration files. This can be Fibre Channel / iSCSI SAN or SMB file shares or if you're running Windows Server 2016 / 2019, Storage Spaces Direct (S2D, see [Planning Storage](#), below).

To design a new (or expand) a cluster you need to know the number of VMs and their requirements for memory, CPU and storage IO.

You also need to take into account the resiliency of the cluster itself. Think of this as “cluster overhead”; in a two-node cluster you can really only use 50% of the overall capacity of the hardware for workloads. Any less and you won't be able to keep all your VMs running during patching. If there's an outage (planned or unplanned) the remaining node must be able to run the VMs from the downed node as well as its original workload. To manage this, you can really only plan to use half of the memory, processing and storage IO on each node.

In a three-node cluster on the other hand, the load can be spread across the remaining two nodes, resulting in 66% available resources being useable. Four nodes have 25% overhead and so forth. By the time you start reaching eight node clusters you may have a requirement to survive two nodes being down simultaneously, again leading to 25% overhead. The largest cluster size in Windows is 64 nodes. **Given cluster overhead I recommend more nodes rather than a few powerful ones**, although you need to factor licensing costs into your calculations.

Three new features in Windows Server 2016 / 2019 are worth considering when planning your Hyper-V clusters. [Storage Replica](#) (SR) (Datacenter only in 2016, there's an SR "lite" coming in Windows Server 2019 Standard) allows you to replicate data synchronously (up to 150 Km) and asynchronously (anywhere on the planet) from any disk in one location to any disk in another location. This lets you create [stretched clusters](#), where one part of your Hyper-V cluster resides in one location and another part elsewhere, improving your DR preparedness.

[Cloud witness](#) allows you to use an Azure location for a tie-breaking vote for the quorum in a stretched cluster but it can also be used with other cluster types. The third feature is the [Health service](#) in Windows Server. The storage stack in earlier versions of Windows provided telemetry for monitoring but it was up to each performance and monitoring tool to consume that data and analyze it. In Windows Server 2016+ the service gathers disk data in S2D clusters and tracks the state of disks, restores redundancy automatically when a failed disk is replaced and so forth. It also lets you use cmdlets such as Get-StorageHealthReport and Debug-StorageSubSystem (to see faults).

```

PS C:\> Get-StorageSubSystem Cluster* | Get-StorageHealthReport
CPUUsageAverage      : 3.33 %
CapacityPhysicalPooledAvailable : 3.12 TB
CapacityPhysicalPooledTotal   : 8.15 TB
CapacityPhysicalTotal     : 8.15 TB
CapacityPhysicalUnpooled    : 0 B
CapacityVolumesAvailable   : 1.35 TB
CapacityVolumesTotal       : 1.46 TB
IOLatencyAverage          : 0 ns
IOLatencyRead             : 2.66 us
IOLatencyWrite            : 0 ns
IOPSRead                 : 0 /S
IOPSTotal                : 0 /S
IOPSSWrite               : 0 /S
IOTThroughputRead         : 0 B/S
IOTThroughputTotal        : 0 B/S
IOTThroughputWrite        : 0 B/S
MemoryAvailable          : 55.81 GB
MemoryTotal               : 128 GB

ExtendedStatus : 
ReturnValue   : 0
PSComputerName :

```

Get-StorageHealthReport for a S2D cluster

As long as any third-party software you use works with Windows Server 2016 that is the recommended version to deploy, until 2019 is released. If you use the Standard SKU you only get two virtual Windows Server VM licenses, whereas Datacenter gives you unlimited Windows Server VM licenses on that host. For more information on licensing for virtual environments check out [this free Altaro eBook](#).

The option of using Server Core for your Hyper-V hosts is recommended by Microsoft but you really need to consider your own or your team's skills with PowerShell and command line troubleshooting before selecting that installation option. Windows Admin Center (WAC) does make it easier to manage Server Core. Larger environments with very tight standardization of server hardware might be good candidates for Server Core.

Buy servers with AMD or Intel processors that support Second Level Address Translation (Intel calls it Extended Page Tables and AMD calls it Rapid Virtualization Indexing).

Windows Server 2016 brought nested virtualization to the table. This means you can run a physical host, with VMs that in turn have Hyper-V enabled and run VMs within them (and so on, you can go several layers deep). It's incredibly handy for labs and learning environments but not something you'll run your production workloads on, apart from containers which can run inside VMs instead of on the bare metal.

Hosts should only run the Hyper-V role and the only other software that should run on the hosts are management and backup agents. The debate on whether anti-virus software should be run on hosts or only in VMs isn't settled. If you do decide to put it on your hosts please follow [this list](#) to exclude the relevant files, folders and processes.

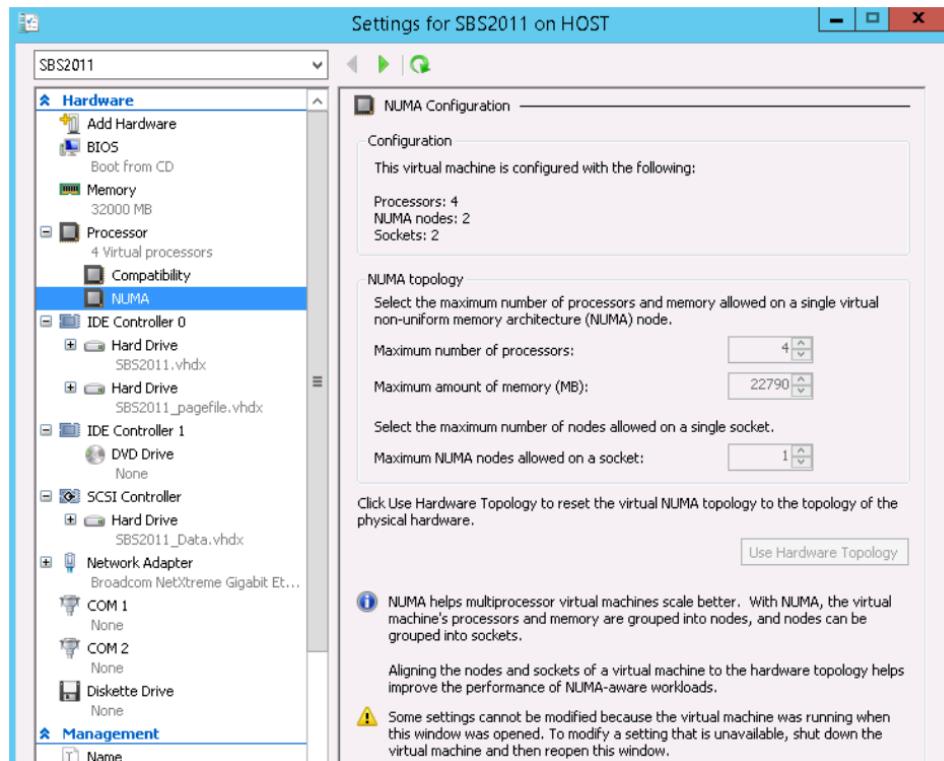
For a comprehensive checklist with details on how to design a Windows Server 2012 R2 Hyper-V environment (most tips in the article are applicable to 2016 / 2019 as well), or fix one that has been neglected, there's an excellent [article](#) here. Hosts should have RDP printer mapping disabled.

The single most important tip for hosts is to make sure your drivers are up to date! In many cases simply updating drivers leads to significant performance and stability improvements. That said, it pays to check the forums of your hardware vendor to see if anyone has had issues with a particular driver version.

If you're planning to run Windows / Hyper-V containers on your hosts, the recommendation would be to run separate hosts, or to use nested virtualization in Windows Server 2016 or later, rather than mixing containers and normal VMs on the same host.

Another performance-related issue is understanding Non-Uniform Memory Access (NUMA) for your particular hardware. A standard virtualization host today might come with two physical processors with eight cores each and 256 GB memory. All those cores do not have the same high-speed access to all the memory; the server is instead divided into NUMA nodes. An example system has four cores and 64 GB of memory in each node. If you deploy a VM that's larger than a node, the performance of that VM could be adversely impacted. Hyper-V has always had the option to allow a VM to span NUMA nodes and since Window Server 2012, the underlying NUMA topology is

also projected into VMs so that OSs and applications inside of the VM can configure themselves optimally based on the underlying NUMA topology. One other caveat here is if you have Hyper-V hosts of different memory configuration in the same cluster and the NUMA node sizes don't match between hosts you need to configure your VMs to fit into the smallest NUMA node in the cluster. Otherwise you might get great performance on one host but when the VM is Live Migrated to another host performance might be adversely impacted.



NUMA settings for a VM

STEP 2. HOST PATCHING

Keeping your hosts up to date with patches is another important aspect of performance management. Just turning on Windows Update or relying on an internal Windows Server Update Services (WSUS) or System Center Configuration Manager (SCCM) is not enough. There are several updates specific to Hyper-V that aren't rated as critical so with default settings they'll never be applied to your hosts. Find the list for Windows

Server 2008 [here](#), 2008 R2 [here](#), 2012 [here](#), 2012 R2 [here](#). Windows Server 2016/2019 don't have a corresponding list since the updates are now monthly cumulative packages. This means that if you deploy a clean Windows Server 2016 for instance, you only need to install the latest monthly cumulative update package to get all updates released since 2016.

Providing a highly performant fabric for your business VMs to run on and providing good uptime for those VMs should never compromise security.

Make sure the location where you keep virtual disks as well as checkpoints are secure with appropriate permissions.

The easiest way to keep a Hyper-V cluster up to date is using [Cluster Aware Updating](#) (CAU). This automates the process of draining VMs from one host and then installing the patches, restarting the host if required and then repeating the process on the next host. If VMM is deployed it has [built in functionality](#) for creating baselines and applying updates across all fabric servers.

HOST DRIVERS

It was stated above but it bears repeating: it's very important to update drivers in your hosts for optimal performance. This is a step that often is overlooked, after all "it is working now, don't change anything" is a well-worn mantra. But the fact is that if you check on your server manufacturers support pages you'll likely find many issues fixed in later updates for drivers and firmware.

Make sure to regularly check for software updates for network cards, RAID controllers, mainboards, HBAs and other devices on your hosts.

STEP 3. PLANNING VMs TO MINIMIZE THE RISK OF PERFORMANCE ISSUES

The most critical point here is ensuring that your Integration Services (sometimes called Integration Components) are up to date. Early on in Hyper-V's life this was done manually from the host (by inserting the Integration Services ISO file into each VM) but now all supported versions of Windows Server and client get updates to Integration Services through Windows Update (or WSUS / Configuration Manager). For Linux VMs, the Integration Services are updated when you update the kernel.

Use the latest versions of Windows and Linux in your VMs. Successive generations of OSs are better at running as VMs.

When assigning processor resources to guests, you will see the terms “logical processor” (LP) and “virtual processor” (VP). A LP is the operating system’s way of showing a single pipeline for processing which only operates on one thread at a time. On AMD CPUs, and on Intel CPUs with Hyper-threading disabled, each LP is a physical core. There is no direct relationship between an LP and a VP. Hyper-V doesn’t have the issue around gang scheduling that VMWare has and it’s safe to assign many VPs to a VM. If a VM doesn’t use the CPU resources, other VMs can use them. With Hyper-threading enabled each core counts as two LPs. Note that the processing on this second LP only provides about 25% more performance because execution of instructions is tightly bound to the primary core.

For VMs that are going to be Active Directory Domain Controllers, turn off the Hyper-V Time Synchronization guest service and make sure that the PDC FSMO role in the first domain in your forest is synchronizing its clock with an authoritative NTP server.

Windows Server 2012 R2 Hyper-V introduced [generation 2 VMs](#). These VMs have no emulated hardware and install and boot faster (but do not run faster). If you’re using recent Linux distros or Windows Server 2012 or later as the OS in your VMs, consider using generation 2 VMs. Windows VMs benefit from virtual Secure Boot in generation 2 and in Windows Server 2016 Hyper-V, so can Linux VMs.

Another benefit in 2012 R2 and later is Enhanced Session mode, giving you remote access to VMs even before they have an OS installed; great copy and paste functionality between host and VMs, as well as USB device redirection from a remote desktop into a VM.

STEP 4. PLANNING STORAGE TO MINIMIZE THE RISK OF PERFORMANCE ISSUES

As stated before, by far the most common mistake in Hyper-V deployments is not providing enough IO performance. The scariest is where capacity is mistaken for IOPS. For instance, “each server takes up 50 GB, so that means we should be able to fit in 70 servers on this new 4 TB hard drive”. The single most important thing you need to consider when planning a Hyper-V deployment or expansion is to **account for IOPS and throughput required, not capacity**.

Before virtualization, when each OS lived on its own physical server, how did we plan storage? For example – “a couple of SAS disks in a mirror for the OS, better make them 15,000 RPM disks”, and, “For the data – a RAID 5 set with six disks to get the required performance”. Nothing in that equation has changed when you move to a virtualized server. It still needs a certain minimum number of IOPS, even if it’s now delivered from a virtual disk on a SAN array or an S2D file share. Also, plan for future growth. Many systems were probably planned with adequate capacity a few years ago but since then more hosts have been hooked up to the same storage with more VMs, resulting in IOPS starvation.

The release of Windows Server 2016 has brought a new option for Hyper-V storage; [**Storage Spaces Direct \(S2D\)**](#), available in the Datacenter SKU only. This uses the foundations from Storage Spaces in 2012 but instead of disk trays attached to multiple servers, it pools the locally attached storage in each host together. This brings two main benefits, less hardware to buy (no external disk boxes) and the ability to use new storage technologies as shared storage. You do need at least 10 Gb/s networking between the nodes (data written to one virtual hard disk must be copied to two other nodes for redundancy). S2D supports SATA/SAS-connected hard drives and SSD drives. It also supports NVMe drives, essentially SSD drives connected direct to the PCI Express

bus instead of connected through SAS/SATA. Finally (in Windows Server 2019 only), S2D also supports persistent memory – battery-backed RAM sticks for storage, also known as [Storage Class Memory / NVDIMM-N](#). Most people know that HDDs are slow and SSDs are faster but what you might not know is that NVMe drives are about three times faster than SSDs (because of the faster data transfer bus) and use about half the CPU of SSDs. Persistent memory is (as you can probably guess) insanely fast with very low latency.

You can [combine HDDs and SSDs, HDDs and NVMe](#), or if your budget allows, SSDs for capacity storage and NVMe for cache. Note that the faster storage medium will be used as a distributed read / write cache and not for data storage. Make sure you select [enterprise-grade SSDs / NVMe drives](#) that have the write endurance required for a production deployment.

In Windows Server 2016 the recommended redundancy setting is three-way mirroring which means that if you have 30 TB of total HDD space and 4 TB of total SSD/NVMe space, you'll have 10 TB of usable storage. [Here's a calculator](#) to help you work out your usable storage. On all flash configurations of S2D, Microsoft has demonstrated 6 Million IOPS. The minimum number of nodes is 2 (which only offers two-way mirroring but may be suitable for small branch offices) and the maximum is 16.

Other [options for redundancy in S2D](#) are single or double parity (space efficient but only good for archival storage) and Multi Resilient Volumes (MRV) which combines mirroring and parity on the same volume. Its performance in 2016 is poor but 2019 promises to make MRV a viable option. Rather than building your own cluster, most major hardware vendors have pre-validated configurations under the name [Windows Server Software Defined](#) (WSSD). The recommended file system for Hyper-V deployments in 2016 is ReFS, rather than NTFS. In Windows Server 2019 you can also use deduplication for Hyper-V virtual disk storage on ReFS, in 2016 this is only available on NTFS.

S2D has fast become the best deployment flavor for Hyper-V.

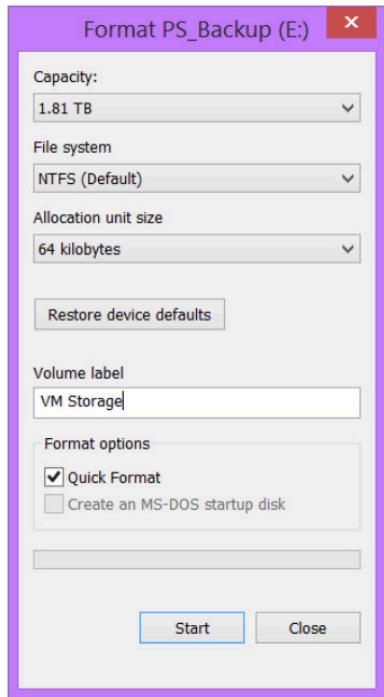
You can either deploy separate storage clusters that share out their storage to Hyper-V hosts via SMB shares or the Hyper-V hosts themselves can also be storage hosts, referred to as Hyper Converged Infrastructure (HCI). S2D is easier to manage than Storage Spaces with most tasks automated.

Profiling the IOPS requirements of your workloads is an important part of planning for new deployments. Use Performance Monitor to profile them to see what IOPS requirements they have. You can also use [this script](#) which tests your storage to give you performance information. It relies on the slightly older SQLIO tool; there's also the newer [DiskSPD](#) which can be used to give your storage a good workout.

The type of shared storage to use for your Hyper-V clusters is also a planning consideration with four main options. For Windows Server 2016 or later, use S2D or an existing SAN. For earlier OS versions, if you already have an iSCSI SAN, it'll work well with Hyper-V, as long as it has the needed IO (not just storage capacity) capability for your VMs. Fibre Channel SANs will also work well with Hyper-V, and you can use virtual Fibre Channel SAN to connect VMs directly to shared storage.

Hyper-V hosts connect to shared storage using ordinary SMB 3.0 file shares for easy management while the S2D servers do the “smarts” that a SAN would do, including caching reads and writes on the faster storage and de-staging writes to the slower tier in a synchronous fashion, which gives you excellent performance from your HDDs. S2D can be more cost effective than SANs in many scenarios as well as being easier to manage.

Starting in Windows Server 2012 R2, you can assign a minimum and maximum storage IOPS (counted in 8KB increments) to a virtual disk to ensure that a single, hungry VM doesn't starve other VMs of storage performance. Note that the maximum will always be enforced, but if the SAN / S2D can't deliver the minimum storage IO you have specified to all VMs, there's nothing the hosts can do about it. In Windows Server 2016 or 2019 we have a more advanced centralized Storage Quality of Service engine that can be controlled through policies, assigning IOPS limits to individual VMs or groups of VMs (see above).



Changing to 64 KB allocation units

Format your disks with the recommended 64 KB allocation unit size.

STEP 5. PLANNING NETWORKING TO MINIMIZE THE RISK OF PERFORMANCE ISSUES

Not so long ago your aim for networking was to have as many 1 Gbps network connections in each host as you could, and then dedicate them to [different types of traffic](#) such as Live Migration, backup, storage, client connections, cluster heartbeat etc.

Today, most typical Hyper-V hosts for medium to large enterprises come with two (or more) 10 Gbps interfaces. This often calls for a converged infrastructure where software QoS divides up the bandwidth for different traffic types.

Since Windows Server 2012, a consideration in larger implementations is Remote Direct Memory Access (RDMA) NICs for high speed networking, especially if you're going to deploy S2D. It comes in [three flavors](#): iWarp, Infiniband and RoCE. RDMA gives you high

speed (40, 56 and 100 Gbps) networking, with near zero CPU overhead. If your shared storage for your cluster is SOFS (see Planning for Storage) it's recommended to use RDMA to connect the Hyper-V hosts to the SOFS hosts when possible and the same goes for S2D clusters. In Windows Server 2012 R2 or later you can also use RDMA for Live Migration traffic (see below) but these need to be on two separate networks, increasing cost. Windows Server 2016 allows you to merge both types of traffic on the same physical RDMA NICs.

For Live Migration traffic, in particular Windows Server 2012 R2 and later offer more choices with the default being compression. In this mode Hyper-V trades some CPU cycles (and most Hyper-V hosts that I've seen have lots of spare processor capacity) for decreased network load by compressing the data stream on the source server and then decompressing on the destination server. Compression is also pretty smart, and it continually monitors the hosts, if CPU load increases, compression will back off.

Compression can lead to 50 % reduction in times for Live Migrations
(your mileage may vary).

Another option is RDMA / SMB Direct if your network cards offers the capability, this will make your Live Migrations lightning fast.

For applications in your VMs where clients need to have very low latency access, an option to consider in your planning is Single Root I/O Virtualization (SR-IOV) NICs. As long as your server motherboards and BIOS supports it SR-IOV will project virtual copies of itself into VMs, providing close to bare-metal performance. Using SR-IOV NICs doesn't stop your VM from being Live Migrated, if the destination host doesn't have SR-IOV NICs (or the ones that are there have no more virtual functions to spare) the Hyper-V virtual switch will transition over to a standard synthetic adapter with minimal impact.

On top of all this you can use NIC teaming, either switch dependent or switch independent flavors, to provide increased bandwidth as well as failover in case of link failure. Teaming can also be done both at the host level and inside of individual VMs.

Because there are so many different business situations and requirements it's difficult to provide general network recommendations beyond the above. It is, however, important to [familiarize yourself](#) with the virtual switch and the different technologies that can speed up VM network throughput. Altaro has several good blog posts explaining [Hyper-V networking](#) in detail, including:

- [The Complete Guide](#),
- [The Really Simple Guide](#),
- [How many cluster networks](#),
- [Architect networks for a Hyper-V Cluster](#).

The virtual switches on each Hyper-V host in the cluster need to have the same name and settings for Live Migration to work. In a larger environment you should use the Logical switch in Virtual Machine Manager to push out a centrally controlled switch to all cluster hosts.

Centrally defining all your virtual switch settings negates the need to configure the switch on each host individually.

While Hyper-V has had [Software Defined Networking](#) (SDN) technology since 2012, it was rebuilt in 2016, with the NVGRE protocol giving way to the more widely adapted VXLAN. Another thing that changed is that the reliance on System Center Virtual Machine Manager (VMM) was removed by building all required components into Hyper-V. This brings [network virtualization](#), the ability to easily create isolated networks on top of your physical network to smaller environments. And the benefits, particularly for security, are tangible. Microsoft provides [scripts and documentation](#) for your deployments, note that if you do have VMM deployed you'll be able to use its UI to configure SDN networks, whereas without it, PowerShell is your only option (although WAC is adding more support for SDN each month). There's an System Center Operations Manager [management pack for SDN](#) and [here's a good list of troubleshooting tools](#). WAC also lets you [manage SDN-enabled clusters](#), including enabling a new feature in

Windows Server 2019 – the ability to automatically encrypt virtual network traffic in an SDN deployment.

If you're deploying container hosts, take particular care to [get the networking right](#), especially if you're [using the SDN stack](#).

STEP 6. PLANNING MANAGEMENT TO MINIMIZE THE RISK OF PERFORMANCE ISSUES

In small environments you should be fine with WAC, and maybe the occasional trip to Hyper-V Manager and Failover Cluster Manager, along with PowerShell / scripting to keep things humming along. Unlike the MMC tools, WAC is being continually developed, with a new version coming out every month. And third-party extensions will continue to add value – I can't wait for the major server vendors to integrate their hardware management into WAC – truly providing a single pane of glass for management. It's important that you include [PowerShell management](#) and scripting into your management toolbelt.

Automating deployments, [network configuration](#), performance and [health monitoring](#) and configuring settings across all hosts or groups of VMs makes you a more efficient Hyper-V administrator.

For larger deployments, System Center Virtual Machine Manager (VMM) is your friend. It does a lot more than just manage VMs; deploys the OS and Hyper-V / SOFS / S2D roles to brand new, bare-metal servers and interfaces with your SANs, S2D clusters and SOFS storage for automatic provisioning.

It manages both Hyper-V and VMware environments, manages Top of Rack switches through OMI and can create private clouds which abstracts the underlying fabric and presents quota-based VM resources to application owners. VMM uses templates to deploy individual VMs as well as multi-tier applications with potentially many VMs as services which can be managed as a unit.

The screenshot shows the System Center Virtual Machine Manager (VMM) console interface. The ribbon at the top has tabs for Server Tools, Virtual Machine Tools (which is selected), Administrator - SCVMMvNext.cloud.local - Virtual Machine Manager (Evaluation version), and Help. Below the ribbon is a toolbar with icons for Create, Shut Down, Power On, Power Off, Reset, Pause, Resume, Save State, Discard Saved State, Migrate Storage, Migrate Virtual Machine, Create Checkpoint, Manage Checkpoints, Connect or View Window, Delete, and Properties. The main pane displays a list of VMs (6) in a table format. The table columns include Name, Status, V., Avai..., Host, Cloud, Job Status, O., U., CPU..., Serv..., and Op. The VMs listed are: Win10ServerStorage2 (Running, hv3), Win10ServerRDS (Running, hv3), Win10Server (Running, hv3), Win10ServerStorage1 (Unsupported VM Configuration, hv3, Failed), SCVMMvNext (Running, hv3), and SCCM2012R2 (Running, hv3). The SCVMMvNext VM is highlighted. The left sidebar has a tree view with nodes for Tenants, Clouds, VM Networks, Storage, All Hosts, and Win10Cluster, with hv3 selected. At the bottom, there are sections for Virtual machine information (Status: Running, Owner: [empty], Processors: 1, Memory: 2.00 GB), Logical networks (cloud), Recent job (Name: No recent job, Job status: [empty]), Go to related object (Host: hv3.cloud.local), Storage (1 disks) (Total storage (127.00 GB)), and Daily performance (CPU) (Average).

The VMM console

To keep track of performance, the right System Center family member is Operations Manager (OM). Extensible through Management Packs, OM will collate events and performance data from your host hardware, storage, networking fabric as well as Hyper-V itself. Microsoft offers a Management Pack (MP) for Hyper-V.

TAKE AWAY ACTION ITEMS:

1. Monitor the performance of your hosts – use PAL and WAC or ALA.
2. Keep a close eye on the performance of your fabric, in particular storage.
3. Use performance data to understand your environment and VMs.
4. Install and configure WAC and use it to manage your infrastructure and VMs.
5. Plan your fabric environment to make sure you minimize the risk of performance problems.
6. Keep your hosts firmware, drivers and OS patches up to date.

ABOUT THE AUTHOR: PAUL SCHNACKENBURG



Paul Schnackenburg started in IT when DOS and 286 processors were the cutting edge. He works part time as an IT teacher at a Microsoft IT Academy. He also runs Expert IT Solutions, a small business IT consultancy on the Sunshine Coast, Australia. Paul writes in-depth technical articles, focused on Hyper-V, System Center, private and hybrid cloud and Office 365 and Azure public cloud technologies. He has MCSE, MCSA, MCT certifications. He can be reached at paul@expertitsolutions.com.au, follow his blog at TellITasITis, <http://tellitasitis.com.au/>.

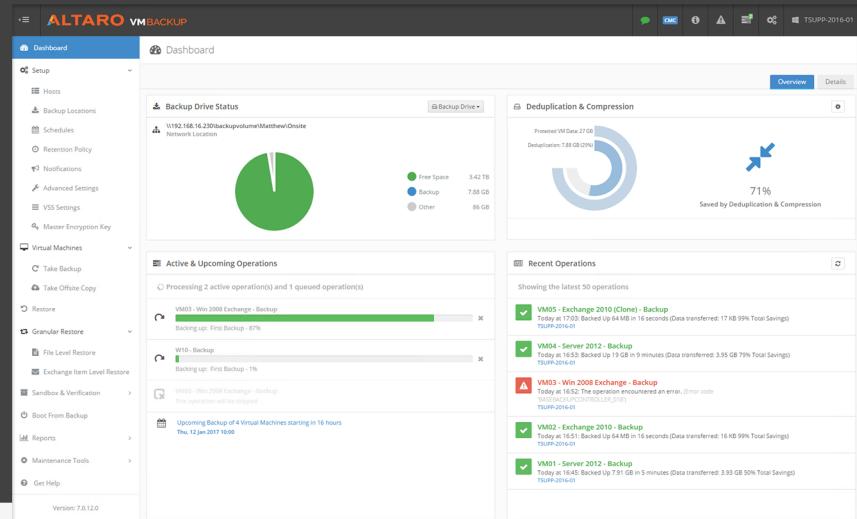
Altaro VM Backup - Trusted by over 40,000 SMBs

Altaro VM Backup for VMware & Hyper-V is hassle-free and affordable virtual machine backup solution. Start your free trial today!

- ✓ Hassle-free and effective
- ✓ Unbeatable Value
- ✓ Outstanding Support

The free trial enables you to backup unlimited VMs for 30 days. Afterwards, you can continue to use the free version to backup 2 VMs per host, forever – our way of assisting micro businesses.

**Download your
30-day trial**

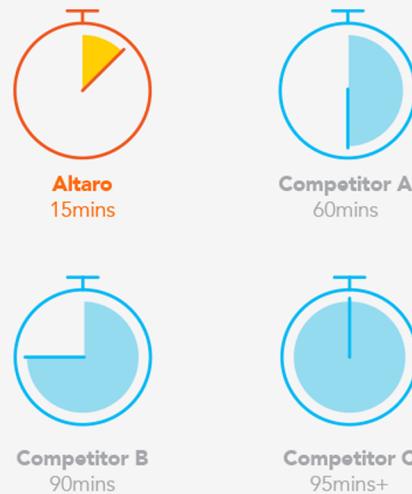


Up and running quickly, without the need for complex configurations!

With Altaro VM Backup, you can install and run your first virtual machine (VM) backup in less than 15 minutes. Get up and running quickly, without the need for complex configurations or software dependencies.

Altaro VM Backup is designed to give you the power you need, without the hassle and steep learning curve.

- Easy to use, intuitive UI - making it easy to implement a rock solid backup strategy
- Managing and configuring backup/restore jobs across multiple hosts has never been simpler
- Full control & scalability - Monitor and manage all your Hyper-V and VMware hosts from a single console



Virtual machine backup software packed with powerful features for VMware and Hyper-V.

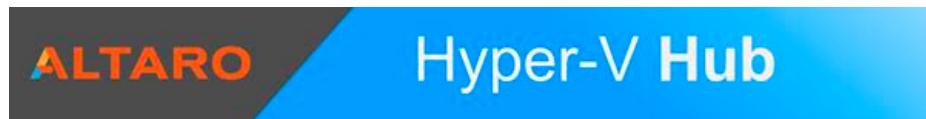
View features

ABOUT ALTARO

[Altaro Software](#) is a fast-growing developer of easy-to-use backup solutions which backs up and restores both Hyper-V and VMware-based virtual machines, built specifically for MSPs and SMBs customers with up to 50 host servers. Altaro take pride in their software and their excellent level of personal customer service and support, and it shows. Founded in 2009, Altaro already services over 40,000 satisfied customers worldwide and are a Gold Microsoft Partner for Application Development and Technology Alliance VMware Partner.

FOLLOW ALTARO

Like this eBook? **There's more!**



[Subscribe to our Hyper-V blog](#) and receive best practices, tips, optimization guides and more!:

<https://www.altaro.com/hyper-v-backup>

Follow Altaro at:



 **SHARE THIS RESOURCE!**

Liked the eBook? Share it now on:



PUBLISHED BY
Altaro Software
<http://www.altaro.com>

Copyright © 2018 by Altaro Software
All rights reserved. No part of this book may be reproduced or transmitted in any form
or by any means without the prior written permission of the publisher or authors.

WARNING AND DISCLAIMER

Every effort has been made to make this book as complete and as accurate as possible,
but no warranty or fitness is implied. The information provided is on an “as is” basis.
The authors and the publisher shall have neither liability nor responsibility to any
person or entity with respect to any loss or damages arising from the information con-
tained in this book.

FEEDBACK INFORMATION

We'd like to hear from you! If you have any comments about how we could
improve the quality of this book, please don't hesitate to contact us by visiting
www.altaro.com or sending an email to our Customer Service representative
Sam Perry: sam@altarosoftware.com