Pexip Infinity

Server Design Guide

Software Version 34

Document Version 34.a

March 2024

]pexip[

Contents

Introduction

This document describes the recommended specifications and deployment for servers hosting the Pexip Infinity platform, which apply to both on-premises hardware and cloud deployments. It starts with a Summary of recommendations and some Example Conferencing Node server configurations, which are supplemented by further details and explanations in the following Appendices:

- Appendix 1: Detailed server hardware requirements provides a more detailed breakdown of the minimum and recommended hardware requirements for servers hosting the Management Node and Conferencing Nodes respectively.
- Appendix 2: Achieving high density deployments with NUMA provides details on NUMA architecture and how this impacts server architecture and overall performance of Conferencing Nodes.
- Appendix 3: VMware NUMA affinity and hyperthreading is for administrators with advanced VMware knowledge. It explains how to experiment with VMware NUMA affinity and make use of hyperthreading for Pexip Infinity Conferencing Node VMs, in order to achieve up to 50% additional capacity.
- Appendix 4: Hyper-V NUMA affinity and hyperthreading is for administrators with advanced Hyper-V knowledge. It explains how to experiment with Hyper-V NUMA affinity and make use of hyperthreading for Pexip Infinity Conferencing Node VMs, in order to achieve up to 50% additional capacity.

Summary of recommendations

This section summarizes the terminology, recommended specifications and deployment guidelines for servers hosting the Pexip Infinity platform. These apply to both on-premises hardware and cloud deployments.

Terminology

The table below provides descriptions for the terms used in this guide, in the context of a Pexip Infinity deployment.

Term	Description
Processor	The hardware within a computer that carries out the basic computing functions. It can consist of multiple cores.
Core	One single physical processing unit. An Intel Xeon Scalable processor typically has between 8 and 32 cores, although both larger and smaller variants are available.
Socket	The socket on the motherboard where one processor is installed.
RAM	The hardware that stores data which is accessed by the processor cores while executing programs. RAM is supplied in DIMMs (Dual In-line Memory Modules).
Channel	A memory channel uses a dedicated interface to the processor, and can typically support up to 2 DIMMs. An Intel Xeon Scalable series processor has 6-8 memory channels, older processors have fewer channels.
Virtual CPU (vCPU)	The VM's understanding of how many CPU cores it requires. Each vCPU appears as a single CPU core to the guest operating system. When configuring a Conferencing Node, you are asked to enter the number of virtual CPUs to assign to it. We recommend no more than one virtual CPU per physical core, unless you are making use of CPUs that support Hyper-Threading.
NUMA node	The combination of a processor (consisting of one or more cores) and its attached memory.

Management Node

Recommended host server specifications

- minimum 4 vCPUs* (most modern processors will suffice)
- minimum 4 GB RAM* (minimum 1 GB RAM for each Management Node vCPU)
- 100 GB SSD storage
- The Pexip Infinity VMs are delivered as VM images (.ova etc.) to be run directly on the hypervisor. No OS should be installed.

* Sufficient for typical deployments of **up to 30** Conferencing Nodes. For deployments with **more than 30** Conferencing Nodes, you will need to increase the number of cores and the amount of RAM on the Management Node. Please contact your Pexip authorized support representative or your Pexip Solution Architect for guidance on Management Node sizing specific to your environment.

Management Node performance considerations

There are a number of factors that can impact the performance of the Management Node, and these should be taken into consideration alongside the recommended specifications described above when determining the size of the Management Node in your deployment.

- Individual cores can vary significantly in capability.
- Large numbers of roughly simultaneous join and leave events will increase the load on the Management Node, when compared to the same number of events spread over a broader time range.
- Different Pexip Infinity features will place different loads on the Management Node, so the overall load will be impacted by the features you have implemented and the frequency with which they are used.
- Heavy or frequent use of the management API will significantly increase the load on the Management Node.
- Multiple Live view sessions will increase the load on the Management Node.

Transcoding Conferencing Nodes

Below are our general recommendations for Transcoding Conferencing Node servers. For some specific examples, see Example Conferencing Node server configurations.

Recommended host server specifications

- We recommend 3rd- or 4th-generation Intel Xeon Scalable Series processors (Ice Lake / Sapphire Rapids) Gold 63xx/64xx for Transcoding Conferencing Nodes.
  - Earlier Intel Xeon Scalable Series processors and Intel Xeon E5/E7-v3 and -v4 series process are also supported where newer hardware is not available. Machines based on these architectures will work well for Management and Proxying Edge nodes, we recommend prioritizing the newest hardware for transcoding nodes.
  - Other x86-64 processors from Intel and AMD that support at least the AVX instruction set can be used but are not recommended. Some features are only available on underlying hardware that supports at least the AVX2 instruction set.
- 2.6 GHz (or faster) base clock speed if using Hyper-Threading on 3rd-generation Intel Xeon Scalable Series (Ice Lake) processors or newer.
  - 2.8 GHz+ for older Intel Xeon processors where Hyper-Threading is in use
  - 2.3 GHz+ where Hyper-Threading is not in use
- Minimum 4 vCPU per node
- Maximum 48 vCPU per node, i.e. 24 cores if using Hyper-Threading
  - Higher core-counts are possible on fast processors: up to 56 vCPU has been tested successfully
  - Slow (under 2.3GHz) processors may require lower core counts
- 1 GB RAM for each vCPU that is allocated to the Conferencing Node

    - Each memory channel should be populated:
        - Intel Xeon Scalable Series processors have 6-8 channels
        - Intel Xeon E5/E7 series processors have 4 channels
        - AMD 3rd-Gen EYPC (Rome/Milan) processors have 8 memory channels
        - AMD 4th-Gen EPYC (Genoa/Bergamo/Siena) processors have 12 memory channels
    - For most bus speeds, 16 GB is the smallest DIMM that is available, so it is often necessary to install more than 1 GB of RAM per vCPU to populate all the memory channels.
- CPU and RAM must be dedicated
- Populate all memory channels
    - For small (up to 12vCPU) nodes, populating 4 memory channels will be sufficient provided there is nothing else running on the socket
- Storage:
    - 500 GB total per server (to allow for snapshots etc.), including:
    - 50 GB minimum per Conferencing Node
    - SSD recommended
    - RAID 1 mirrored storage (recommended)
- Hypervisors: VMware ESXi 6.7, 7.0 and 8.0; KVM
- The Pexip Infinity VMs are delivered as VM images (.ova etc.) to be run directly on the hypervisor. No OS should be installed.

Proxying Edge Nodes

The servers hosting Proxying Edge Nodes do not require as high a specification as those servers hosting Transcoding Conferencing Nodes because they do not process any media.

Recommended host server specifications

- Any x84-64 processor that supports the AVX instruction set or newer (AVX2, AVX512)
- 4 vCPU per node
- 4 GB RAM per node
    - For large or busy systems 8 vCPU / 8 GB RAM can be used, but this must not be exceeded
    - We recommend scaling with multiple Proxying Edge Nodes for redundancy
- CPU and RAM must be dedicated
- Storage:
    - 500 GB total per server (to allow for snapshots etc.), including:
    - 50 GB minimum per Conferencing Node
    - SSD recommended
    - RAID 1 mirrored storage (recommended)
- Hypervisors: VMware ESXi 6.7, 7.0 and 8.0; KVM
- The Pexip Infinity VMs are delivered as VM images (.ova etc.) to be run directly on the hypervisor. No OS should be installed.

Special considerations for AMD EPYC systems

We recommend using 3rd-generation Intel Xeon Scalable Series processors (Ice Lake) and newer. AMD EPYC processors are supported, but the performance per core is notably lower than contemporary Intel parts. Where AMD EPYC processors are used:

- We recommend using NPS4 and up to 32vcpu VMs i.e. four VMs per socket, and that each VM is pinned to one NUMA node.
- For optimal performance, populate 8 DIMMs for 1 DIMMs per Channel (DPC) configuration.

CPU microcode updates and performance

Microcode updates for Intel and AMD CPUs can have a significant negative impact on transcoding performance. Customers hosting Pexip Infinity in their own trusted environment might choose to not apply these updates; if Pexip Infinity is the only application running on the hardware then security risk is minimal.

Low-level configuration

Hyper-Threading

- Hyper-Threading (also referred to as Hyper-Threading Technology), if supported, should always be left enabled by default.
- When Hyper-Threading is in use, we recommend that Conferencing Nodes are NUMA pinned to their sockets to avoid memory access bottlenecks.

Sub-NUMA clustering

Sub-NUMA Clustering [SNC] should be turned off unless you are using the ultra-high density deployment model or you have been specifically recommended otherwise by your Pexip authorized support representative.

BIOS performance settings

Ensure all BIOS settings pertaining to power saving are set to maximize performance rather than preserve energy. (Setting these to an energy-preserving or balanced mode may impact transcoding capacity, thus reducing the total number of HD calls that can be provided.) The actual settings depend on the hardware vendor; some examples are given below:

Typical HP settings

- HP Power Profile: Maximum Performance
- Power Regulator modes: HP Static High Performance mode
- Energy/Performance Bias: Maximum Performance
- Memory Power Savings Mode: Maximum Performance

Typical Dell settings

- System Profile: Performance Optimized

Typical Cisco UCS B-Series settings

- System BIOS Profile (Processor Configuration) - CPU Performance: Enterprise
- System BIOS Profile (Processor Configuration) - Energy Performance: Performance
- VMware configuration: Active Policy: Balanced

Network

- Although the Conferencing Node server will normally not use more than 1-2 Mbps per video call, we recommend 1 Gbps network interface cards or switches to ensure free flow of traffic between Pexip Infinity nodes in the same datacenter. We do not recommend 100 Mbps NIC.
- Redundancy: for hypervisors that support NIC Teaming (including VMware), you can configure two network interfaces for redundancy, connected to redundant switches (if this is available in your datacenter).

Disk

- Although Pexip Infinity will work with SAS drives, we strongly recommend SSDs for both the Management Node and Conferencing Nodes. General VM processes (such as snapshots and backups) and platform upgrades will be faster with SSDs.
- Management Node and Conferencing Node disks should be Thick Provisioned.
- Pexip Infinity can absorb and recover relatively gracefully from short bursts of I/O latency but sustained latency will create problems.

- The Management Node requires a minimum of 800 IOPs (but we recommend providing more wherever possible).
- A Conferencing Node requires a minimum of 250 IOPs (but we recommend providing more wherever possible).
- Deployment on SAN/NAS storage is possible, but local SSD is preferred. Interruption to disk access during software upgrades or machine startup can lead to failures.
- Redundancy: when using our recommended RAID 1 mirroring for disk redundancy, remember to use a RAID controller supported by VMware or your preferred hypervisor. The RAID controller must have an enabled cache. Most vendors can advise which of the RAID controllers they provide are appropriate for your hypervisors.

Power

- Sufficient power to drive the CPUs. The server manufacturer will typically provide guidance on this.
- Redundancy: Dual PSUs.

Example Conferencing Node server configurations

This section provides some example server configurations for Transcoding Conferencing Nodes, with an estimate of the capacity (in terms of how many HD connections — sometimes referred to as "ports") you can expect to achieve with each option. These figures assume that all memory channels are populated, that NUMA affinity and hyperthreading has been enabled, and that all other actions in the best practices checklist have been completed.

This topic covers:

Issues to consider when specifying hardware

When choosing hardware, you should consider carefully what you want to optimize for. The best choice varies depending on how you intend to use your Pexip Infinity deployment, your budget and the costs of hosting your hardware.

Management and Proxying Edge Nodes

Management and proxying nodes rarely require a full socket. In many cases, the Management Node and some proxying node nodes share a socket, sometimes with a small transcoding node.

Because these nodes do not perform any media processing, you can use older or less capable hardware for these nodes. Please note that proxying nodes do require at least the AVX instruction set. Always prioritize the newest and most capable hardware for transcoding nodes.

Small deployments

Where the requirement is for a fixed number of ports overall or within a particular physical location and this requirement is for up to 90HD ports, transcoding for that deployment or location can easily be provided on a single socket.

Large conferences

Large conferences work best on large nodes. Any one conference can span only three nodes in a given logical location. The conference takes one backplane on each node that it uses, so minimizing the number of nodes that a conference spans reduces this overhead.

In this scenario we would normally recommend building the largest individual nodes possible.

Many smaller conferences or gateway calls

When a high volume of small conferences or gateway calls are expected, optimizing purely for individual node capacity is less important. There are benefits of having fewer larger nodes over more smaller ones as there is less likelihood of a conference being fragmented across two or more nodes thus saving on backplane overheads.

In this scenario the number of ports per socket is probably more important than the number of ports per node. Where rack space is at a premium you may want to consider Achieving ultra-high density with Sub-NUMA Clustering.

Pexip recommendations

Best all-rounder

We recommend 3rd- or 4th-generation Intel Xeon Scalable Series processors. In general, the Gold line represents the best value in terms of the number of ports provided for a given hardware spend. We recommend the Xeon Gold 6342 for its 2.8GHz base clock

speed and 24 physical cores. When optimally deployed it can offer up to 97-100HD per socket or up to 195HD in a 1U 2-socket server.

The Xeon Gold 6348 and 6354 parts are slightly less capable, but still represent good options if the 6342 is not available. We have no data for the Xeon Gold 6442Y, but expect its performance to be similar.

Less powerful hardware is available, but as a proportion of the server cost the savings are not large for a significant reduction in capability. Where possible you should over-specify your hardware because forthcoming features of the Infinity platform may require additional processing power.

For ultra-high density deployments

If capacity per rack unit is your main requirement, the Intel Xeon Platinum 8458P is worth considering. We have not yet tested this, but with 2.7GHz and 44 physical cores we would expect 350HD or more from a 1U 2-socket server.

For recyclers

Sometimes new hardware is not an option. If you need to use existing hardware, we recommend a 6248R machine. When new, this was our preferred hardware option and it is being used successfully to run Pexip Infinity by a variety of organizations globally. Ensure all 6 memory channels are populated on each socket; you should achieve 87-95HD per socket or around 180HD in a 1U 2-socket server.

Recommended server sizes

For the Pexip Infinity platform, the following server configurations provide maximum performance for cost:

	Capacity (no. of connections)		
	Up to 195HD	**350HD+ (estimate)**	**Up to 180 HD**
Cores / Generation	2x24-core Ice Lake Launched: Q2 2021	2x44-core Sapphire Rapids Launched: Q1 2023	2x24-core Cascade Lake Launched: Q1 2020
CPU	2 x Intel Xeon Gold 6342 • 10nm lithography • 24 core • 2.8 GHz • 36 MB cache	2 x Intel Xeon Platinum 8458P • Intel 7 lithography • 44 core • 2.7 GHz • 82.5 MB cache	2 x Intel Xeon Gold 6248R • 14nm lithography • 24 core • 3.0 GHz • 35.75 MB cache
RAM	16 x 16 GB (16 x 8 GB if available) 8 DIMMs per socket		12 x 8 GB / 12 x 16 GB 6 DIMMs per socket
Network	1 Gbps NIC (we recommend dual NIC for redundancy)		
Storage	• 500 GB total per server (to allow for snapshots etc.), including: • 50 GB minimum per Conferencing Node • SSD recommended • RAID 1 mirrored storage (recommended)		
Power	We recommend redundant power		

Other processors

We are unable to test all processors on the market. We do maintain some data on real-world usage, but this is not always reliable as we have no way of telling if the deployment has been performed according to our best practices. If you have a particular processor in mind and would like an estimate of its capability, please contact your Pexip authorized support representative or your Pexip Solutions Architect.

Appendix 1: Detailed server hardware requirements

Host server hardware requirements

The following table lists the recommended hardware requirements for the Management Node and Conferencing Node (Proxying Edge Nodes and Transcoding Conferencing Nodes) host servers.

	Management Node	Transcoding Conferencing Node	Proxying Edge Node
Server manufacturer		Any	
Processor make (see also Performance considerations)	Any	We recommend 3rd- or 4th-generation Intel Xeon Scalable Series processors (Ice Lake / Sapphire Rapids) Gold 63xx/64xx for Transcoding Conferencing Nodes. • Earlier Intel Xeon Scalable Series processors and Intel Xeon E5/E7-v3 and -v4 series process are also supported where newer hardware is not available. Machines based on these architectures will work well for Management and Proxying Edge nodes, we recommend prioritizing the newest hardware for transcoding nodes. • Other x86-64 processors from Intel and AMD that support at least the AVX instruction set can be used but are not recommended. Some features are only available on underlying hardware that supports at least the AVX2 instruction set.	Any x86-64 processor which supports at least the AVX instruction set. Most Intel Xeon Scalable Series and Xeon E5/E7-v3 and -v4 processors are suitable. If a mixture of older and newer hardware is available, we recommend using the older or less capable hardware for proxying nodes and the newest or most powerful for transcoding nodes.
Processor instruction set	Any	AVX2 or AVX512 (AVX is also supported)	AVX
Processor architecture		x86-64	
Processor speed	2.0 GHz	2.6 GHz (or faster) base clock speed if using Hyper-Threading on 3rd-generation Intel Xeon Scalable Series (Ice Lake) processors or newer. • 2.8 GHz+ for older Intel Xeon processors where Hyper-Threading is in use • 2.3 GHz+ where Hyper-Threading is not in use	2.0 GHz
No. of vCPUs *	Minimum 4†	Minimum 4 vCPU per node Maximum 48 vCPU per node, i.e. 24 cores if using Hyper-Threading • Higher core-counts are possible on fast processors: up to 56 vCPU has been tested successfully • Slow (under 2.3GHz) processors may require lower core counts	Minimum 4 vCPU per node Maximum 8 vCPU per node
Processor cache	no minimum	20 MB or greater	no minimum

	Management Node	Transcoding Conferencing Node	Proxying Edge Node
Total RAM *	Minimum 4 GB† (minimum 1 GB RAM for each Management Node vCPU)	1 GB RAM per vCPU, so either: • 1 GB RAM per physical core (if deploying 1 vCPU per core), or • 2 GB RAM per physical core (if using Hyper-Threading and NUMA affinity to deploy 2 vCPUs per core).	1GB RAM per vCPU
RAM makeup	Any	All channels must be populated with a DIMM, see Memory configuration below. Intel Xeon Scalable series processors support 6 DIMMs per socket and older Xeon E5 series processors support 4 DIMMs per socket.	Any
Hardware allocation	The host server must not be over-committed (also referred to as over-subscribing or over-allocation) in terms of either RAM or CPU. In other words, the Management Node and Conferencing Nodes each must have dedicated access to their own RAM and CPU cores.		
Storage space required	100 GB SSD	• 500 GB total per server (to allow for snapshots etc.), including: • 50 GB minimum per Conferencing Node • SSD recommended • RAID 1 mirrored storage (recommended)	
	Although Pexip Infinity will work with SAS drives, we strongly recommend SSDs for both the Management Node and Conferencing Nodes. General VM processes (such as snapshots and backups) and platform upgrades will be faster with SSDs.		
GPU	No specific hardware cards or GPUs are required.		
Network	Gigabit Ethernet connectivity from the host server.		
Operating System	The Pexip Infinity VMs are delivered as VM images (.ova etc.) to be run directly on the hypervisor. No OS should be installed.		
Hypervisor (see also Performance considerations)	Recommended hypervisors: • VMware ESXi 6.7, 7.0 and 8.0 • KVM Supported but not recommended for new deployments: • Microsoft Hyper-V 2019		

* This does not include the processor and RAM requirements of the hypervisor.

† Sufficient for typical deployments of **up to 30** Conferencing Nodes. For deployments with **more than 30** Conferencing Nodes, you will need to increase the number of cores and the amount of RAM on the Management Node. Please contact your Pexip authorized support representative or your Pexip Solution Architect for guidance on Management Node sizing specific to your environment.

Capacity

The number of calls (or ports) that can be achieved per server in a Pexip Infinity deployment will depend on a number of things including the specifications of the particular server and the bandwidth of each call.

As a general indication of capacity: Servers that are older, have slower processors, or have fewer CPUs, will have a lower overall capacity. Newer servers with faster processors will have a greater capacity. The use of NUMA affinity and Hyper-Threading can significantly increase capacity.

Performance considerations

The type of processors and Hypervisors used in your deployment will impact the levels of performance you can achieve. Some known performance considerations are described below.

Intel AVX2 / AVX512 processor instruction set

Pexip Infinity can make full use of the AVX2 and AVX512 instruction set provided by modern Intel processors. This increases the performance of video encoding and decoding.

The VP9 codec is also available for connections to Conferencing Nodes running the AVX2 or later instruction set. VP9 uses around one third less bandwidth for the same resolution when compared to VP8. Note however that VP9 calls consume around 1.25 times the CPU resource (ports) on the Conferencing Node.

AMD processors

We have observed during internal testing that use of AMD processors results in a reduction of capacity (measured by ports per core) of around 40% when compared to an identically configured Intel platform. This is because current AMD processors do not execute advanced instruction sets at the same speed as Intel processors.

AMD processors older than 2012 may not perform sufficiently and are not recommended for use with the Pexip Infinity platform.

Memory configuration

Memory must be distributed on the different memory channels (i.e. 6 to 8 channels per socket on the Xeon Scalable series, and 4 channels per socket on the Xeon E5 and E7 series).

There must be an equal amount of memory per socket, and all sockets must have all memory channels populated (you do not need to populate all slots in a channel, one DIMM per channel is sufficient). Do not, for example, use two large DIMMs rather than four lower-capacity DIMMs — using only two per socket will result in half the memory bandwidth, since the memory interface is designed to read up from four DIMMs at the same time in parallel

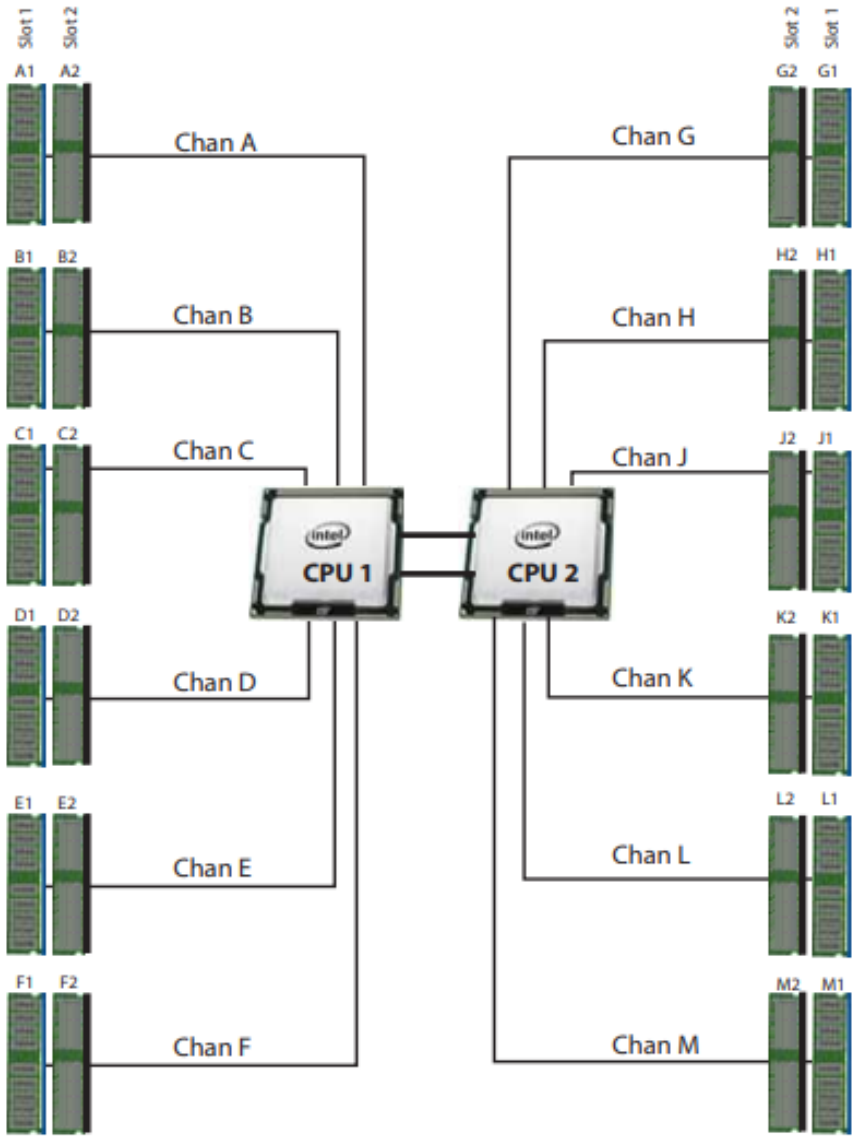
The smaller Intel Xeon Scalable Possessors (Silver and Bronze series) can be safely deployed with 4 memory channels per socket, but we do not recommend the Silver and Bronze series for most Pexip workloads.

Example - dual socket, 6 channels

Intel Xeon Scalable series dual socket system:

- Each socket has 6 channels
- All 6 channels must be populated with a DIMM
- Both sockets must have the same configuration

Therefore for a dual socket Gold 61xx you need 12 identical memory DIMMs.



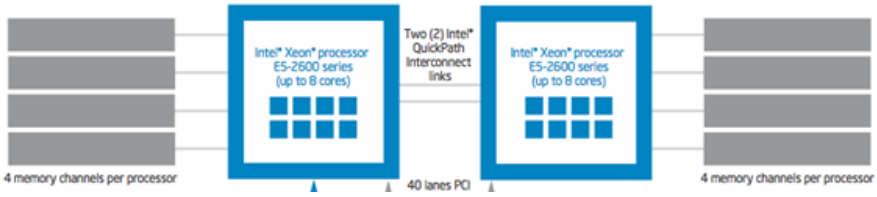
Cisco UCS C220 M5 SFF Memory Organization

Example - dual socket, 4 channels

Xeon E5-2600 dual socket system:

- Each socket has 4 channels
- All 4 channels must be populated with a DIMM
- Both sockets must have the same configuration

Therefore for a dual socket E5-2600 you need 8 identical memory DIMMs.


Intel® Xeon® processor
E5-2600 series
(up to 8 cores)
Two (2) Intel®
QuickPath
Interconnect
links
Intel® Xeon® processor
E5-2600 series
(up to 8 cores)
4 memory channels per processor
40 lanes PCI
4 memory channels per processor

Appendix 2: Achieving high density deployments with NUMA

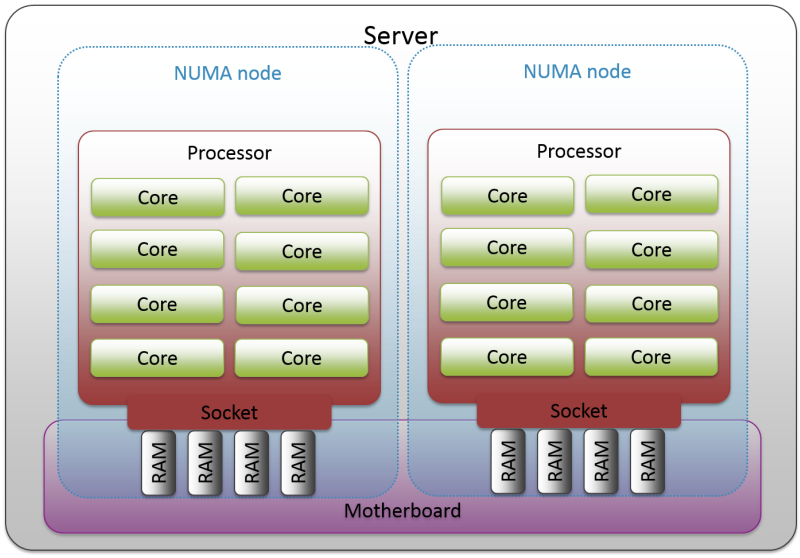
There are many factors that can affect the performance of Virtual Machines (VMs) running on host hardware. One of these is how the VM interacts with NUMA (non-uniform memory access).

This section provides an overview of NUMA and how it applies to Pexip Infinity Conferencing Nodes. It summarizes our recommendations and suggests best practices for maximizing performance.

About NUMA

NUMA is an architecture that divides the computer into a number of nodes, each containing one or more processor cores and associated memory. A core can access its local memory faster than it can access the rest of the memory on that machine. In other words, it can access memory allocated to its own NUMA node faster than it can access memory allocated to another NUMA node on the same machine.

The diagram (right) outlines the physical components of a host server and shows the relationship to each NUMA node.



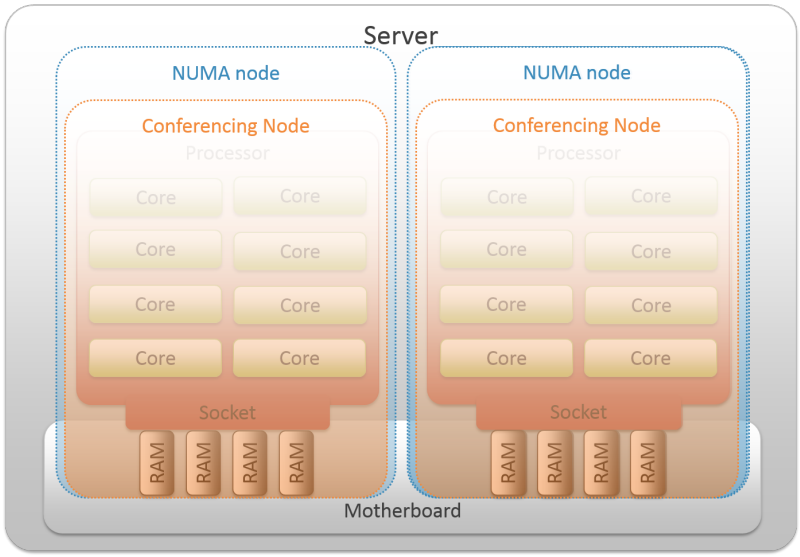
Conferencing Nodes and NUMA nodes

We strongly recommend that a Pexip Infinity Conferencing Node VM is deployed on a single NUMA node to avoid the loss of performance incurred when a core accesses memory outside its own node.

In practice, with modern servers, each socket represents a NUMA node. We therefore recommend that:

- one Pexip Infinity Conferencing Node VM is deployed per socket of the host server
- the number of vCPUs that the Conferencing Node VM is configured to use is the same as or less than the number of physical cores available in that socket (unless you are taking advantage of hyperthreading to deploy one vCPU per logical thread — in which case see NUMA affinity and hyperthreading).

This second diagram shows how the components of a Conferencing Node virtual machine relate to the server components and NUMA nodes.



You can deploy smaller Conferencing Nodes over **fewer** cores/threads than are available in a single socket, but this will reduce capacity.

Deploying a Conferencing Node over **more** cores (or threads when pinned) than provided by a single socket will cause loss of performance, as and when remote memory is accessed. This must be taken into account when moving Conferencing Node VMs between host servers with different hardware configuration: if an existing VM is moved to a socket that contains fewer cores/threads than the VM is configured to use, the VM will end up spanning two sockets and therefore NUMA nodes, thus impacting performance.

To prevent this occurring, ensure that either:

- you only deploy Conferencing Nodes on servers with a large number of cores per processor
- the number of vCPUs used by each Conferencing Node is the same as (or less than) the number of cores/threads available on each NUMA node of even your smallest hosts.

NUMA affinity and hyperthreading

You can utilize the logical threads of a socket (hyperthreading) to deploy a Conferencing Node VM with two vCPUs per physical core (i.e. one per logical thread) to achieve up to 50% additional capacity.

However, if you do this you must ensure that all Conferencing Node VMs are **pinned** to their respective sockets within the hypervisor (also known as NUMA affinity). Otherwise, the Conferencing Node VMs will end up spanning multiple NUMA nodes, resulting in a loss of performance.

Affinity does NOT guarantee or reserve resources, it simply forces a VM to use only the socket you define, so mixing Pexip Conferencing Node VMs that are configured with NUMA affinity together with other VMs on the same server is not recommended.

NUMA affinity is not practical in all data center use cases, as it forces a given VM to run on a certain CPU socket (in this example), but is very useful for high-density Pexip deployments with dedicated capacity.

NUMA affinity for Pexip Conferencing Node VMs should only be used if the following conditions apply:

- If you are using Hyper-V, it is part of a Windows Server Datacenter Edition (the Standard Edition does not have the appropriate configuration options).
- The server/blade is used for Pexip Conferencing Node VMs only, and the server will have only one Pexip Conferencing Node VM per CPU socket (or two VMs per server in a dual socket CPU e.g. E5-2600 generation).
- vMotion (VMware) or Live Migration (Hyper-V) is NOT used. (Using these may result in having two nodes both locked to a single socket, meaning both will be attempting to access the same processor, with neither using the other processor.)
- You fully understand what you are doing, and you are happy to revert back to the standard settings, if requested by Pexip support, to investigate any potential issues that may result.

Step-by-step guides

For instructions on how to achieve NUMA pinning (also known as NUMA affinity) for your particular hypervisor, see:

Achieving ultra-high density with Sub-NUMA Clustering

In almost all circumstances, we recommend that Sub-NUMA Clustering [SNC] is turned off. Where SNC is enabled and there is a single Pexip Infinity node on that socket, the node is likely to underutilize resources and can fail.

We recommend a maximum of 48 vCPU per transcoding node, up to 56 vCPU for parts with a high base clock speed. Some 3rd- and 4th-generation Intel Xeon Scalable Series processors have well in excess of 28 physical cores, so it is not possible to utilize the whole processor with Hyper-Threading on a single transcoding node.

Optimizing for density

Our standard recommendation for high performance is the Intel Xeon Gold 6342. To gain higher density it is currently necessary to move up to the Xeon Platinum line which is significantly more expensive. In many cases it is cheaper to deploy more Xeon Gold 6342 machines to gain extra capacity.

Where rack space is at a premium or a requirement for more than 2S dictates a Xeon Platinum line anyway, increasing density with SNC often represents a sensible choice.

Two transcoding nodes per socket

All 3rd- and 4th-generation Intel Xeon Scalable Series processors support SNC. For parts with 32 physical cores or more, we recommend using SNC to treat the processor as two separate NUMA nodes. Under normal operation, a 1U 2-socket server with 2x Intel Xeon Gold 6342 processors would achieve around 195HD capacity. Using a processor with a higher core count and SNC allows the same 1U 2-socket chassis to offer over 300HD of transcoding capacity over four transcoding nodes.

Deployment

As with most hypervisor features, we recommend that this is carried out by people who possess advanced skills with the relevant hypervisor.

Each socket should be split into two equally-sized sub-NUMA nodes, 0 and 1. For node 0, use the entirety of the node for the transcoding node; for node 1 reserve 2 vCPU for the hypervisor and use the rest of it for another transcoding node.

Example

An Intel Xeon Platinum 8360Y has 36 physical cores. With only a 2.4GHz base clock speed, it is not the ideal choice: a processor with a higher clock speed will give better results.

Use cores 0-17 as sub-NUMA node 0 and cores 18-35 as sub-NUMA node 1. Cores 0-17 should be used as 36 vCPU Hyper-Threaded transcoding node, and cores 18-34 should be used as a 34 vCPU transcoding node with core 35 reserved for the hypervisor.

In this case the 2-socket server produces around 280HD of capacity; a faster or larger processor could easily exceed 300HD per rack unit.

Summary of deployment recommendations

We are constantly optimizing our use of the host hardware and expect that some of this advice may change in later releases of our product. However our current recommendations are:

- Prefer processors with a high core count.
- Prefer a smaller number of large Conferencing Nodes rather than a larger number of smaller Conferencing Nodes.
- Deploy one Conferencing Node per NUMA node (i.e. per socket).
- Configure **one vCPU per physical core** on that NUMA node (without hyperthreading and NUMA pinning), or **one vCPU per logical thread** (with hyperthreading and all VMs pinned to a socket in the hypervisor).
- Populate memory equally across all NUMA nodes on a single host server.
- Do not over-commit resources on hardware hosts.

Appendix 3: VMware NUMA affinity and hyperthreading

This topic explains how to experiment with VMware NUMA affinity and Hyper-Threading Technology for Pexip Infinity Conferencing Node VMs, in order to achieve up to 50% additional capacity.

If you are taking advantage of hyperthreading to deploy two vCPUs per physical core (i.e. one per logical thread), you must first enable NUMA affinity; if you don't, the Conferencing Node VM will end up spanning multiple NUMA nodes, resulting in a loss of performance.

Affinity does NOT guarantee or reserve resources, it simply forces a VM to use only the socket you define, so mixing Pexip Conferencing Node VMs that are configured with NUMA affinity together with other VMs on the same server is not recommended.

NUMA affinity is not practical in all data center use cases, as it forces a given VM to run on a certain CPU socket (in this example), but is very useful for high-density Pexip deployments with dedicated capacity.

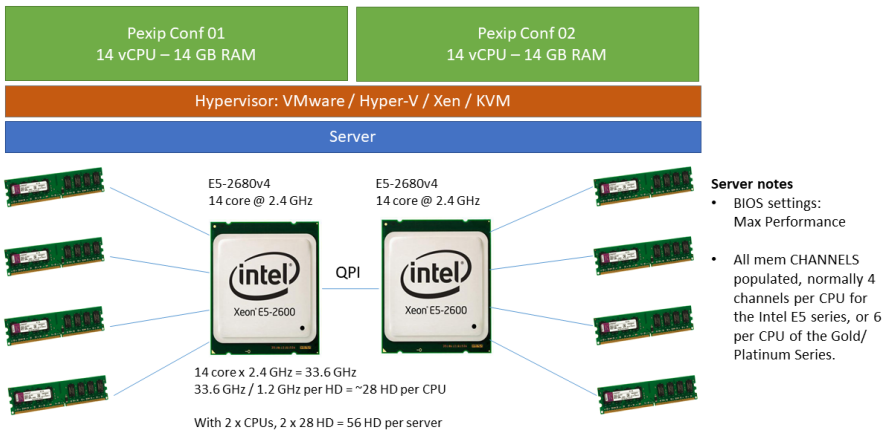
This information is aimed at administrators with a strong understanding of VMware, who have very good control of their VM environment, and who understand the consequences of conducting these changes.

Please ensure you have read and implemented our recommendations in Appendix 2: Achieving high density deployments with NUMA before you continue.

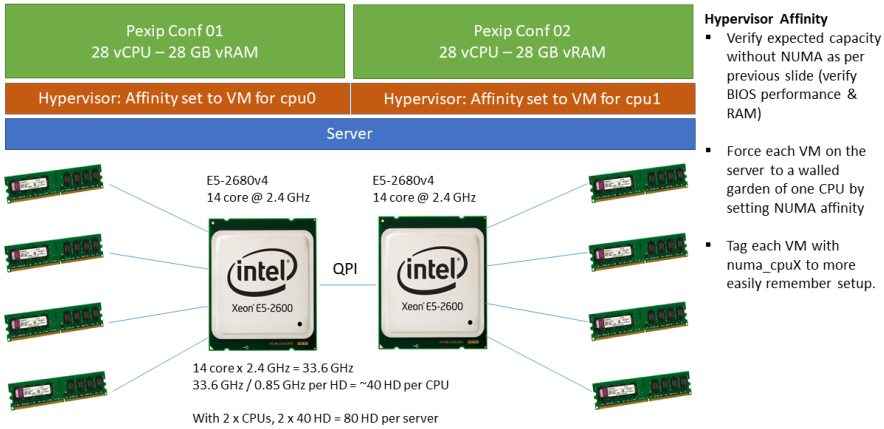
Prerequisites

VMware NUMA affinity for Pexip Conferencing Node VMs should only be used if the following conditions apply:

- The server/blade is used for Pexip Conferencing Node VMs only, and the server will have only one Pexip Conferencing Node VM per CPU socket (or two VMs per server in a dual socket CPU e.g. E5-2600 generation).
- vMotion is NOT used. (Using this may result in having two nodes both locked to a single socket, meaning both will be attempting to access the same processor, with neither using the other processor.)
- You fully understand what you are doing, and you are happy to revert back to the standard settings, if requested by Pexip support, to investigate any potential issues that may result.



Example server without NUMA affinity - allows for more mobility of VMs



Example server with NUMA affinity - taking advantage of hyperthreading to gain 30-50% more capacity per server

Overview of process

We will configure the two Conferencing Node VMs (in this example, an E5-2600 CPU with two sockets per server) with the following advanced VMware parameters:

Conferencing Node A locked to Socket 0

- cpuid.coresPerSocket = 1
- numa.vcpu.preferHT = TRUE
- numa.nodeAffinity = 0

Conferencing Node B locked to Socket 1

- cpuid.coresPerSocket = 1
- numa.vcpu.preferHT = TRUE
- numa.nodeAffinity = 1

You must also double-check the flag below to ensure it matches the number of vCPUs in the Conferencing Node:

- numa.autosize.vcpu.maxPerVirtualNode

For example, it should be set to 24 if that was the number of vCPUs you assigned.

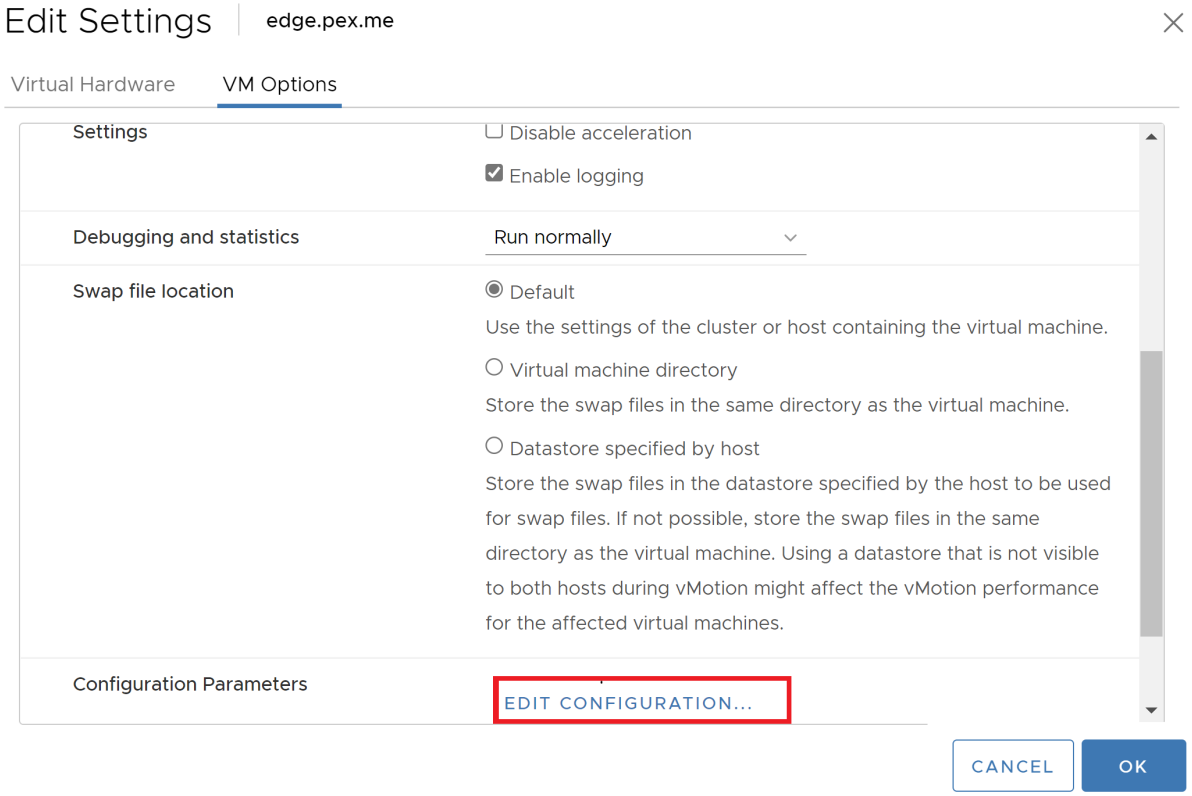
Note that if you are experiencing different sampling results from multiple nodes on the same host, you should also ensure that `Numa.PreferHT = 1` is set (to ensure it operates at the ESXi/socket level). See https://kb.vmware.com/s/article/2003582 for more information.

Setting NUMA affinity

Before you start, please consult your local VMware administrator to understand whether this is appropriate in your environment.

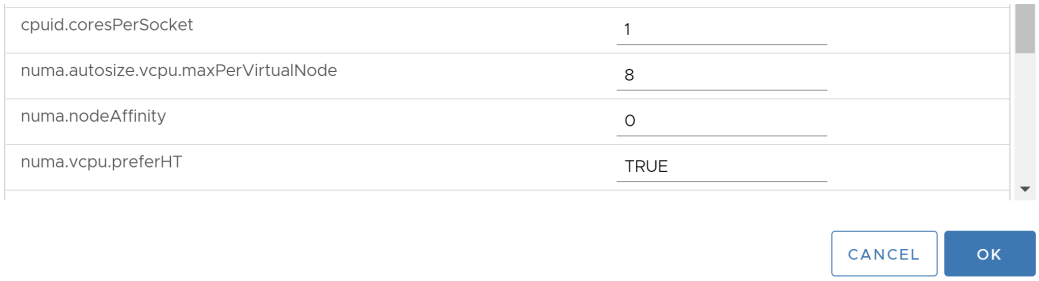
1. Shut down the Conferencing Node VMs, to allow you to edit their settings.
2. Give the Conferencing Node VMs names that indicate that they are locked to a given socket (NUMA node). In the example below the VM names are suffixed by **numa0** and **numa1**:

3. Right-click the first Conferencing Node VM in the inventory and select **Edit Settings**.
4. From the **VM Options** tab, expand the **Advanced** section and select **Edit Configuration**:



5. At the bottom of the window that appears, enter the following Names and corresponding Values for the first VM, which should be locked to the first socket (**numa0**):
   - cpuid.coresPerSocket = 1
   - numa.vcpu.preferHT = TRUE
   - numa.nodeAffinity = 0

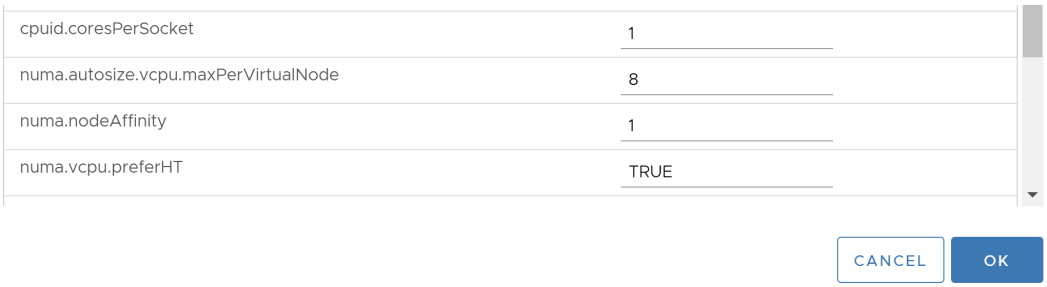
   It should now look like this in the bottom of the parameters list:



6. Select **OK** and **OK** again.

   Now our **conf-node_numa0** Virtual Machine is locked to **numa0** (the first socket).
7. Repeat the above steps for the second node, entering the following data for the second VM, which should be locked to the second socket (**numa1**):
   - cpuid.coresPerSocket = 1
   - numa.vcpu.preferHT = TRUE
   - numa.nodeAffinity = 1

   It should now look like this in the bottom of the parameters list:



8. Select **OK** and **OK** again.

   Now our **conf-node_numa1** Virtual Machine is locked to **numa1** (the second socket).

It is very important that you actually set **numa.nodeAffinity** to **1** and not **0** for the second node. If both are set to **0**, you will effectively only use numa node 0, and they will fight for these resources while leaving numa node 1 unused.

Increasing vCPUs

You must now increase the number of vCPUs assigned to your Conferencing Nodes, to make use of the hyperthreaded cores. (Hyperthreading must always be enabled, and is generally enabled by default.)

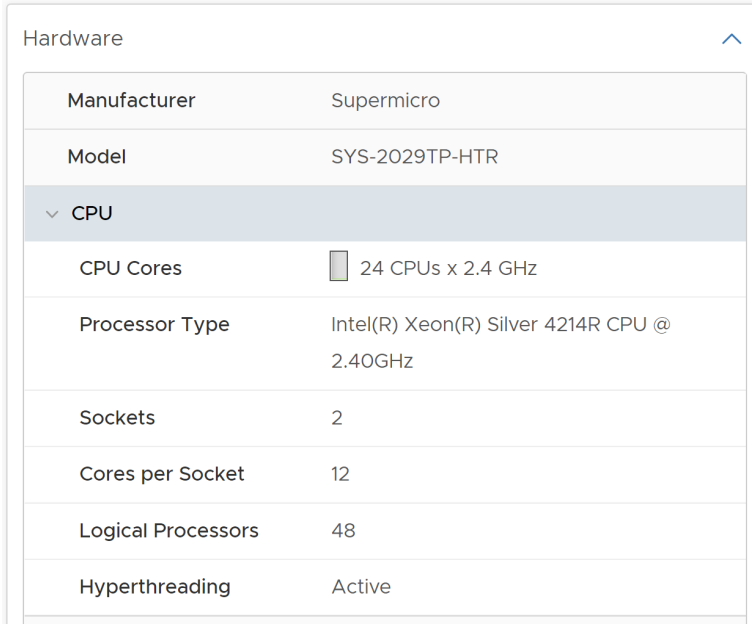
Count logical processors

First you must check how many logical processors each CPU has.

In the example screenshot below, the E5-2680 v3 CPU has **12** physical cores per CPU socket, and there are two CPUs on the server.

With hyperthreading, each physical core has **2** logical processors , so the CPU has **24** logical processors (giving us a total of 48 with both CPUs).

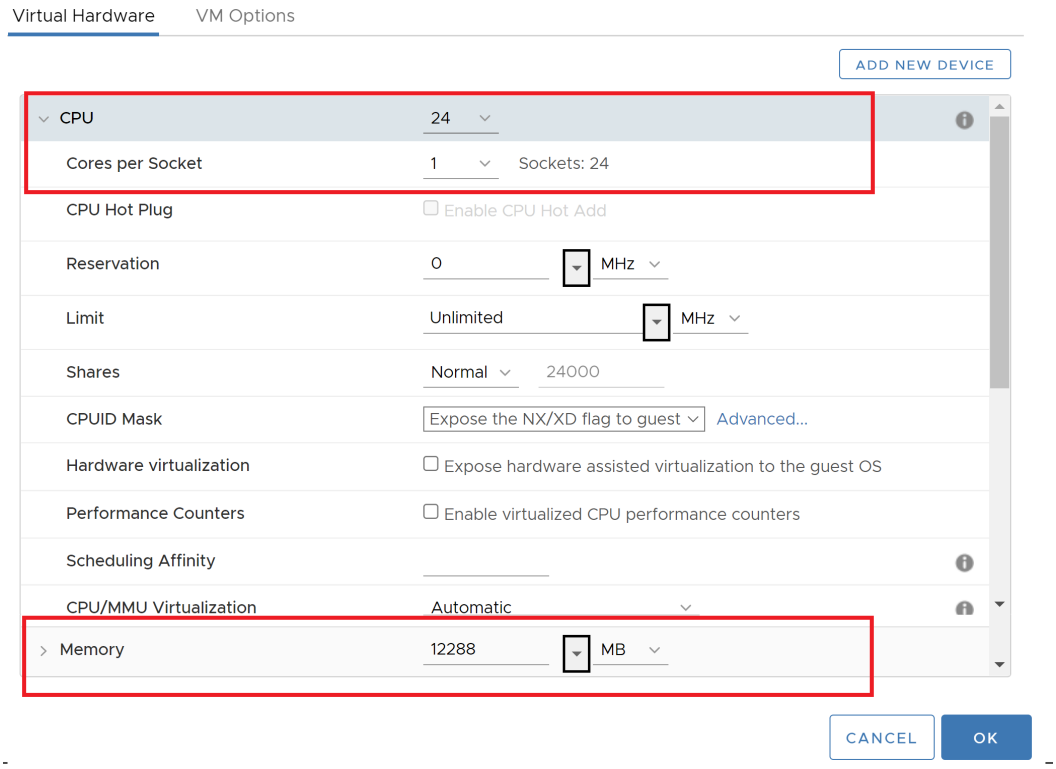
In this case **2 x 12 = 24** is the "magic number" we are looking for with our Conferencing Nodes - which is double the amount of **Cores per Socket**.



Assign vCPU and RAM

Next, you must edit the settings on the Virtual Machines to assign 24 vCPU and 24 GB RAM to each of the two Conferencing Nodes.

Ensure that the server actually has 24 GB of RAM connected to each CPU socket. Since all four memory channels should be populated with one RAM module each, you will normally require 4 x 8 GB per CPU socket.



Reboot

Finally, save and boot up your virtual machines. After about 5 minutes they should be booted, have performed their performance sampling, and be available for calls.

Viewing updated capacity

To view the updated capacity of the Conferencing Nodes, log in to the Pexip Management Node, select **Status > Conferencing Nodes** and then select one of the nodes you have just updated. The **Maximum capacity - HD connections** field should now show slightly less than one HD call per GHz (compared to the previous one HD call per 1.41 GHz).

In our example, 12 physical cores x 2.6 GHz = 31.2 GHz, so the Conferencing Node should show around 30 or 31 HD calls, assuming a balanced BIOS power profile. With maximum performance BIOS power profiles, the results could be up to 33-34 HD calls per Conferencing Node VM.

Our first VM:

Conferencing Node status

Name	softlayer-lon02-cnf01 (Edit configuration)
IPv4 Address	159.8.179.100
Secondary address	Not configured
System location	London-Softlayer
Deployment status	Deployment succeeded
Maintenance mode	No
Version	14 (33724.0.0)
Last contacted	2017-01-12 12:03:55 (GMT)
Last updated	2017-01-11 21:28:51 (GMT)
Maximum capacity - audio connections	240
Maximum capacity - SD connections	60
Maximum capacity - HD connections	30
Maximum capacity - Full HD connections	14
Media load	0 %

Our second VM:

Conferencing Node status

Name	softlayer-lon02-cnf02 (Edit configuration)
IPv4 Address	5.10.121.84 00
Secondary address	Not configured
System location	London-Softlayer
Deployment status	Deployment succeeded
Maintenance mode	No
Version	14 (33724.0.0)
Last contacted	2017-01-12 12:03:55 (GMT)
Last updated	2017-01-11 21:28:51 (GMT)
Maximum capacity - audio connections	240
Maximum capacity - SD connections	60
Maximum capacity - HD connections	31
Maximum capacity - Full HD connections	14
Media load	0 %

Checking for warnings

You should check for warnings by searching the administrator log (**History & Logs > Administrator Log**) for "`sampling`".

A successful run of the above example should return something like:

```
2015-04-05T18:25:40.390+00:00 softlayer-lon02-cnf02 2015-04-05 18:25:40,389 Level="INFO" Name="administrator.system"
Message="Performance sampling finished" Detail="HD=31 SD=60 Audio=240"
```

An unsuccessful run, where VMware has split the Conferencing Node over multiple NUMA nodes, would return the following warning in addition to the result of the performance sampling:

```
2015-04-06T17:42:17.084+00:00 softlayer-lon02-cnf02 2015-04-06 17:42:17,083 Level="WARNING" Name="administrator.system"
Message="Multiple numa nodes detected during sampling" Detail="We strongly recommend that a Pexip Infinity Conferencing Node is
deployed on a single NUMA node"
2015-04-06T17:42:17.087+00:00 softlayer-lon02-cnf02 2015-04-06 17:42:17,086 Level="INFO" Name="administrator.system"
Message="Performance sampling finished" Detail="HD=21 SD=42 Audio=168"
```

If you have followed the steps in this guide to set NUMA affinity correctly and you are getting the warning above, this could be due to another VMware setting. From VMware, select the Conferencing Node and then select **Edit Settings > Options > General > Configuration Parameters...**). The **numa.autosize.vcpu.maxPerVirtualNode** option should be set to your "magic number". For example, 24 is our "magic number" - the number of logical processors, or vCPUs, assigned in our example.

If this option is set to anything lower, e.g. 8, 10 or 12, VMware will create two virtual NUMA nodes, even if locked on one socket.

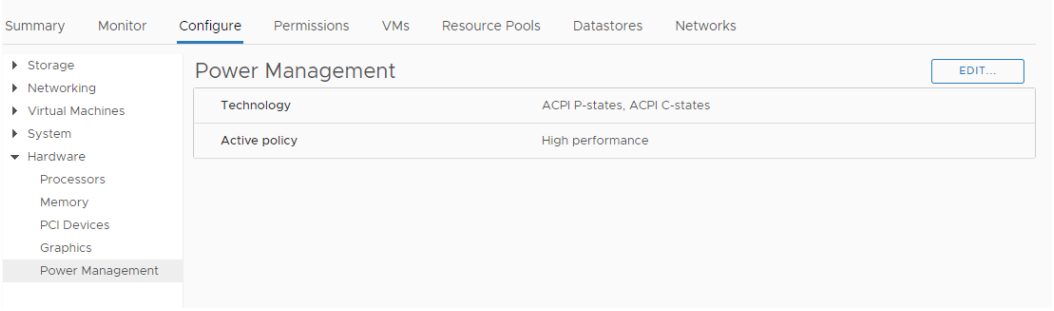
BIOS settings

Ensure all BIOS settings pertaining to power saving are set to maximize performance rather than preserve energy. (Setting these to an energy-preserving or balanced mode may impact transcoding capacity, thus reducing the total number of HD calls that can be provided.) While this setting will use slightly more power, the alternative is to add another server in order to achieve the increase in capacity, and that would in total consume more power than one server running in high performance mode.

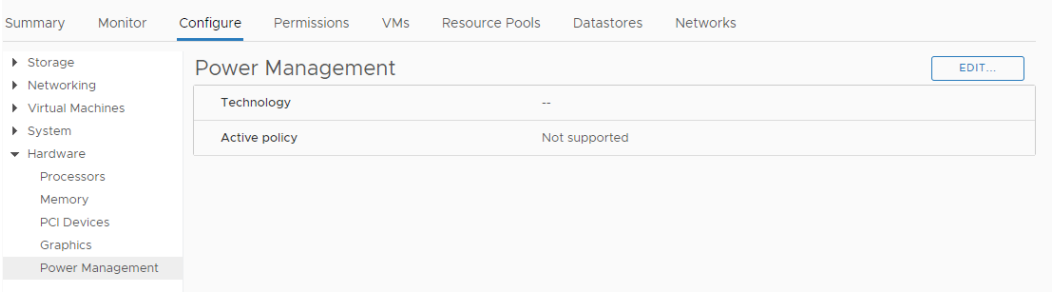
The actual settings will depend on the hardware vendor; see BIOS performance settings for some examples.

A quick way to verify that BIOS has been set appropriately is to check the hardware's **Power Management** settings in VMware (select the host then select **Configure > Hardware > Power Management**). In most cases, the ACPI C-states should **not** be exposed to VMware when BIOS is correctly set to maximize performance.

If the **ACPI C-states are showing** in VMware (as shown below), the BIOS has most likely **not been set to maximize performance** :



When BIOS has been correctly set to **maximize performance**, it should in most cases look like this:



If your server is set to maximize performance, but VMware still shows ACPI C-states, change it to balanced (or similar), and then change back to maximize performance. This issue has been observed with some Dell servers that were preconfigured with maximize performance, but the setting did not take effect initially.

VMware and NUMA

As well as the physical restrictions discussed above, the hypervisor can also impose restrictions. VMware provides virtual NUMA nodes on VMs that are configured with more than 8 CPUs. If you have fewer than 8 CPUs, you should change this default by setting **numa.vcpu.min** in the VM's configuration file to the number of vCPUs you wish to configure (which will be double the number of CPUs you have available).

For more information, see https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.resmgmt.doc/GUID-3E956FB5-8ACB-42C3-B068-664989C3FF44.html.

Appendix 4: Hyper-V NUMA affinity and hyperthreading

This topic explains how to experiment with NUMA pinning and Hyper-Threading Technology for Pexip Infinity Conferencing Node VMs, in order to achieve up to 50% additional capacity. You must be using Hyper-V as part of a Windows Server **Datacenter Edition** to do this.

If you are taking advantage of hyperthreading to deploy two vCPUs per physical core (i.e. one per logical thread), you must first enable NUMA affinity; if you don't, the Conferencing Node VM will end up spanning multiple NUMA nodes, resulting in a loss of performance.

Affinity does NOT guarantee or reserve resources, it simply forces a VM to use only the socket you define, so mixing Pexip Conferencing Node VMs that are configured with NUMA affinity together with other VMs on the same server is not recommended.

NUMA affinity is not practical in all data center use cases, as it forces a given VM to run on a certain CPU socket (in this example), but is very useful for high-density Pexip deployments with dedicated capacity.

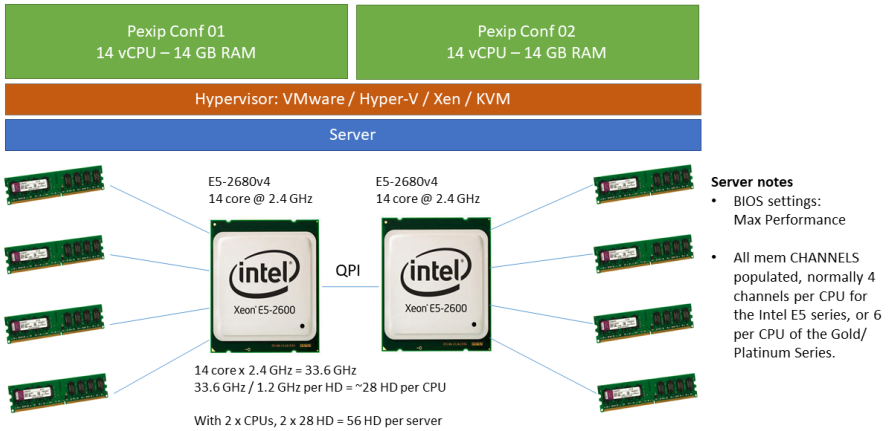
This information is aimed at administrators with a strong understanding of Hyper-V, who have very good control of their VM environment, and who understand the consequences of conducting these changes.

Please ensure you have read and implemented our recommendations in Appendix 2: Achieving high density deployments with NUMA before you continue.

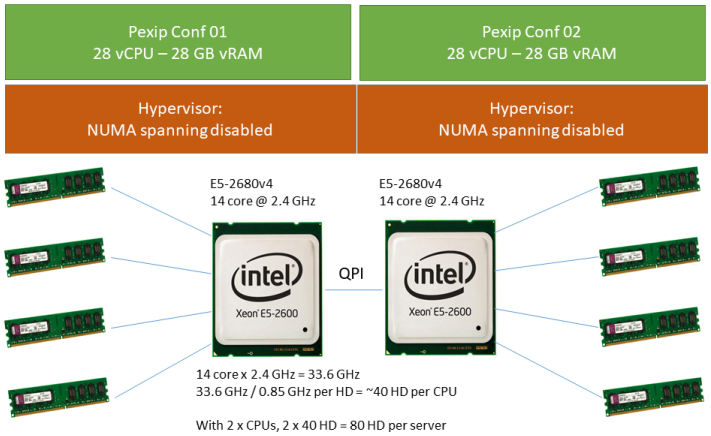
Prerequisites

NUMA affinity for Pexip Conferencing Node VMs should only be used if the following conditions apply:

- You are using Hyper-V as part of a Windows Server Datacenter Edition (the Standard Edition does not have the appropriate configuration options).
- The server/blade is used for Pexip Conferencing Node VMs only, and the server will have only one Pexip Conferencing Node VM per CPU socket (or two VMs per server in a dual socket CPU e.g. E5-2600 generation).
- Live Migration is NOT used. (Using this may result in having two nodes both locked to a single socket, meaning both will be attempting to access the same processor, with neither using the other processor.)
- You fully understand what you are doing, and you are happy to revert back to the standard settings, if requested by Pexip support, to investigate any potential issues that may result.



Example server without NUMA affinity - allows for more mobility of VMs



Example server with NUMA affinity - taking advantage of hyperthreading to gain 30-50% more capacity per server

Example hardware

In the example given below, we are using a SuperMicro SuperServer with dual Intel Xeon E5-2680-v3 processors, 64GB RAM, and 2 x 1TB hard drives.

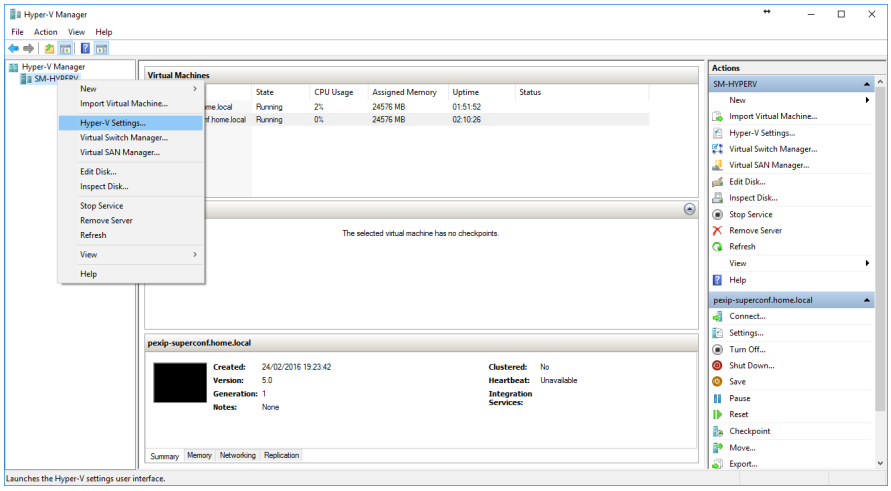
On this server:

- we deploy one Conferencing Node VM per processor/socket, so two Conferencing Nodes in total
- we disable NUMA spanning, so each Conferencing Node VM runs on a single NUMA node/processor/socket
- each processor has 12 physical cores
- we use hyperthreading to deploy 2 vCPUs per physical core
- this gives us 24 vCPUs / 24 threads per Conferencing Node
- therefore we get 48vCPUs / 24 threads in total on the server.

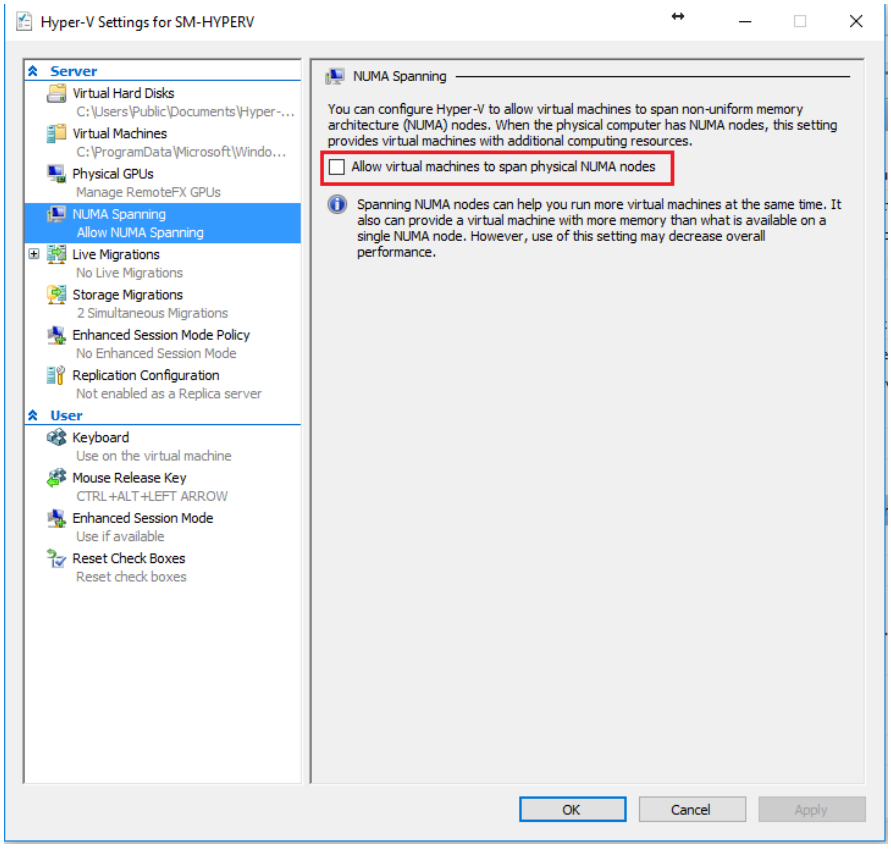
Disabling NUMA spanning on the server

Firstly, we must disable NUMA spanning on the server. To do this:

1. From within Hyper-V Manager, right-click on the server and select **Hyper-V Settings...**:



2. From the Server section, select **NUMA Spanning** and disable **Allow virtual machines to span physical NUMA nodes**. This ensures that all processing will remain on a single processor within the server:

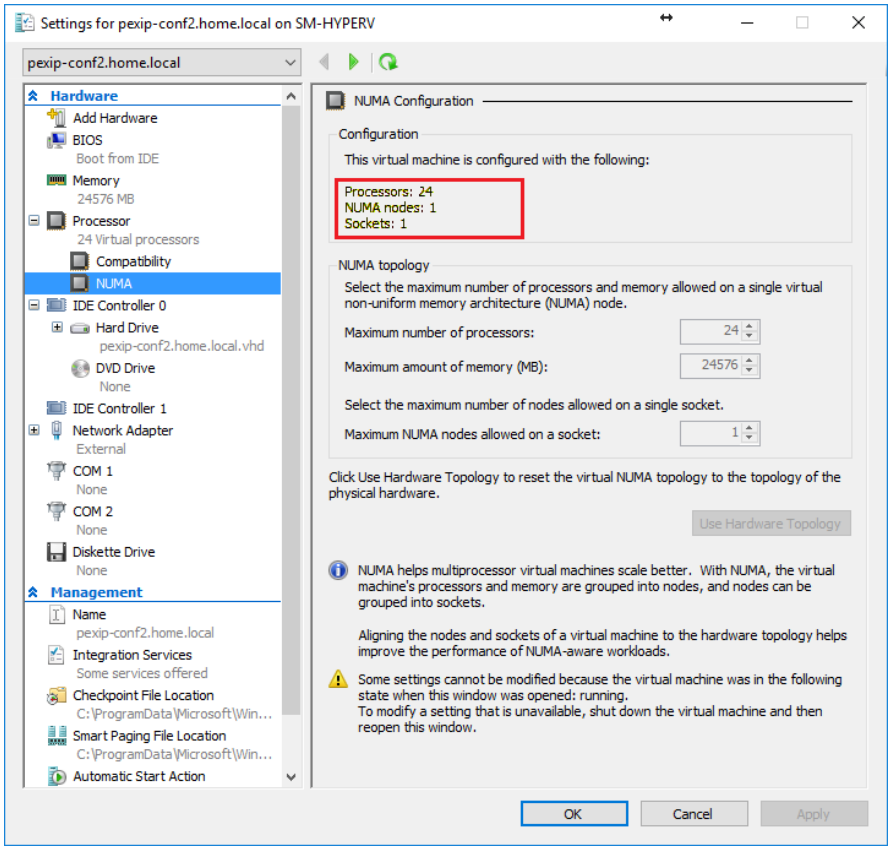


Disable NUMA spanning on the VM

Next we need to ensure the Conferencing Node VMs have the correct settings too, and do not span multiple processors.

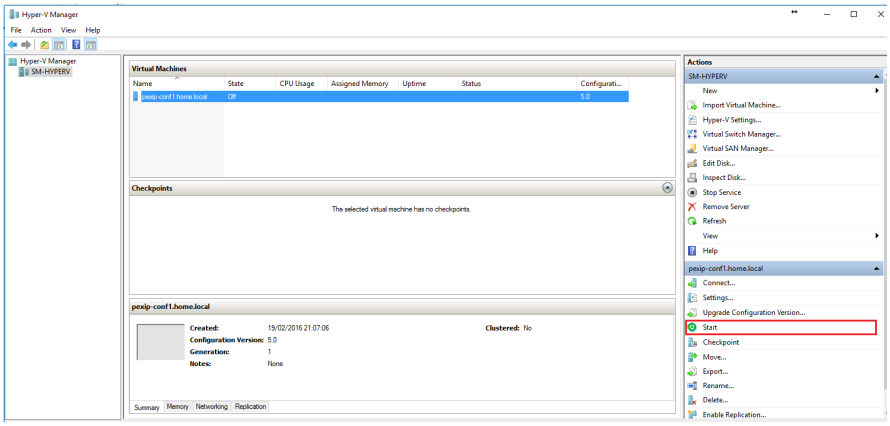
To do this:

1. From within Hyper-V, select the Conferencing Node VM, and then select **Settings > Hardware > Processor > NUMA**.
2. Confirm that only 1 NUMA node and 1 socket are in use by each Conferencing Node VM:



Starting the Virtual Machine

After the NUMA settings have been changed, you can start up each of the Conferencing Node VMs:



Viewing performance and checking for warnings

Every time a Conferencing Node is started up or rebooted, the Pexip Infinity Management Node will perform a sampling of the system to understand what capabilities it has. To view this information, go to the administrator log (**History & Logs > Administrator Log**) and search for "`sampling`".

A successful run of the above example should return something like:

```
2015-04-05T18:25:40.390+00:00 softlayer-lon02-cnf02 2015-04-05 18:25:40,389 Level="INFO" Name="administrator.system"
Message="Performance sampling finished" Detail="FULLHD=17 HD=33 SD=74 Audio=296"
```

An unsuccessful run, where Hyper-V has split the Conferencing Node over multiple NUMA nodes, would return the following warning in addition to the result of the performance sampling:

```
2015-04-06T17:42:17.084+00:00 softlayer-lon02-cnf02 2015-04-06 17:42:17,083 Level="WARNING" Name="administrator.system"
Message="Multiple numa nodes detected during sampling" Detail="We strongly recommend that a Pexip Infinity Conferencing Node is
deployed on a single NUMA node"
2015-04-06T17:42:17.087+00:00 softlayer-lon02-cnf02 2015-04-06 17:42:17,086 Level="INFO" Name="administrator.system"
Message="Performance sampling finished" Detail="HD=21 SD=42 Audio=168"
```

Moving VMs

When moving Conferencing Node VMs between hosts, you must ensure that the new host has at least the same number of cores. You must also remember to disable NUMA spanning on the new host.

BIOS settings

Ensure all BIOS settings pertaining to power saving are set to maximize performance rather than preserve energy. (Setting these to an energy-preserving or balanced mode may impact transcoding capacity, thus reducing the total number of HD calls that can be provided.) While this setting will use slightly more power, the alternative is to add another server in order to achieve the increase in capacity, and that would in total consume more than one server running in high performance mode.

The actual settings will depend on the hardware vendor; see BIOS performance settings for some examples.