WHITE PAPER
JUNE 2016

IT@INTEL

Simplifying Private Cloud Capacity Management

Through preproduction stress testing, we found that a private cloud host with the 16-core, 2.3 GHz Intel® Xeon® processor E5-2698 v3 and 512 GB memory was the most cost-effective cloud host configuration.

Jon Petersen
Cloud Capacity Manager, Intel IT

Leo Godin
Cloud Analytics Engineer, Intel IT

Pushpa Jahagirdar
Senior Financial Analyst,
Intel Finance

Steven Jones
Cloud Engineer, Intel IT

Lauri Minas
Contributing Author, Intel IT

Executive Overview

To drive down cost and increase host utilization in Intel's Office and Enterprise private cloud, Intel IT has simplified private cloud capacity management by transitioning from allocation-based capacity management to performance-based capacity management. Focusing on host and application performance in conjunction with optimizing our private cloud host server configuration has significantly increased the amount of virtual machines (VMs) and reduced our private cloud host server footprint and cost per VM.

We use three metrics to increase the density of our private cloud hosts (each for private cloud host and VM): total MHz CPU utilization, CPU Ready percentage, and maximum active memory utilization. Using these performance metrics to inform our capacity decisions has resulted in the following:

- An 11:1 average virtual CPU to physical CPU allocation ratio
- An average of 180-percent memory oversubscription
- A 19-percent increase in VM growth while decreasing our private cloud host server footprint by 23 percent during the fall 2014 to fall 2015 time period
- A licensing cost avoidance of between 34 percent and 76 percent

During the time we were using allocation-based performance management, we believed that our private cloud was memory-constrained and needed hosts with more memory. Transitioning to performance-based capacity management and increasing the level of our memory oversubscription led us to discover that CPU utilization—not memory—was the key metric that was preventing us from achieving higher private cloud host density.

Through performance testing, we found that a private cloud host with the 16-core, 2.3 GHz Intel® Xeon® processor E5-2698 v3 and 512 GB memory is the most cost-effective private cloud host configuration for our Office and Enterprise private cloud.

Contents

Acronyms

BKM	best-known method
PoC	proof of concept
pCPU	physical CPU
TCO	total cost of ownership
vCPU	virtual CPU
VM	virtual machine

Intel® Xeon® Processor E5-2698 v3 Features

Some of the notable features of the Intel® Xeon® processor E5-2698 v3 that were used in our test studies include the following:

- Intel® Turbo Boost Technology 2.0 helped with performance upside.
- PCI Express* 3.0 support provided better I/O latency and bandwidth.
- High-bandwidth, low-latency bidirectional ring interconnect allowed faster access to the 45-MB multi-banked last-level L3 cache.
- Intel® Hyper-Threading Technology enabled up to 32 computational threads per socket.
- Integrated memory controller with four DDR4 memory channels and 46-bit physical addressing facilitated greater memory capacity.

Business Challenge

We needed to better understand several aspects of our private cloud hosting environment before we could meet our goals of reducing costs and improving private cloud host utilization.

Capacity Management: Allocation or Performance

When we first began virtualizing our physical server footprint, we took a safe approach by striving to allocate 80 percent of the CPU and memory resources on a private cloud host, much like an engineer would do when sizing an application for a physical dedicated server. As we became more comfortable with virtualization, we began to allocate slightly over 100 percent of the resources. The problem with using allocation metrics as an indicator for cloud host density is that not all VMs are created equal, and the performance requirements of VMs can vary drastically depending on when the VM is used most, its service-level agreement, and its tier. When coupled with the CPU and memory optimizations that most modern hypervisors offer, we found that the majority of our private cloud hosts reached a maximum of only 40-percent CPU utilization and 20-percent memory utilization.

Utilizing the Full Potential of Cloud Host Server Resources

We strive to follow three best-known methods (BKMs) to maximize our investment in private cloud host server resources: right sizing, overcommitting memory, and checking storage performance.

- **Right-sizing.** We allocate the exact amount of CPU and memory resources a VM needs to perform its function. We routinely review large VMs in our environment and reduce CPU and memory configurations in cases where resources are not being consumed.
- **Overcommitting memory.** Utilizing the memory optimization features on modern hypervisors such as memory sharing, compression, and ballooning, we overcommit as much memory as possible on our private cloud hosts to achieve an active memory utilization metric of 80 percent, which is an effective memory allocation of over 200 percent on some of our private cloud hosts.
- **Checking storage performance.** We periodically review the storage performance of our private cloud. The latency endured by a storage frame can mask itself as CPU or memory contention, leading many capacity managers to believe an environment needs to add cloud hosts. Routinely reviewing storage metrics with storage administrators can pinpoint the issues before money has to be spent on capacity. The storage performance of our storage area network infrastructure has been an insignificant data point; we did not need to purchase additional storage capacity.

Choosing a Cloud Host Hardware Configuration for Cost Optimization

Before choosing a private cloud host server configuration, we needed to understand the performance characteristics of the VMs in our environment. Since our storage solution was sized properly, we investigated the CPU and memory demand of our VMs. We did this by determining the VMs' total CPU MHz utilization, sensitivity to the CPU Ready metric, and over-allocation of the cloud host core, and the maximum amount of active memory used. Before we virtualized anything, we identified the performance characteristics of our workloads. Identifying these applications was an important first step.

Building a Server Capacity Plan

A server capacity plan is a valuable tool when measuring performance or determining a cloud hosting strategy. At Intel IT, our server capacity plan has four vectors: capacity demand, capacity show-back and reclamation, performance monitoring, and business continuity and disaster recovery.

- **Capacity demand.** We determine the application performance requirements for today and in the future. Application owners are asked to benchmark their applications in preproduction. We do this in order to minimize over-allocating resources to virtual machines (VMs), which can degrade application performance. We also look at the expected application demand, growth, and usage according to the time of day, week, month, the time zone, and during holiday times. Once we have an estimate of the requested capacity demand, we can then determine an efficiency percentage goal for hardware utilization. The more efficient the hardware used, the greater the costs savings, but this also comes at a cost of increased outage risks from unexpected resource spikes and new application growth. Overcommitment of resources can happen on cloud hosts as well as internal servers. Allocating too many virtual CPUs to a VM will degrade application performance.
- **Capacity show-back and reclamation.** We strive to right-size resources and document end-of-life procedures for VMs. For VMs that are not sized correctly, we document a process for reclaiming CPU and memory. One way to help align resources with capacity is to set priorities of application resources. Critical applications can have cloud resources reserved at the expense of decreased cloud host density. Our end-of-life process monitors VM value over time. Abandoned VMs are reclaimed. Some application owners perceive cloud resources as free, leading to wasted resources. To maintain IT resource efficiency, we are deploying a tool that shows resource usage in dollars to upper management according to user and application.
- **Performance monitoring.** We monitor the entire private cloud stack. This includes operating systems, cloud hosts, hypervisors, network components, and storage components. We recommend monitoring tools that are hardware- or vendor-agnostic. For applications, we analyze performance, capacity, and configuration. We use these analytics to help forecast demand and find performance bottlenecks before they happen.
- **Business continuity and disaster recovery.** Fault tolerance and clustering needs can affect the required cloud pool resources. Our plans identify which applications need availability and support in multiple regions. Because we have data centers around the world, we also document backup suppliers and the backup procedures per region. This is important with cloud pools, because there can be unique backup capabilities and requirements for virtualization.

Cloud Host Density TCO Study

With the release of the Intel® Xeon® processor E5-2600 v3 product family, we began testing private cloud host server configurations to find the best cost-per-virtual CPU (vCPU) ratio. Testing was conducted in an actively used preproduction environment with a mix of private cloud host servers based on the Intel® Xeon® processor x5675, Intel® Xeon® processor E5-2670, Intel® Xeon® processor E5-2680 v2, and Intel® Xeon® processor E5-2698 v3. We compared the performance throughput and total cost of ownership (TCO) of two-socket servers with varying core counts and frequency levels starting from 6 cores through 18 cores.

25% MORE CORES

Intel® Xeon® processor E5-2600 v3 product family includes up to 18 cores and up to 45 MB of last-level cache.

We included the following four-year TCO elements in our analysis:

- **Hardware platforms.** We based our analysis on mainstream two-socket servers from major manufacturers.
- **Software.** We included the license and maintenance costs of software including the OS, applications, middleware, security products, backup and restore, and manageability (monitoring, alerting, compliance, patching, and provisioning).
- **Data center.** We included data center power, cooling, storage connectivity, and network connectivity costs.

Test Methodology

We ran stress tests on live preproduction workloads and collected data for one week. Host allocation was collected as point-in-time metrics at 30-minute intervals. Performance data was collected at 5-minute intervals using averages for utilization metrics and summation for CPU Ready percentage. Transformations included normalizing all memory metrics to megabytes and normalizing CPU Ready to the percentage of time a VM is ready to perform work but is waiting for physical CPU resources.

We recorded throughput for each platform, measuring and comparing the time taken to complete a specific workload. To maximize throughput, we configured the applications to maximize use of the available cores. This resulted in multiple simultaneous jobs on each platform.

Total CPU MHz Utilization and CPU Ready Percentage

The total CPU MHz utilization metric is translated the same for both private cloud host servers and VMs. For both cases, we strive to achieve as close to an average of 80 percent of the average CPU utilization as possible. In many instances, CPU utilization can run at 100 percent for long periods of time without affecting a VM's service-level agreement. The CPU Ready percentage is the guiding metric for understanding whether this level of CPU utilization is possible.

CPU Ready is defined as the amount of time a VM is ready to run on a physical CPU but is unable to because all physical cores are busy. Most VMs can incur 10 percent of CPU Ready before their performance is degraded. Web servers and application servers generally fall into this category. Database servers and VoIP applications are more sensitive to CPU Ready and need to be as close to 0 percent as possible. Performance testing in these situations is the key to understanding the amount of overcommitment that is possible.

We also consider the number of cores that the physical CPUs have when measuring CPU Ready, as well as the average and maximum amount of vCPUs that are assigned to the VMs. When a VM has more vCPUs assigned to it than what is available on a single socket on the private cloud host, the VM has to access cores in another socket. This can increase the CPU Ready percentage for those VMs. If the majority of the environment consists of VMs that require eight or more vCPUs, we have found it advantageous to pair private cloud host servers with high-core-count CPUs so VMs can run within a single socket. High-core-count CPUs are also beneficial in dense environments that have VMs with small vCPU configurations, because they give VMs more opportunities to schedule with physical CPU cores without increasing the CPU Ready percentage.

In our experience, it is also critical to right-size VMs with the proper amount of vCPUs to maximize the performance of an individual VM and reduce the performance impact to other VMs and the private cloud host servers based on high-core count processors. Over-allocating vCPUs when they are not needed reduces performance and can lead to having to purchase additional private cloud host servers that are unnecessary.

How to Harness the Power of CPU Cores and Threads

In our experience, right-sizing our virtual machine (VM) CPU configurations generates considerable performance benefits. We configure VMs with only the amount of virtual CPUs necessary to reach a peak VM CPU utilization of 80 percent. On each VM and each host, we monitor the CPU Ready percentage so that it remains below 10 percent.

The physical-to-virtual CPU capacity planning ratio depends on the workloads that are running. Virtualized database servers sometimes suffer in performance if the physical-to-virtual ratio is too high. We work with our application owners to determine what the proper ratios are for our environment.

Active Memory, Page Sharing, and Ballooning

Understanding memory in the virtual world can be challenging. When a VM is configured with 16 GB of RAM and powered on, it is assumed that 16 GB of the private cloud host's RAM has been used and is not available for any other VMs. Configured memory is divided into two subcategories: allocated memory and free memory. Allocated memory is what is assigned to applications running on the VM, and it consists of two types: active memory and idle memory. Active memory is cloud host RAM that has been recently accessed or is currently in use by applications. The memory that has been allocated but not accessed or used for an extended period of time is called idle memory. Because active memory is one of our key performance metrics, we monitor it closely while reclaiming any idle or free memory for other VMs. We strive to achieve an active memory percentage of 70 percent. Many hypervisors have technologies to help achieve this.

Many hypervisors remove duplicate memory pages from a cloud host's physical memory. This function is known as page sharing. For example, if 10 applications are running on a cloud host, each VM has the same OS drivers loaded into memory, the same application files, and maybe even the same application pools. The hypervisor will keep only one instance of these files and drivers in memory, which delivers memory efficiencies.

We use a hypervisor that can retrieve idle memory from VMs; this process is called ballooning. Cloud hosts cannot determine how VMs are using their memory and the ballooning driver helps to resolve this. When a cloud host needs to reclaim idle memory, the ballooning driver inflates on a VM. This causes the VM's OS to swap out the idle memory, and then the hypervisor can reclaim it. The hypervisor calls the ballooning driver only in environments where cloud host memory is overcommitted. At Intel, we rely on the ballooning driver to reclaim unused memory from application owners that have not right-sized their VMs. In some of our busiest environments, we have achieved over 200-percent memory overcommitment without affecting VM and private cloud host stability. We have found that not all VMs and applications can tolerate memory overcommitment, so we test each application's level of sensitivity. We set memory reservations for VMs that have stability issues with overcommitment.

Server Virtualization BKMs

Before an application is virtualized, we conduct a baseline performance check on the physical hardware. We have found that some applications have special requirements for virtualization. For example, many collaboration and Java* virtual machine applications require special memory configuration settings. For instance, some Java virtual machines need extra memory for garbage collection, because they do not work well with memory sharing. Garbage collection that is slowed due to limited memory will affect application performance. Latency-sensitive applications, such as VoIP, video streaming, and instant messaging require reserved resources or they may not work. Database servers have unique virtual machine configuration requirements to maximize performance.

We use three primary methods to optimize virtual application performance:

- **Best-known methods (BKMs) published by hardware vendors.** We have found virtualization-specific vendor BKMs helpful. Storage vendor and server vendor BKMs can dramatically improve performance.
- **Judicious use of power options.** We disable power-saving options on private cloud host servers. Although these options save power, they also negatively affect virtualized application performance.
- **Mapping applications to VMs.** This process can be challenging, but it is important for our IT support teams. If there is an outage, the support team may not know which applications are affected or who to contact. We maintain an application catalog repository, which requires all our application owners to register their application.

Sample Results At a Glance

- Over a one week period, a workload of 5,000 VMs were tested on private cloud hosts.
- The average configured VM size in the tested environment was 2.6 vCPU.
- The average RAM configured per VM was 6.4 GB.

Results: Servers Based on 16-Core Intel® Xeon® Processor E5-2698 v3 Meet Our Business Needs

Table 1 shows the server configurations we tested and Table 2 describes the private cloud host requirements per 5,000 VMs in Intel's private cloud preproduction environment. We collected data about the number of hosts required to run the VMs, the compute-intensive density, and the memory-intensive density. We then calculated the VM density, average CPU per VM, and average memory per VM. Private cloud host density was calculated with a 10-percent to 15-percent increase or decrease based on CPU usage and CPU Ready metrics. We used the following calculation:

$$\frac{\texttt{average number of allocated VM CPUs per private cloud host}}{\texttt{average vCPU size per VM in the tested environment}}$$

Table 1. Server Configurations

	Intel® Xeon® Processor x5670	Intel® Xeon® Processor E5-2670	Intel® Xeon® Processor E5-2680 v2	Intel® Xeon® Processor E5-2698 v3
CPU sockets	2	2	2	2
Cores per CPU socket	6	8	12	16
Frequency	3.0 GHz	2.6 GHz	2.8 GHz	2.3 GHz
Cache	12 MB	30 MB	30 MB	40 MB
Memory per cloud host	96 GB	256 GB	256 GB	512 GB
Memory type	DDR2-400 MHz	DDR3-1333 MHz	DDR3-1600 MHz	DDR4-2133 MHz
Thermal design power	120W	115W	120W	135W
Maximum virtual machine (VM) density	30	63	96	131
CPU per VM	0.4	0.25	0.25	0.24
Memory per VM	3.2	4.06	2.67	3.91

Table 2. Study of 5,000 VMs Tested across Two Preproduction Clusters

	Intel® Xeon® Processor x5670	Intel® Xeon® Processor E5-2670	Intel® Xeon® Processor E5-2680 v2	Intel® Xeon® Processor E5-2698 v3
Hosts required	167	80	53	39
Compute intensive density	29	59	88	117
Memory intensive density	34	91	91	183
GHz per VM (compute)	-	-	0.73	0.68
Cache per VM	-	-	0.31	0.31

An analysis showed an 11:1 average core oversubscription. The average memory oversubscription was 180 percent. The average vCPUs configured per VM was 2.6. The average RAM configured per VM was 6.4 GB. Figure 1 shows that the Intel Xeon processor E5-2698 v3 used the lowest amount of energy. As shown in Figure 2, this same processor had the lowest four-year cost per VM. Servers based on the Intel Xeon processor E5-2698 v3 required only 39 hosts to support an average of 5,000 VMs in Intel's preproduction environment—the lowest total number of hosts of all tested processors.

We found that the hypervisor's distributed resource scheduler balanced the host's average CPU consumption. This indicated that the CPU is the primary metric for VM consolidation. The CPU Ready measurement was lower for high-core-count CPUs. We determined that the lower CPU Ready number was due to the VMs having greater opportunities for scheduling on processors with a higher core count.

Our study showed that private cloud hosts configured with two Intel Xeon processors E5-2698 v3 and 256 GB of memory are the optimal configuration for Intel's Office and Enterprise environments. As shown in Figure 3, using servers based on these Intel Xeon processors enabled us to host the same amount of VMs on a significantly smaller number of servers than other servers with lower-core-count processors, thereby also reducing our licensing costs (see Figure 4).

We previously assumed that new server purchases were primarily dictated by memory requirements. However, we discovered it is possible to overcommit memory to over 200 percent without degrading performance on our VMs. For virtualized workloads, servers with a higher core count performed best with memory overcommitment. In our studies using a sample workload of 5,000 VMs, the Intel Xeon processor E5-2698 v3 showed the lowest four-year TCO. Based on our new findings, we have shifted to cost-per-VM as the metric for determining when to refresh our private cloud hosts.

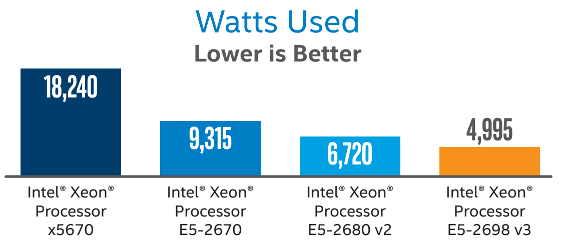

Figure 1. We found that the Intel® Xeon® processor E5-2698 v3 used the lowest amount of energy. Test study using 5,000 VMs. Intel internal measurements, August 2015.

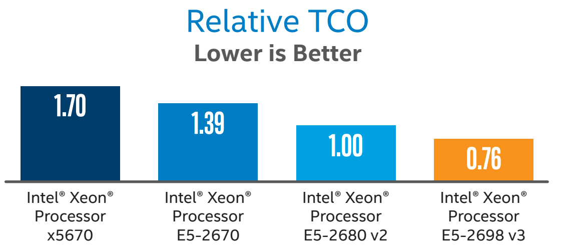

Figure 2. Intel® Xeon® processor E5-2698 v3 had the lowest four-year total cost of ownership (TCO). Test study using 5,000 VMs. Intel internal measurements, August 2015.

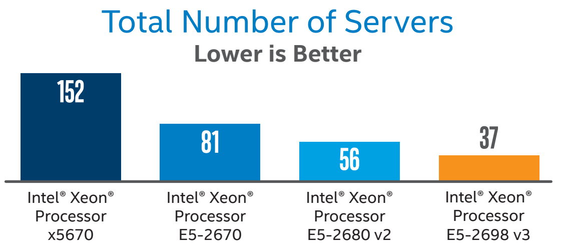

Figure 3. We reduced the number of servers required to host virtual machines by using the Intel® Xeon® processor E5-2698 v3. Test study of required servers based on 5,000 VMs. Intel internal measurements, August 2015.

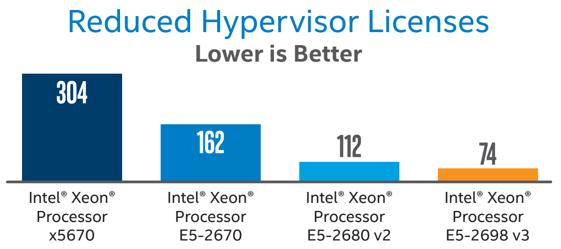

Figure 4. The number of hypervisor licenses decreases using Intel® Xeon® processor E5-2698 v3, contributing to a lower operating cost. Test study using 5,000 VMs. Intel internal measurements, August 2015.

Conclusion

In our study, servers based on the Intel Xeon processor E5-2698 v3 were the most efficient for running applications in VMs on private cloud hosts. The analysis demonstrated that servers with high-core-count processors had the lowest four-year cost per VM. This was due to the growth in the number of cores, better architecture, and increased cache size compared to previous generations of processors.

We also expect servers based on the Intel Xeon processor E5-2698 v3 to help control operational and software licensing costs by greater virtualization density, thus requiring fewer servers than were necessary with previous-generation processors. The dollar value per VM proved so significant that cost per VM now drives our server refresh.

From our analysis, using servers based on high-core-count processors provide the following benefits:

- Substantial private cloud pool hosting and virtualization performance throughput increase for a modest increase in server cost
- Higher performance throughput for a given TCO
- Improved CPU performance of virtualized workloads

Based on these performance throughput and TCO advantages, Intel IT has standardized on two-socket servers with Intel Xeon processors with 16 cores for private cloud application virtualization needs. By doing so, we expect to achieve greater private cloud hosting performance while realizing operational benefits such as cost avoidance of data center construction and reduced power consumption.

Our results suggest that other technical applications with intensive CPU demand, such as simulation and verification applications in the auto, aeronautical, oil and gas, and life sciences industries, could see similar improvements, depending on the workload characteristics.

For more information on Intel IT best practices, visit **intel.com/IT**.

Receive objective and personalized advice from unbiased professionals at **advisors.intel.com**. Fill out a simple form and one of our experienced experts will contact you within 5 business days.

IT@Intel

We connect IT professionals with their IT peers inside Intel. Our IT department solves some of today's most demanding and complex technology issues, and we want to share these lessons directly with our fellow IT professionals in an open peer-to-peer forum.

Our goal is simple: improve efficiency throughout the organization and enhance the business value of IT investments.

Follow us and join the conversation:

- Twitter
- #IntelIT
- LinkedIn
- IT Center Community

Visit us today at **intel.com/IT** or contact your local Intel representative if you would like to learn more.

Related Content

Visit **intel.com/IT** to find content on related topics:

- Intel IT's Data Center Strategy for Business Transformation paper
- Cloud Computing Cost: Saving with a Hybrid Model paper
- Developing a Highly Available, Dynamic Hybrid Cloud Environment paper
- Improving Business Continuity with Data Center Capacity Planning paper
- Making Private-Public Cloud Decisions on the Way to a Hybrid Cloud paper
- SaaS Security BKMs for Minimizing Risk in the Cloud paper
- Simplifying the Path for Building an Enterprise Private Cloud paper

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families: Learn About Intel® Processor Numbers.

THE INFORMATION PROVIDED IN THIS PAPER IS INTENDED TO BE GENERAL IN NATURE AND IS NOT SPECIFIC GUIDANCE. RECOMMENDATIONS (INCLUDING POTENTIAL COST SAVINGS) ARE BASED UPON INTEL'S EXPERIENCE AND ARE ESTIMATES ONLY. INTEL DOES NOT GUARANTEE OR WARRANT OTHERS WILL OBTAIN SIMILAR RESULTS.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS AND SERVICES. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS AND SERVICES INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Printed in USA ♻ Please Recycle 0616/LMIN/KC/PDF