1 Title

Energy-efficient data centers based on heterogeneous hardware

2 Summary

Due to the an ever-growing number of computers in data centers, energy consumption in these data centers increases continually. Virtualization has enabled significant energy savings, but a further reduction of energy usage is possible. We propose to use a heterogeneous set of hardware configurations in data centers together with live migration in order to get virtual machines running on hardware with a desirable performance/watt ratio. These hardware configurations differ in processor, memory, accelerators, storage and network provisioning. By using sensors integrated in the hypervisor, we estimate how balanced the current hardware configuration is for a virtual machine and we migrate it to another machine when performance and energy usage models indicate this is beneficial. Using these methods, we expect to significantly decrease data centers' energy usage and explore what a good mix of hardware configurations will be for future data centers.

3 Description of the research

3.1 Introduction

With computers becoming more pervasive, their impact on the total worldwide energy consumption grows by the day. A large portion of energy used by IT equipment is consumed in data centers, which in the United States nowadays account for around 2% of the total energy used [7]. With the advent of wide-spread virtualization and the cloud paradigm, methods have been introduced that enable data center owners to significantly save on energy consumption. While the way in which these techniques are currently used, provides a giant leap forward from the conventional one-server-per-machine paradigm, we believe it can be taken a step further. In this proposal, we will look at an integrated approach to reduce energy even more by mapping servers to heterogeneous hardware configurations.

It is well-known that different types of computers consume different amounts of energy. A simple netbook consumes only a few watts, while a typical node in a high-performance cluster can use hundreds of watts. The cause for these orders of magnitude of difference is the hardware of which the machines are composed. Since even the two machines mentioned above share many hardware interfaces, it is often possible to run the same, unmodified workloads on both of them. Choosing a suitable hardware platform thus requires making a trade-off between speed and energy consumption. For data centers with a largely homogeneous set of machines, methods already exist to schedule a set of applications to different machines while taking power into account [11]. For heterogeneous sets of hardware, no such methods have been developed. We propose to create a framework that encourages a high level of heterogeneity, so workloads are more likely to find a hardware configuration which allows them to be run in a more energy-efficient way. Using such a framework, data center providers can continue to increase energy-efficiency in the face of increasing demands.

3.2 Our approach

In order to map virtual machines dynamically to heterogeneous hardware configurations, these hardware configurations have to provide a uniform interface to these machines. Due to the availability of a variety of hypervisors [3, 10, 14, 13] many tools exist that are able to provide such a uniform interface. Another requirement is the possibility of *live migration*, meaning that

running virtual machines can be moved from one physical hardware platform to another without downtime. This has been shown possible by Clark et al. [4]

With these two requirements in place, it is possible to develop a framework that dynamically maps virtual machines to different hardware configurations. The heterogeneity within these configurations can come from many components, not necessarily restricted to the boundary of one physical machine. These different components will be discussed in section 3.2.1. Our proposed methodology to exploit this heterogeneity and reduce power consumption will be laid out in section 3.2.2. By building on existing approaches to reduce energy consumption throughout the project, we increase the chance that our project is successful and we make sure that it is manageable by one PhD student in four years.

3.2.1 Heterogeneity components

Processors and memory One of the biggest energy consumers within a given machine is the processor. Several techniques exist to reduce the energy consumption for one CPU, most notably DVFS [12]. While these techniques are able to dynamically vary the amount of power used by a CPU, fundamental differences between CPUs can still result in varying energy intakes for a given task. For example, state-of-the-art Intel x86 processors have high floating-point performance, large caches, support for advanced vector operations, hardware AES instructions, etc. All of this functionality uses die size of the processor and consumes power when executing tasks. On the other end of the spectrum we can find processors that lack many of these features, such as the AMD Fusion platform. These processors will be significantly slower than their top-of-the-line counterparts, but will execute many tasks using considerably less energy. For non-CPU intensive tasks, such as I/O-intensive tasks or task where a large portion of the calculations is offloaded to an accelerator, large caches and more powerful instruction sets might not be preferable. We plan to offer virtual machines a rich set of machine instructions through the virtualization layer, which emulates these instructions if they are not implemented in the underlying hardware. Emulation of machine instructions is already being used in [3] for specific purposes. This way, we increase generality by allowing all applications to use these instructions, while only providing hardware support to applications that use them extensively.

One could argue that simply putting many virtual machines together on one physical machine ensures that processors run efficiently because they are almost never idle. This is a true statement, but it is not possible to do this when the virtual machines that are to be consolidated on one machine are not CPU-bound, but bound by one of the other factors mentioned in this section.

As for memory, there exist differences in power usage between different memory modules, but since CPUs are usually bound to a specific type of memory, the only configuration variables we can change are the amount of memory and the frequency of the memory bus.

Many-core accelerators Many-core accelerators (mainly GPUs) are able to speed up many kinds of numerical computations, often using significantly less energy than a CPU [9]. It is, however, very inefficient to fit one in every machine in a datacenter, since only a small portion of programs makes use of them. Our plan is to take an approach such as the one taken with *gVirtuS*[6], *rCUDA*[5] and *Hybrid OpenCL*[2], in which access to the accelerator is made possible via the virtualization layer or over the network, ensuring that every virtual machine can have a virtual accelerator in its software configuration and is only moved to a machine with a physical accelerator when accessing it with a high bandwidth and a low latency is necessary. These decisions will be left to the mapping algorithm (section 3.2.2).

Storage Since permanent storage is too expensive to migrate on-the-fly together with a virtual machine, it is placed outside of the machines in dedicated storage compartments. This is already common practice for data centers in which live migration is used. The performance of the virtual disks that are housed in these storage compartments is determined by the hardware of the

compartment itself and the type of hard drives it contains. Thus, we will have compartments with very fast, but power-hungry disks and compartments with slower, but more energy-efficient disks. Migration of a virtual disk from one compartment to another can take a considerably longer time than the migration of a virtual machine and is thus more expensive, consequently, decisions on these migrations will be handled separately from virtual machine migrations by the mapping algorithm.

Connectivity With ever-increasing bandwidth demands, networks are becoming an increasingly large consumer of energy. We will start with existing approaches to reduce energy usage in data center networks, such as [8] and [1], and try to modify these in order to support different *zones*. These zones will correspond to the trade-off between bandwidth capacity and energy usage. This way, virtual machines that are very computationally intensive, but require only little network communication, can be placed in a zone that is thinly provisioned bandwidth-wise in order not to waste energy by having routers on their paths that only use a small portion of their capacity.

3.2.2 Optimizing the mapping of virtual machines

With a heterogeneous data center as described in the previous section in place, we need algorithms that tell us how to map virtual machines to hardware configurations. These models will need input on how virtual machines are using their current underlying hardware. Together with information from energy meters connected to every node, we will use existing sensors at the hypervisor level and implement new ones that can provide us with information necessary to make scheduling decisions, such as: how high is the load on the processor and the memory; how much are processor instruction set extensions being used; how much are accelerators being used; how many I/O-operations are being performed per second and how much bandwidth to the storage is being used; how big is the load on the network; etc. We will also explore the possibility to incorporate user-definable sensors such as the processing time for a user-specified task or the latency for a specific remote request. By developing models similar to the ones in [11], while taking estimated performance and energy usage on another hardware configuration into account, together with migration costs, we will be able to make decisions on where to move which virtual machine and what the accompanied change in performance and energy usage will be. To validate and fine-tune our models, we plan to obtain traces of real-world workloads from existing data centers.

Moving many machines around all the time may seem too expensive to offset the power reduction by running virtual machines on more suitable hardware. We conjecture that there will be an initial phase in which a lot of moving around is necessary to find a good fit for a virtual machine, but that later on the virtual machine mapping will stabilize and migration will only be necessary if the applications running on a virtual machine change or when virtual machines are added to or removed from the set of running instances.

3.3 Planning

Activities for the PhD student (4 years):

- First year: Review literature on optimizing energy-efficiency in data centers; explore hardware possibilities in the different degrees of heterogeneity; compare hypervisors; compose benchmark systems and perform micro-benchmarks; design experiment testbed.

- Second year: Assemble and configure testbed; start developing optimization framework, performance and energy usage models; integrate sensors into hypervisor; incorporate accelerator virtualization in hypervisor; obtain traces of real-world workloads.

- Third year: Further develop framework and models; run experiments with obtained traces; use results to determine good combinations of hardware configurations for future data centers.

- Fourth year: Finalize experiments and process their results; write a thesis on the use of heterogeneous hardware to increase energy efficiency.

4 Literature

[1] D. Abts et al. "Energy proportional datacenter networks". In: *ACM SIGARCH Computer Architecture News* 38.3 (2010), pp. 338–347.

[2] R. Aoki et al. "Hybrid OpenCL: Enhancing OpenCL for Distributed Processing". In: *Parallel and Distributed Processing with Applications (ISPA), 2011 IEEE 9th International Symposium on*. IEEE. 2011, pp. 149–154.

[3] P. Barham et al. "Xen and the art of virtualization". In: *ACM SIGOPS Operating Systems Review* 37.5 (2003), pp. 164–177.

[4] C. Clark et al. "Live migration of virtual machines". In: *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2*. USENIX Association. 2005, pp. 273–286.

[5] J. Duato et al. "rCUDA: Reducing the number of GPU-based accelerators in high performance clusters". In: *High Performance Computing and Simulation (HPCS), 2010 International Conference on*. IEEE. 2010, pp. 224–231.

[6] G. Giunta et al. "A GPGPU transparent virtualization component for high performance computing clouds". In: *Euro-Par 2010-Parallel Processing* (2010), pp. 379–391.

[7] Silicon Valley Leadership Group. *Data Center Energy Forecast*. URL: `http://svlg.org/campaigns/datacenter/docs/DCEFR\_report.pdf` (visited on 04/12/2011).

[8] B. Heller et al. "ElasticTree: Saving energy in data center networks". In: *Proceedings of the 7th USENIX conference on Networked systems design and implementation*. USENIX Association. 2010, pp. 17–17.

[9] S. Huang, S. Xiao, and W. Feng. "On the energy efficiency of graphics processing units for scientific computing". In: *Parallel & Distributed Processing, 2009. IPDPS 2009. IEEE International Symposium on*. IEEE. 2009, pp. 1–8.

[10] A. Kivity et al. "kvm: the Linux virtual machine monitor". In: *Proceedings of the Linux Symposium*. Vol. 1. 2007, pp. 225–230.

[11] H. Lim, A. Kansal, and J. Liu. "Power budgeting for virtualized data centers". In: *2011 USENIX Annual Technical Conference (USENIX ATC11)*. 2011, p. 59.

[12] P. Macken et al. "A voltage reduction technique for digital systems". In: *Solid-State Circuits Conference, 1990. Digest of Technical Papers. 37th ISSCC., 1990 IEEE International*. IEEE. 1990, pp. 238–239.

[13] *Microsoft Hyper-V*. URL: `http://www.microsoft.com/hyper-v-server/` (visited on 04/12/2011).

[14] *VMWare vSphere*. URL: `http://http://www.vmware.com/products/vsphere/` (visited on 04/12/2011).