The impact of workload on energy efficiency of virtualized systems

Jukka Kommeri
Helsinki Institute of Physics,
Technology program, CERN,
CH-1211 Geneva 23, Switzerland
jukka.kommeri@cern.ch

Tapio Niemi
Helsinki Institute of Physics,
Technology program, CERN,
CH-1211Geneva 23, Switzerland
tapio.niemi@cern.ch

Olli Helin
Helsinki Institute of Physics,
Technology program, CERN,
CH-1211 Geneva 23, Switzerland
olli.helin@cern.ch

***Abstract*—Virtualization, i.e., running several virtual computers on the same physical hardware, is an essential technology in data centers. Since demand for cloud computing services is constantly growing, an increasing number of data centers are focusing on improving their energy efficiency. This has made energy efficiency of virtualization technologies an important research domain. So far, maximizing performance of virtualization technologies has received a lot of attention in cloud computing industry and several academic studies on performance optimization can be found, too. However, these studies usually focus on improving energy efficiency by applying server consolidation methods. In this paper we focus on energy efficiency of virtualization technologies, i.e., how a virtual service can be made more energy efficient. Our aim is to reduce energy consumption without decreasing the quality of service. We have studied this by performing a large set of measurements with different system settings. We used both synthetic benchmarks and real applications. We found out that energy efficiency depends on 1) the workload of the virtual servers, and 2) the number of virtual servers on the physical server. We noticed that it is more energy efficient to maximize workload of virtual servers and to minimize their number. Additionally, we observed that properly configured idle virtual servers hardly increase energy consumption. Thus, our conclusion is that it is better to load virtual servers heavily or let them run idle.**

***Keywords*-virtualization; energy-efficiency; server consolidation; xen; kvm; invenio; cmssw**

I. INTRODUCTION

The work presented in this paper is based on our earlier work, that was published in ENERGY 2012 conference [1], and partially also on work, that was published in ICGREEN 2012 conference [2]. The current paper contains enhanced background discussion and more detailed analysis on results and also presents some new results such as latency measurements.

Web based applications have gained popularity and an increasing number of these applications are hosted by cloud computing in large data centers containing thousands of virtualized servers [3], [4]. Traditionally, a server has been purchased to host only one service (e.g., a web server, a DNS server). This is not very efficient, since according to many studies the average utilization rate of a physical server hosting a web site is around 15% of maximum but depends a lot on the service and it can be even as low as 5% [5], [6].

This level of utilization is very low compared to any field in industry. A common explanation for the low utilization is that data centers are build to manage peak loads. However, this is not a new data center specific issue, since high peak loads are common in many other fields. Even with this low level of utilization the servers are usually operational and consuming around 60% of their peak power [7]. Low utilization level is inefficient through the increased impact on infrastructure, maintenance and hardware costs. For example, low utilization reduces the efficiency of power supplies [8] causing over 10% losses in power distribution. Thus, servers should run in near full power when they do value adding work, because then they operate most efficiently considering consumed energy per executed task [5].

Scientific computing clusters at CERN have traditionally allocated resources for one analysis job such that the job gets one computing core and 2 GB of memory. As the number of cores in the CPU and the number of CPUs in the server increase, more jobs can be processed in parallel by the server. Modern servers for scientific computing can have 16 cores per CPU, two to four CPUs and hundreds of gigabytes of memory. Combining this with the need for different analysis environments, computing resources should be divided into smaller logical units.

Server consolidation by using virtualization technologies is a solution for increasing utilization, since it allows one to combine several services into one physical server. In this way, these technologies make it possible to take better advantage of hardware resources. Virtualization makes it possible to create logical containers, virtual machines, that contain a complete operating system with a user specific analysis environment. This virtual machine can be modified to meet users resource requirements and moved between physical machines to improve the total energy efficiency of a larger server cluster or computing center [9].

In this study, we focus on energy efficiency of different virtualization technologies. Our aim is to help the system administrator to decide how services should be consolidated to minimize energy consumption without violating the quality of service agreements. Virtualization in the context of this paper refers to system virtualization where several operating system instances are run on single physical hardware, as depicted in Figure 1.

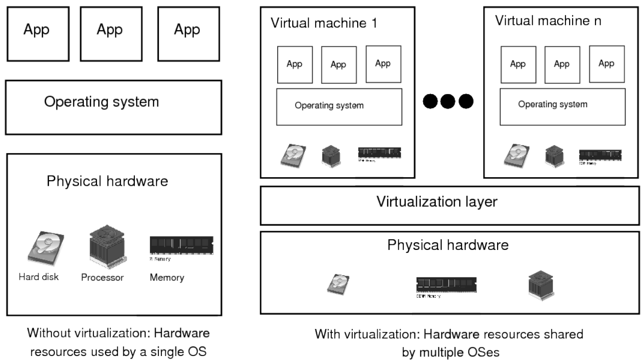

Figure 1. A non-virtualized and a virtualized system

We studied energy consumption of virtualized servers with two open source virtualization solutions; KVM and Xen. They were tested both under heavy load and being idle. Several synthetic tests were used to measure the overhead of virtualization on different server components. Also two realistic test applications were used: 1) the Invenio catalog program's database service, and 2) CMSSW, the CMS Software Framework, a physics analysis software for the data generated by the Compact Muon Solenoid, CMS, experiment at CERN. The results were compared with the results of the same tests run directly on hardware without any virtualization layer. We also studied how overhead of virtualization develops by sharing resources of physical machines equally among different number of virtual machines and running the same test set on each virtual machine set.

The paper has been organised as follows. After introduction, we review the related work in Section II. Our test environment and tests are explained in Section III and their results given in Section IV. Finally, conclusions are given in Section V.

II. Related work

Virtualization and its performance is a well-studied area. Previous studies mainly focus on performance, isolation, and scheduling. Even though energy efficiency is one of the main reasons for server consolidation and virtualization, it has not received much attention. Instead, many of the existing studies evaluate overhead differences between different virtualization solutions and how virtual resource could be provisioned between physical servers in an energy-efficient way.

Virtualization technologies are a key component of cloud computing [10]. Large data centers host cloud applications on thousands of servers [3], [4]. In such environments, the benefits of virtualization are obvious. Xu et al. [11] mention just-in-time compute and storage capacity, reducing management and administration cost through automation and providing greater control over end-user service levels.

Virtualization of the high energy physics grid computing clusters has been studied by many researchers. Fenn et al. [12] have tested high performance applications (HPC) in clusters that are made of virtual machines. They found KVM to be usable in non I/O intensive loads. There has been many improvements to the KVM I/O since, and nowadays there is a paravirtualized driver for KVM network and disk, improving I/O performance significantly.

Regola et al. studied the use of virtualization in high performance computing (HPC) [13]. They believed that virtualization and the ability to run heterogeneous environments on the same hardware would make HPC more accessible to a bigger scientific community. They concluded that the I/O performance of full virtualization or para-virtualization is not yet good enough for low latency and high throughput applications such as MPI applications.

Nussbaum et al. [14] made another study on the suitability of virtualization on HPC. They evaluated both KVM and Xen in a cluster of 32 servers with HPC Challenge benchmarks. These studies did not find a clear winner but the authors were able to conclude that the performance of full virtulization is far behind that of paravirtualization. Moreover, running workload among different number of virtual machines did not seem to have an effect. Verma et al. [15], [9] have also studied virtualizaton of HPC applications. They focused on power aware dynamic placement of virtual machines between physical hosts. Though these tests were made with low memory footprint applications, they demonstrated the benefits of virtualization on energy-efficiency. A similar paper by Lui et al. [16] studied the cost of moving virtual machines between physical machines and modelled the energy consumption of virtual machine replacament. The study showed that the cost of migration, meaning the movement of virtual machines between physical hosts, depends mainly on three things; application memory usage, application memory footprint, and network speed.

Padala et al. [17] carried out a performance study of virtualization. They studied the effect of server load and virtual machine count on multi tier application performance. They found OS virtualization to perform much better than paravirtualization. The overhead of paravirtualization is explained by L2 cache misses, which in the case of paravirtualization increased more rapidly when load increased.

Another study from Deshane et al. [18] compares scalability and isolation of a paravirtualized Xen and a full virtualized KVM server. Results said that Xen performs significantly better than KVM both in isolation and CPU performance. Surprisingly, a non-paravirtual system outperforms Xen in I/O performance test.

Virtualization in multi processor environment was studied by Petrides et al. [19]. They found out that virtualization can be used to stabilize performance for HPC load when executing multiple threads in multi processor environment. In their study the possibility of binding processes and threads to certain cores or processors on the operating system level has not been studied. Nevertheless the study shows how virtualization can improve performance of a multicore and multiuser environment.

As we can see from previous studies, the topic of this paper, the energy-efficiency of virtualization on a single server has not yet received much attention among existing studies.

III. Tests and Test Environment

Our tests aimed at measuring the energy consumption and overhead of virtualization with a diverse test set. We used both synthetic and real applications in our tests and measured how performance is affected by virtualization. We compared the results of the measurements, that were done on virtual machines, with the results of the same tests on physical hardware. We started by measuring the idle consumption of virtualized machines using different number of virtual machines. After that, we compared different virtualization technologies and operating systems.

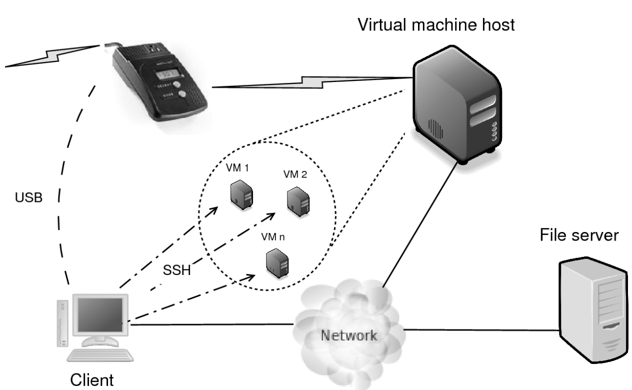

Figure 2. Test environment

For running the tests and collecting measurements, we have used a separate client machine that is connected to the test servers with gigabit local area network. Figure 2 shows the test environment. Cient machine controls how many virtual machines are used and how many applications are run in parallel on the virtual machine host or virtual machines. The client both starts the tests and collects energy meter values. Power usage data was collected with a Watts up? PRO meter via a USB cable. Power usage values were recorded every second.

A. Test Hardware

In our tests we used diverse server hardware. We had both dual CPU servers and single CPU servers. For the Dell 210 II single processor server we had two different types of processors. These two processors represent two different approaches; the E31260L is energy-efficient, while the E31280 is for higher performance. This collection of different types of servers and processors allowed us to study the effect of processor and server architecture more thoroughly.

In our test we used following servers:

- Dell PowerEdge R410, Intel Xeon E5520 w/o Hypertheading, 16 GB memory, 250 GB hard disk
- 2CPU 12 core server, Opteron 2427, 32GB 800MHz memory, 1TB hard disk
- Dell Poweredge R210, Xeon X3430, 8GB 1333MHz DDR3, 250GB hard disk
- Dell Poweredge R210 II, Xeon E31260L, 8GB 1333MHz DDR3, 1TB hard disk
- Dell Poweredge R210 II, Xeon E31280, 8GB 1333MHz DDR3, 1TB hard disk

B. Used Virtualization Technologies

The operating system used in all machines, virtual or real, was a standard installation of 64-bit Ubuntu Server 10.04.3 LTS. The same virtual machine image was used with both KVM and Xen guests. The image was stored in a raw format, i.e., a plain binary image of the disk image. Linux 3.0.0 kernel was chosen as it had the full Xen hypervisor support. With this kernel we were able to compare Xen with KVM without a possible effect of different kernels on performance.

For CMSSW tests and idle tests, a virtual machine with Scientific Linux 5 was installed with CMSSW version 4.2.4. For these tests real data files produced by the CMS experiment were used. These data files were stored on a Dell PowerEdge T710 server and shared to the virtual machines with a network file system, NFSv4.

C. Test Applications

Our synthetic test collection consisted of Linpack [20], BurnInSSE [1], Bonnie++ [21] and Iperf [22]. Processor performance was measured with an optimized 64-bit Linpack test. This benchmark was run in sets of thirty consecutive runs and power usage was measured for whole sets. In addition, processor power consumption measurements were conducted with ten minute burn-in runs with 64-bit BurnInSSE collection using one, two and four threads. Disk input and output performance were measured using Bonnie++ 1.96. The number of files for a small file creation test was 400. For a large file test the file size was 4 GB. For Bonnie++ tests, the amount of host operating system memory was limited to 2.5 GB with a kernel parameter and

[1] http://www.roylongbottom.org.uk

the amount of guest operating system memory was limited to 2 GB. For hardware tests, a kernel limit of 2 GB was used. The tests were carried out ten times. Network performance was measured using Iperf 2.0.5. Three kinds of tests were run: one where the test computer acted as a server, another where it was the client and a third where the computer did a loopback test within itself. Testing was done using four threads and a ten minute time span. All three types of tests were carried out five times.

As real world applications, we used two different systems. The first one was based on the Invenio document repository [23]. We used an existing Invenio installation, connected to copy of a large bibliogaphic database called Inspire. The Invenio document repository software suite was v0.99.2. The document repository was run on an Apache 2.2.3 web server and MySQL 5.0.77 database management system. All this software were run on Scientific Linux CERN 5.6 inside a chroot environment. Another server was used to send HTTP requests to our test server. The requests were based on an anonymous version of a real-life log data of the identical document repository in use at CERN. The requests were sent using the Httperf web server performance test application [24].

Table I shows the rates and resources given to virtual machines in the Invenio tests. The MaxClients setting refers to the maximum clients setting in Apache web server configuration.

Table I
SETTINGS FOR CHANGING LOAD AND RESOURCES OF A SINGLE VIRTUAL MACHINE

VCPUs	Memory (GB)	MaxClients	Request rate
2	5	8	5
4	8	15	10
6	15	24	15

The second real application was a physics data analysis that used the CMSSW framework [25]. This analysis task is a typical one in high-energy physics. We used real data created at CERN. The data was stored in a ROOT image[26] files, which our case were of size 4GB. Normally, a data analysis with this data takes days to perform, thus we limited the number of events of one analysis task to 300. With this limitation the analysis takes 10 minutes on the Opteron hardware. The data was located on network file system, NFS, and reading it caused very little network traffic, 2kB per task.

Tables II and III show how the resources of the 12-core Opteron server were shared between virtual machines when testing the constant load with different number of virtual machines running the CMSSW tests. In all the cases the hardware resources were shared equivalently among different virtual machines.

Table II
SETTINGS FOR A SINGLE VIRTUAL MACHINE IN 12-CORE OPTERON SERVER

VM count	VCPUs	Memory (GB)
1	12	31.5
4	3	7.88
8	2	3.94
10	2	3.15
12	1	2.6

Table III
SETTINGS FOR A SINGLE VIRTUAL MACHINE IN 4-CORE DELL 210 II

VM count	VCPUs	Memory (GB)
1	8	7.5
2	4	3.75
3	3	2.50
4	2	1.88
5	2	1.50
6	1	1.25

IV. RESULTS

A. Idle consumption

First we studied idle energy consumption with different virtualization solutions and with different number of virtual machines. Figure 3 shows the power consumption of two different virtualization solutions, and the power consumption of the hardware with no virtual machines. In this test both the host system and the virtual machines were running only basic operating system functions without any analysis tasks. In all the measurements with virtual machines we had three virtual machines running idle. The figure shows how energy consumption of two different virtualization solutions behave when the servers are idle. It shows how overhead of virtualization depends on the virtualization solution and kernel version. The difference between KVM and hardware is less than 3%, which is already a big improvement compared to three separate physical machines running idle. This test was run with the Dell R410 server.

The second idle measurement was run on the dual processor Opteron server. The test measured the energy consumption of an idle physical hardware for 20 minutes. Figure 4 shows how the operating system affects the idle consumption. The same test was repeated with different number of virtual machines on the same physical hardware. The energy consumption accumulates with the virtual machine count with Scientific Linux 5 (SLC5) but with Ubuntu it remains almost the same as with bare hardware. This test shows that the choice of virtual machine has big effect on energy-efficiency. It also shows that an idle virtual machine can have a very low energy consumption.

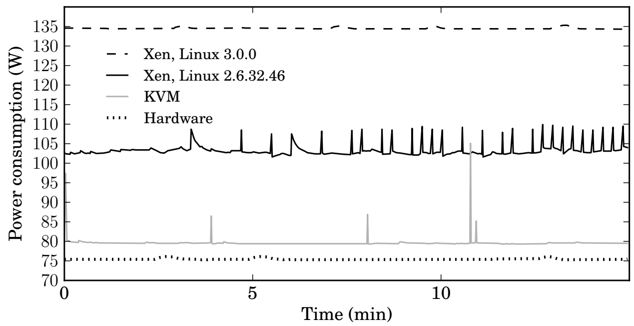

Figure 3. Typical idle power consumption

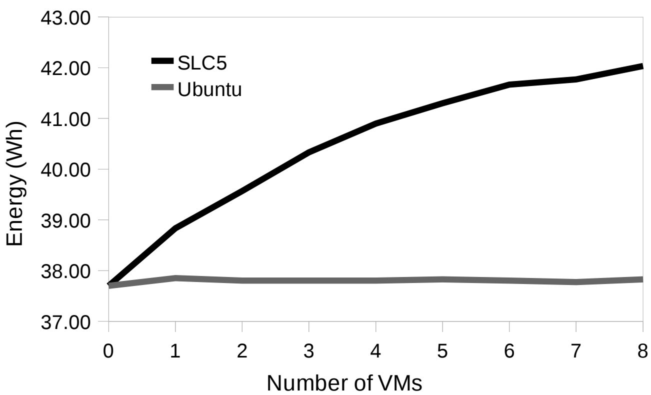

Figure 4. Energy consumption of idle virtual machines

B. Synthetic tests

We used synthetic tests to stress different server components; CPU, I/O and network. With these tests we studied in which situations virtualization causes the most overhead. First we tested the overhead of disk reads and writes with Bonnie++. The consumption of the virtualization solutions; Xen and KVM was compared to hardware consumption.

In images from 5 to 12 the y-axis represents the percentual difference to corresponding measurents done without virtualization.

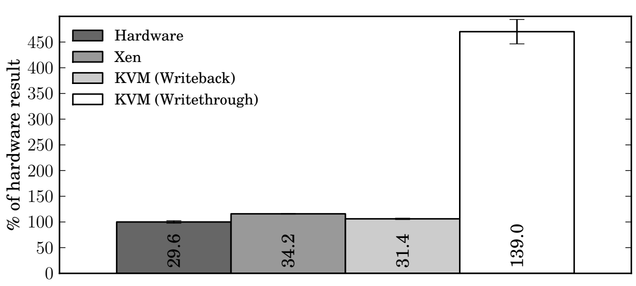

Figure 5. Energy consumption of Bonnie++ (Wh)

As can be seen from Figure 5, when running a set of synthetic disk operations Xen uses slightly more energy compared to hardware. With KVM the situation is different. When using the default cache setting, write through cache, KVM uses around 350% more energy than hardware. About 90% of the test time is spent doing the small file test part of Bonnie++. Switching to write back cache, results of KVM are actually slightly better than hardware results. Write back cache writes only to a cache and stores data to the disk only just before the cache is replaced. This is a cache mode that is not safe for production use and is available mainly for testing purposes.

Next, we tested the overhead of virtualization of CPU with two different benchmarks; BurninSSE and Linpack. Figure 6 shows the power consumption of the server while running BurninSSE on a non-virtualized server and on virtual machines. We compared the technologies by introducing CPU load with 1 and 4 threads of BurnInSSE. With 1 thread, KVM and hardware use the same amount of power, but Xen uses around 10% more. With 4 threads the situation is the other way around: Xen uses less power than KVM and hardware. The explanation can be seen in Figure 7. Even though Xen uses more power in the single-threaded LINPACK test, it is slower: the CPU is not running at its full turbo boosted speed, but Xen has a systematic overhead in power consumption compared to the others. With 4 threads, Xen's CPUs are not running at full speed so the power usage is not as great as with hardware or KVM, and the overhead in power consumption is overshadowed by the power usage of 4 computing threads.

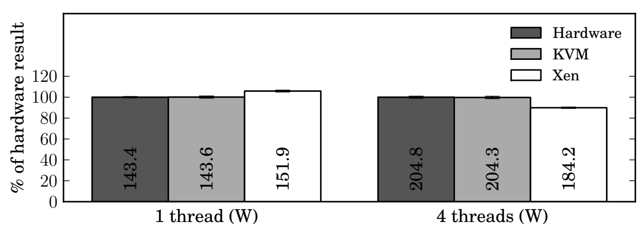

Figure 6. Power consumption under high CPU load with BurnInSSE

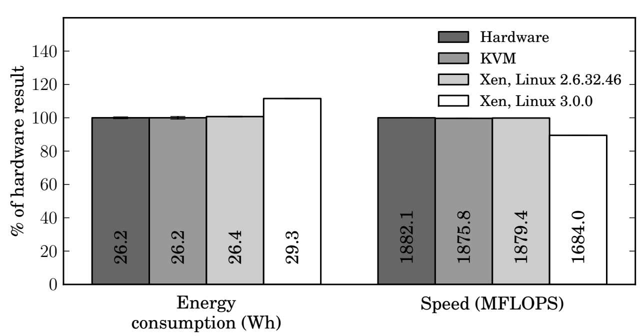

Figure 7. Energy consumption under high CPU load with Linpack

As the last server component we tested the network and the overhead of virtualization to the power consumption of the network. To stress the network we used Iperf network benchmark in dual mode, where the traffic is tested to both directions. The power consumption of Iperf test results are shown in Figure 8. Direction of the traffic does not have an effect on the results, which show a similar trend for I/O and CPU tests: KVM uses slightly more power than hardware while Xen consequently uses slightly more power than KVM. Interestingly, when a Xen virtual machine was running as server it used slightly more power than when running as client. With KVM and hardware it was the other way around. In the loopback mode, one can find similar results with Xen as in the LINPACK test in Figure 7: for some reason, Xen's performance is capped and consequently bandwidth in the loopback mode is much worse than with KVM or hardware, and on the other hand mean power consumption is lower.

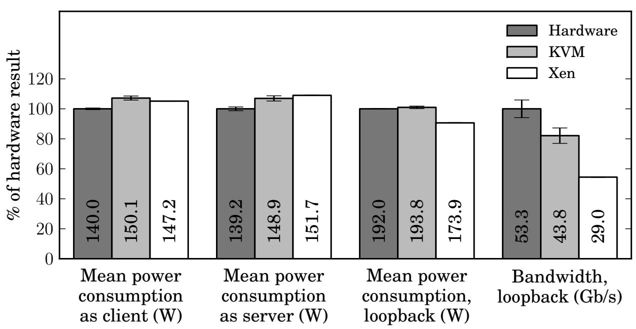

Figure 8. Power consumption under Iperf network traffic test

C. Realistic load

Realistic tests were designed such that we would get better understanding of energy usage in two different real world situations: web services and physics analysis. Both benefit from virtualization differently as they use server resources in a different way. Web server based systems benefit from being able to combine idle services into single physical server and the overhead of virtualization is not so critical. The physics analysis benefits from an isolated and job specific environment provided by virtualization, but this causes it to run a longer time.

We started our realistic tests with the Invenio CERN document server repository case. In this test, we sent HTTP requests, which were based on CERN library log data, to a virtualized web server. We measured both performance and power consumption. We ran the same tests with and without virtualization. We compared two virtualization solutions to hardware to measure the overhead of virtualization. In all the Invenio tests, the Invenio installation was in a chroot environment with a complete SLC5 installation. To assure that chroot between the operating system and the Invenio web application did not have any negative effects on test results, a comparative test was performed between the base system and another chroot environment using a copy of the base system as the new root.

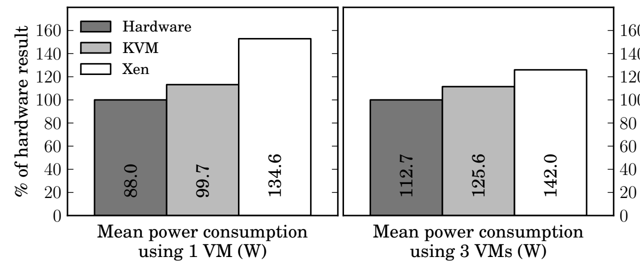

Figure 9. Power consumption of different virtualization solutions with different number of virtual machines in the repository test

Both the amount of virtual machines and virtualization technology have an impact on the energy-efficiency. This effect was tested here by running Invenio document server in different number of virtual machines and with different virtualization technologies. Figure 9 shows how the power consumption varies between different virtualization technologies. On the left side we have the results of running one virtual machine with a rate of 5 queries per second workload and on the right side 3 virtual machines with a rate 5 queries per second per virtual machine and total request rate of 15 queries per second. Figure shows that the power consumption evens out when we have more virtual machines and load.

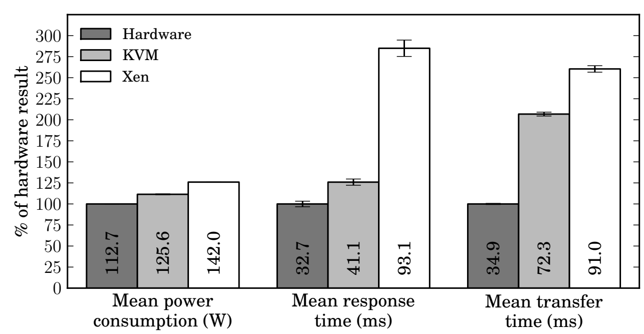

Figure 10. Power consumption and Httperf results of different virtualization solutions

Figure 10 illustrates the effect of virtualization on the web performance when running Invenio on three virtual machines with request rate of 5 on each. In the figure we have both the energy consumption from the previous figure and the results from the Httperf test. One can see that even though there is not much overhead on energy consumption there is a bigger impact to the quality of service as the response times and transfer times increase.

Previous results showed us that KVM performed closer to hardware level and proved to be more interesting for

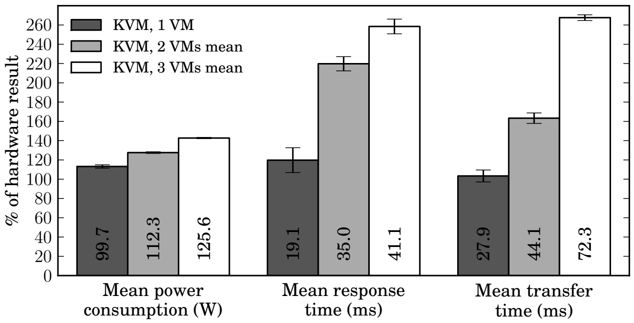

Figure 11. The effect of workload on virtual machine performance with KVM using different amounts of virtual machines

further studies. KVM was used in our test where we studied the overhead of virtualization by increasing the number of virtual machines with similar workload. These results are illustrated in Figure 11, showing that the power consumption increased linearly as a function of virtual machines. In the case of 3 virtual machines the total consumption decreases of 47%, but on the other hand the response times and transfer times increase more than 250%.

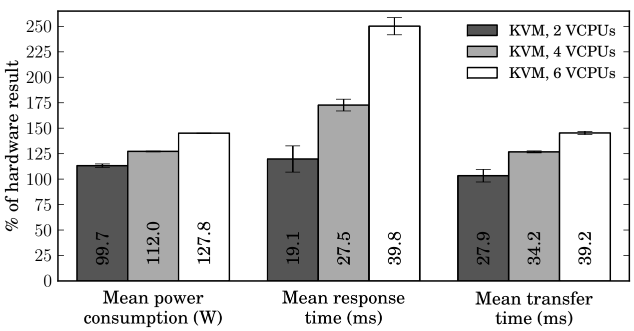

Figure 12. The effect of workload on virtual machine performance with KVM using different virtual machine resources

Virtual machines share server resources and the share of resources given to a virtual machine is configurable. One can either stretch the physical resources thin between several virtual machines or one can create a few virtual machines with more resources. Table I shows how the resources were shared. Figure 12 illustrates how the performance and energy-efficiency is affected when we increase the resources and load of a single virtual machine. Here you can see that the power consumption grows almost the same way as in the Figure 11. Difference being that increasing the resources of one virtual machine seems to improve the performance.

To illustrate the effect of virtualization on quality of service we have Figures 13 and 14. These figures show a cumulative distribution of response times that the Httperf test application reported for the HTTP requests. The distribution shows how the response times behave with different virtualization technology and load.

In Figure 13, we have the distribution of response times from a test with 3 virtual machines and total request rate of 15 requests per second. This distribution corresponds to the results illustrated by the right side of Figure 9 and Figure 10. One can see that both KVM and Xen decrease quality of service, but still more than 95% of requests are served in 100ms.

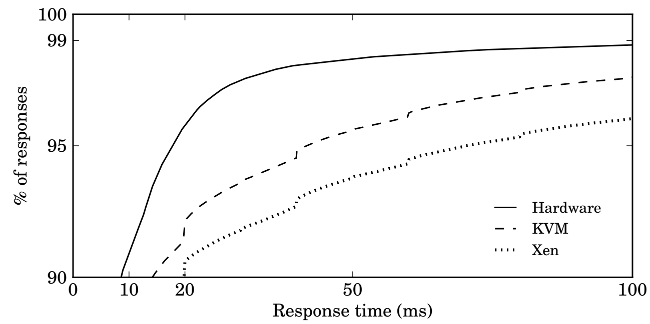

Figure 13. The impact of virtualization solution on quality of service

To show how virtual machine count affect the quality of service, we made a similar distribution from the test that was illustrated by Figure 11. Figure 14 shoes how the virtualization overhead effect on different workload and number of virtual machines. We compared KVM and one to three virtual machines with corresponding rates 5, 10, and 15 requests per second. Every additional virtual machine decreased the quality of service.

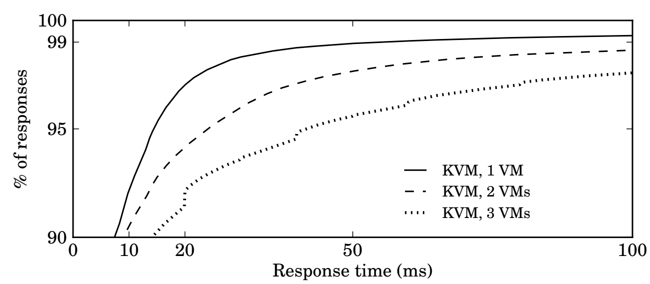

Figure 14. The impact of different loads on quality of service

As our second realistic load, we had a physics analysis application, CMSSW. First we tested how the number of virtual machines affects the performance of CMSSW. In the following tests we consider one run of the test application as a job. In Figure 15, we have the results of running 15 jobs in 5 different virtual machine sets and also on hardware. The figure shows how the energy efficiency degrades as the number of virtual machines increases and, at the same time, throughput decreases. 15 smaller virtual machines running one job are 6.8 times less energy efficient than running 15 jobs on one bigger virtual machine.

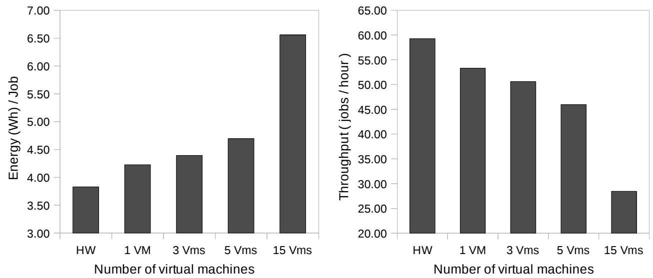

Figure 15. Running 15 jobs in different number of virtual machines

Next, we tested the effect of dual processor architecture on the overall performance. We run the same tests both on 2 CPU 12 core Opteron server and on single processor quad core R210 server. This test differs from the previous not only by the hardware, but also by the load introduced. Here we started with a very low load, that was increased to see how the overhead behaves on lower load. Figures 16 and 17 show the energy consumption and throughput from a test with different amount virtual machines running one job each. One can see that the virtualization introduces some overhead on both servers and this overhead increases as the amount virtual machines is increased. Single CPU server's performance is limited by the 8GB memory as a single CMSSW analysis job together with the operating system use approximately 1.2GB of memory. Figure 16 shows the results from both virtualized servers and physical hardware.

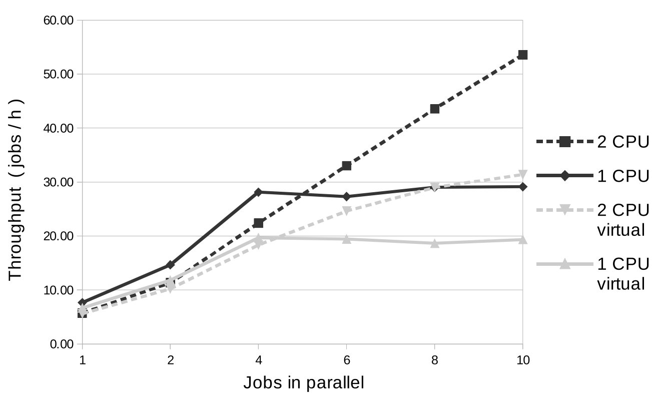

Figure 16. Throughput with different number of virtual machines running one job each

As shown in Figure 17, the single processor server has a better energy-efficiency on low loads, but this balances when the load is increased. One thing to notice is how static the overhead is on single processor server. In two processor server overhead increases as a function of virtual machines.

In the previous tests, we used light load on virtual machines, but used them in high numbers. Now we show how physics analysis job behaved in different sized virtual machines. As the Invenio tests showed, bigger virtual machines

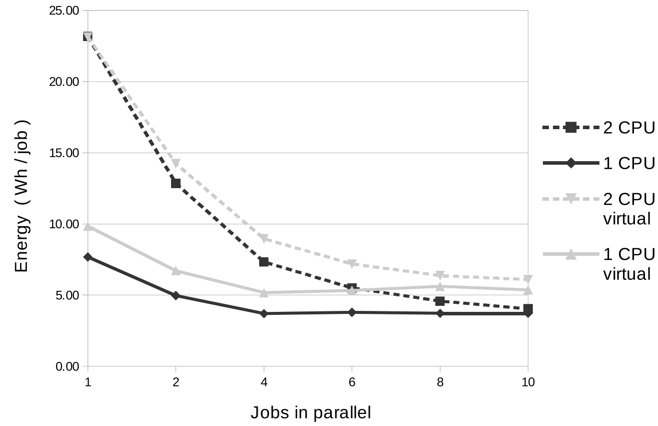

Figure 17. Energy consumption per job with different number of virtual machines running one job each

perform better. Figure 18 shows the effect of workload on energy-efficiency with physics analysis software. We tested different workloads on 5 identical virtual machines sharing the Opteron 12 core server. One can see that the energy-efficiency and throughput improve as we increase the load. This is in line with our earlier studies where we noticed that the commonly used one job per CPU core does not give the best performance or energy efficiency [27], [28]. Here we tested how it applies to virtualized environments.

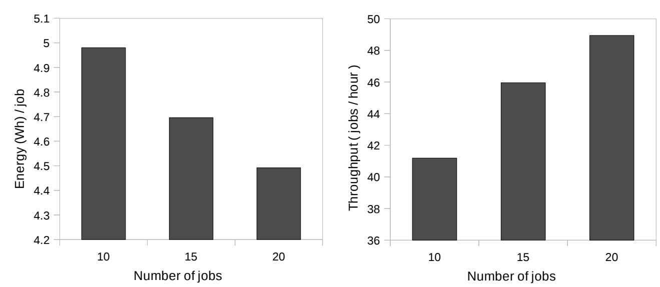

Figure 18. Running different workload on 5 virtual machines

As the number of virtual machines and their size seem to make a difference on dual processor server we repeated the test on single processor server. Figures 19 and 20 illustrate the effect of parallelism with rising load. Here we have run different amount of jobs on one physical server and divided the load equally between one, two, four and six virtual machines. One can see that the performance increases when the load is increased up to a point where the system throughput levels and eventually decreases. One can notice that the amount of virtual machines has big effect on the throughput and energy-consumption. The virtualization overhead increases exponentially as the function of virtual machine count and increases even more when the number of virtual machines is more than the number of cores in the server's processor.

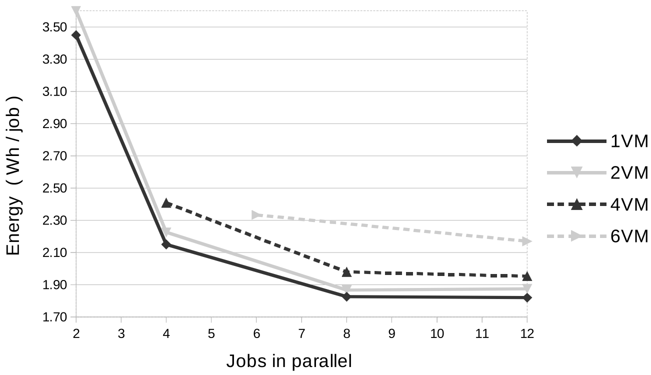

Figure 19. Energy consumption per job in virtual machines when varying resources and job parallelism

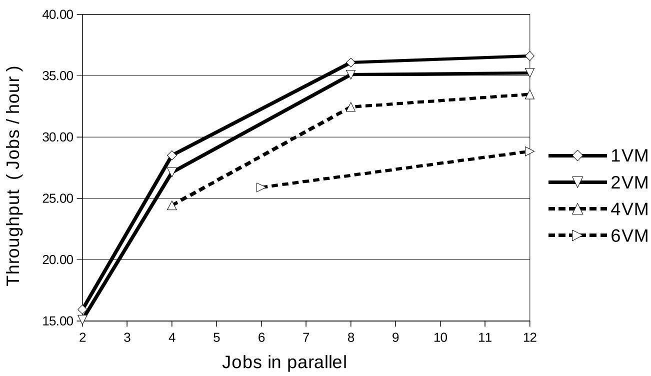

Figure 20. Job throughput of virtual machines with varying resources and job parallelism

These single processor tests above were performed with two types of CPUs; energy-efficient and powerful ones. Results shown in Figures 19 and 20 are from tests run with energy-efficient processor. The results from the powerful processor were similar to those of the energy efficient processor and produced similar figures. The difference was the energy-efficient processor used 17% less energy per job and the throughput of the powerful processor 40% better when comparing the minimum of energy consumption and the maximum of throughput.

We ran an additional test in parallel with the physics tests to study the reason behind the overhead. We tested how the network performance is affected by the increasing load and increasing number of virtual machines. This was done by running a ping command from the virtual machines towards the client machine. This was tested by running 12 jobs equally among different amount of virtual machines; 1,4,6 and 12. We noticed that the latencies did not drop while adding more virtual machines. This test showed that the resource sharing between virtual machines was fair and latencies varied very little between virtual machines.

V. Conclusions

Virtualization technologies develop at fast pace. New technologies arise and better interfaces are made to improve the usability of virtualization. In this study we have used two mature open source virtualization solutions; KVM and Xen. The performance of Xen and paravirtualization have been good for a long time, but for the version used in our tests suffered from the early adoption on Linux kernel and had not had enough time to mature in the vanilla Linux kernel. KVM on the other hand have come far from its early versions and proves comparable with the commercial virtualization solutions. Even though the technologies in this study were compared and tested against each other this should not be considered as a comparison between different virtualization technologies, but as a study on virtualization technologies in general. The performance balance between different technologies varies constantly, but the main idea is that resources are shared among multiple systems and this causes overhead to applications inside virtualized servers, which needs be taken into consideration.

The overhead of virtualization is a well-known fact and reported in many publications. Although the technologies have been improving a lot during the past five years, the performance of a virtualized system is still far from the hardware level. However, this does not mean that virtualization could not be useful in improving energy-efficiency in large data centers but it means that one should know how to apply this technology to achieve savings in energy consumption.

We studied the energy-efficiency of virtualization technologies and how different loads affect it. Our research indicates that idle power consumption of a virtualized server is close to zero. However, this depends a lot on the operating system running on the virtual machine, but it is always a small number compared to idle energy consumption of a physical server. Our study also indicates that virtualization overhead has great impact on energy-efficiency. This means that it would make more sense to share the physical resources among few virtual machines with heavy load instead of a larger set of light-loaded ones. Pure CPU-load in larger virtual machine groups does not seem to impose as much overhead as the more complex physics analysis job, that both requires network connectivity and disk storage. The physical core count also seem to pose a limit for the virtual machine pool size.

Acknowledgment

Many thanks to Jochen Ott from CMS@CERN experiment for providing help in installing the CMSSW and providing with a typical analysis job. Also we would like to thank Salvatore Mele, Tibor Simko, Jean-Yves LeMeur of CERN library and Invenio developers, for providing realistic data and a test case for our analysis; and Marko Niinimaki for comments.

REFERENCES

[1] J. Kommeri, T. Niemi, and O. Helin, "Study of virtualization energy-efficiency in high energy physics computing," in *Proc. Energy 2012*, 2012.

[2] J. Kommeri, T. Niemi, and M. Niinimki, "Study of virtualization energy-efficiency in high energy physics computing," in *Proc. ICGREEN'12*, 2012.

[3] B. Schäppi, F. Bellosa, B. Przywara, T. Bogner, S. Weeren, and A. Anglade, "Energy efficient servers in europe," Austrian Energy Agency, Tech. Rep. October, 2007.

[4] E. STAR, "Report to congress on server and data center energy efficiency," U.S. Environmental Protection Agency ENERGY STAR Program, Tech. Rep., 2007.

[5] L. A. Barroso and U. Hölzle, "The case for energy-proportional computing," *Computer*, vol. 40, pp. 33–37, 2007.

[6] W. Vogels, "Beyond server consolidation," *Queue*, vol. 6, pp. 20–26, January 2008.

[7] D. Meisner, B. T. Gold, and T. F. Wenisch, "Powernap: eliminating server idle power," in *Proceeding of the 14th international conference on Architectural support for programming languages and operating systems*, ser. ASPLOS '09. Washington, DC, USA: ACM, 2009, pp. 205–216.

[8] U. Hölzle and B. Weihl, "High-efficiency power supplies for home computers and servers," Google, Tech. Rep., 2006.

[9] A. Verma, P. Ahuja, and A. Neogi, "pmapper: power and migration cost aware application placement in virtualized systems," in *Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware*, ser. Middleware '08. New York, NY, USA: Springer-Verlag New York, Inc., 2008, pp. 243–264.

[10] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities," in *High Performance Computing and Communications, 2008. HPCC '08. 10th IEEE International Conference on*, sept. 2008, pp. 5 –13.

[11] M. Xu, Z. Hu, W. Long, and W. Liu, "Service virtualization: Infrastructure and applications," in *The grid: blueprint for a new computing infrastructure*. Wiley, 2004, ch. 14.

[12] M. Fenn, M. A. Murphy, and S. Goasguen, "A study of a kvm-based cluster for grid computing," in *Proceedings of the 47th Annual Southeast Regional Conference*, ser. ACM-SE 47. New York, NY, USA: ACM, 2009, pp. 34:1–34:6. [Online]. Available: http://doi.acm.org/10.1145/1566445.1566492

[13] N. Regola and J.-C. Ducom, "Recommendations for virtualization technologies in high performance computing," in *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on*, 30 2010-dec. 3 2010, pp. 409–416.

[14] L. Nussbaum, F. Anhalt, O. Mornard, and J.-P. Gelas, "Linux-based virtualization for hpc clusters," *Network*, pp. 221–234, 2009.

[15] A. Verma, P. Ahuja, and A. Neogi, "Power-aware dynamic placement of hpc applications," in *Proceedings of the 22nd annual international conference on Supercomputing*, ser. ICS '08. New York, NY, USA: ACM, 2008, pp. 175–184.

[16] H. Liu, C.-Z. Xu, H. Jin, J. Gong, and X. Liao, "Performance and energy modeling for live migration of virtual machines," in *Proceedings of the 20th international symposium on High performance distributed computing*, ser. HPDC '11. ACM, 2011, pp. 171–182.

[17] P. Padala, X. Zhu, Z. Wang, S. Singhal, and G. Shin, K., "Performance evaluation of virtualization technologies for server consolidation," *Work*, no. HPL-2007-59, p. 15, 2007.

[18] T. Deshane, Z. Shepherd, J. Matthews, M. Ben-Yehuda, A. Shah, and B. Rao, "Quantitative comparison of xen and kvm," in *Xen summit*. USENIX association, June 2008.

[19] P. Petrides, G. Nicolaides, and P. Trancoso, "Hpc performance domains on multi-core processors with virtualization," in *Proceedings of the 25th international conference on Architecture of Computing Systems*, ser. ARCS'12, 2012, pp. 123–134.

[20] J. Dongarra, P. Luszczek, and A. Petitet, "The linpack benchmark: past, present and future," *Concurrency and Computation Practice and Experience*, vol. 15, no. 9, pp. 803–820, 2003. [Online]. Available: http://doi.wiley.com/10.1002/cpe.728

[21] B. Martin, "Using bonnie++ for filesystem performance benchmarking," *Linuxcom*, vol. Online edi, 2008.

[22] M. Egli and D. Gugelmann, "Iperf - network stress tool," *Source*, pp. 1–2, 2007.

[23] J. Caffaro and S. Kaplun, "Invenio: A modern digital library for grey literature," in *Twelfth International Conference on Grey Literature*, Prague, Czech Republic, Dec 2010.

[24] D. Mosberger and T. Jin, "httperf - a tool for measuring web server performance," *SIGMETRICS Perform. Eval. Rev.*, vol. 26, pp. 31–37, Dec 1998.

[25] F. Fabozzi, C. Jones, B. Hegner, and L. Lista, "Physics analysis tools for the cms experiment at lhc," *Nuclear Science, IEEE Transactions on*, vol. 55, pp. 3539–3543, 2008.

[26] I. Antcheva and et al., "Root a c++ framework for petabyte data storage, statistical analysis and visualization," *Computer Physics Communications*, vol. 180, no. 12, pp. 2499 – 2512, 2009.

[27] T. Niemi, J. Kommeri, K. Happonen, J. Klem, and A.-P. Hameri, "Improving energy-efficiency of grid computing clusters," in *Advances in Grid and Pervasive Computing, 4th International Conference, GPC 2009, Geneva, Switzerland*, 2009, pp. 110–118.

[28] T. Niemi, J. Kommeri, and H. Ari-Pekka, "Energy-efficient scheduling of grid computing clusters," in *Proceedings of the 17th Annual International Conference on Advanced Computing and Communications (ADCOM 2009), Bengaluru, India*, 2009.